Magento Website User Guide

This user Magento guide will show you how to harness the power of your ecommerce website

Table of Contents

Chapter 1: Managing Products

Chapter 2: Configuring to Sell

Chapter 3 : Managing Non-product Content

Chapter 4 : Marketing Tools

1. Managing Products

After successfully installing Magento, you can now take on the task of creating and configuring your store. You could begin crafting the design that reflects your store's brand, or you could start configuring the many settings that will direct how your customers will interact with your online store.

However, selling online really boils down to the products that you're selling. Additionally, many of Magento's configurations depend on the products that you're offering and how they are arranged into categories.

Therefore, when I create a new Magento-powered store, I begin at the "root," so to speak: the products.

In this chapter, we will tackle the following:

- Creating catalogs and categories
- Adding products manually and importing en masse
- Setting up reviews, tags, and feeds that help promote your products

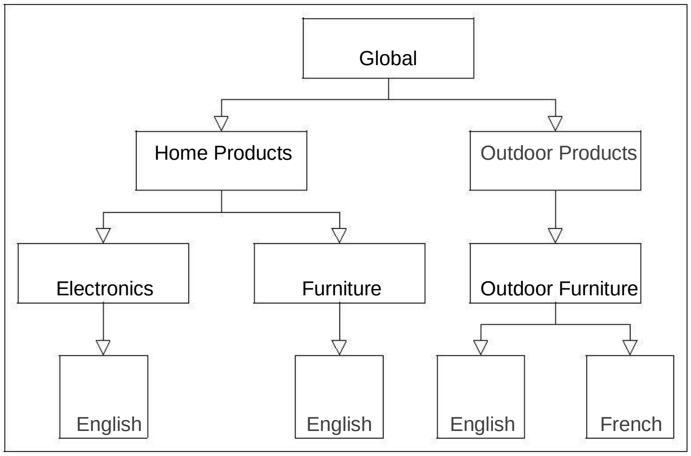
Catalogs and categories

The use of the terms "catalogs" and "categories" in Magento can be a bit confusing, as Magento tends to use these terms with some inconsistency. Let us define these according to our purposes.

Understanding catalogs

In *Chapter 2*, *Successful Magento Installation*, I created two new web sites, three new stores, and four store views. Each store was assigned to a root category which we created under **Catalogs | Manage Categories** in the Magento backend.

I consider each root category as a catalog, a collection of products to be sold within a given Magento store. In my example, as illustrated in the next diagram, we have three stores, each with its own catalog: selling electronics, furniture, and outdoor furniture. Each catalog will have its own hierarchy of categories and its own selection of products.



What if you have products that belong in all Catalogs, such as—to continue our example—wall clocks? No worries. With Magento, you can assign the same product to more than one catalog, and, as you'll see later in this chapter, to more than one category within a catalog.

I like to think of a catalog as just that: A catalog much like you used to get from *Argos*in days gone by. When I was a kid, every year we received the huge *Argos*catalog in the mail, free of charge.
You could also request more specific catalogs, such as appliances or tools. These "specialty" catalogs contained many of the same products found in the main catalog, but also had other products not found in the main catalog. With Magento, we could create the same catalog assortment as Sears produced, only online.

Understanding categories

With catalogs now defined, limiting categories to the subdivision within a catalog is more succinct. This distinction is important though, because if you've had a chance to look around your new Magento installation—and provided you installed the sample data—you'll see that while the top navigation menu in the Admin backend is labelled **Catalogs**, you actually create "root categories" to represent individual catalogs. Magento goes further in this vein of confusion by calling the top, sample root category, "Root Catalog".

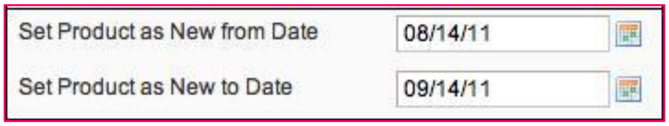
To help illustrate how we can use a common language to plan our product classifications, let's use the following outline for planning the three example catalogs that we described earlier. Catalogs are in **bold**, while categories are indented:

- **Electronics**
  - Personal
  - Home Entertainment
  - Computers
  - Accessories
- **Furniture**
  - Living
  - Dining
  - Bedroom
  - Accessories

While there are similarly named categories among the catalogs, each would contain only those products relevant to the purpose of the catalog. As mentioned before, you will still be able to assign products to more than one catalog, as well as to more than one category within a catalog.

Special categories

Magento provides some inherent tools used while grouping products for special display purposes. For example, by designating **New From** and **New To** dates in a **Product Detail** screen, as shown in the next screenshot, Magento will display a product within a **New Products** block if today's date falls within the range of these dates:



Set Product as New from Date 08/14/11

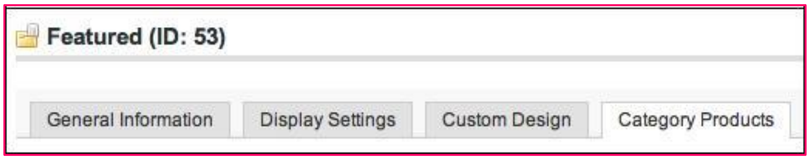
Set Product as New to Date 09/14/11

However, in some cases, you may want to display groups of products for other reasons. It's not uncommon to show "Featured" products on an e-commerce website. You may even want to show products grouped by family or purpose.

Let's take the case of creating a "Featured" products section for our home page. Let's also assume that you don't want "Featured" as a category in your navigation bar, just as a "special" Category.

1. Go to **Catalog | Manage Categories** in your Magento backend.

2. Click on the root category under which you wish to create your special category.
3. Click on **Add Subcategory**.
4. In the central part of the screen, enter the following values:
    - **Name**: *Featured*
    - **Is Active**: *Yes*
    - **Include in Navigation Menu**: *No*
5. Click on the tab at the top labelled **Category Products**.
6. Find the product that you wish to add to this category and check the box in the left-most column.
7. Click on **Save Category**.
8. After the screen refreshes, note the **ID** number of the category at the top of the screen, as shown in the following screenshot (in this example, the category **ID** is "53"):



You've now created a new category called **Featured** and added some products. We now need to add a block to the home page that will display your featured products.

9. Go to **CMS | Pages**.
10. Click on the **Home** page for the store that you wish to update.
11. Click on the side tab labelled **Content**.
12. If the WYSIWYG editor is showing, click on **Show / Hide Editor** to reveal the HTML code.
13. In the code, find where you want to put your **Featured Products** section, and insert the following:

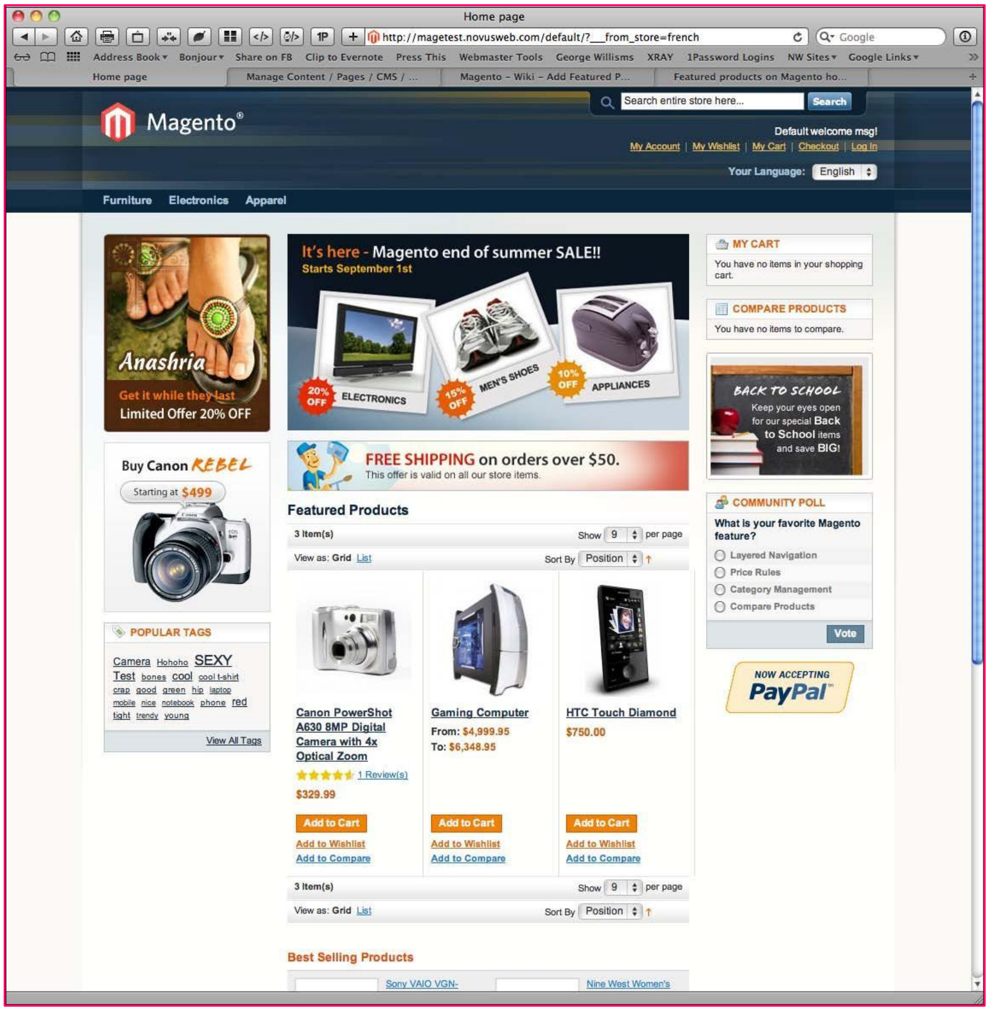
```
{{block type="catalog/product_list"
    category_id="53"
    template="catalog/product/list.phtml"}}
```

14. You may want to add a title before this, such as the following:

```
<h3>Featured Products</h3>
```

15. Click on **Save Page** (or **Save and Continue Edit**).

    When you view the home page, you should see a section displaying the featured items you assigned to this special category:



Home page

http://magetest.novusweb.com/default/?___from_store=french

Default welcome msg!

Your Language: English

It's here - Magento end of summer SALE!!
Starts September 1st

FREE SHIPPING on orders over $50.
This offer is valid on all our store items.

MY CART
You have no items in your shopping cart.

COMPARE PRODUCTS
You have no items to compare.

BACK TO SCHOOL
Keep your eyes open for our special Back to School items and save BIG!

COMMUNITY POLL
What is your favorite Magento feature?
○ Layered Navigation
○ Price Rules
○ Category Management
○ Compare Products

Featured Products

3 Item(s) Show 9 per page
View as: Grid List Sort By Position

Canon PowerShot A630 8MP Digital Camera with 4x Optical Zoom	Gaming Computer	HTC Touch Diamond
1 Review(s) $329.99	From: $4,999.95 To: $6,348.95	$750.00

3 Item(s) Show 9 per page
View as: Grid List Sort By Position

Best Selling Products

Sony VAIO VGN- Nine West Women's

Furthermore, you can access this category and its products by appending the name of the special category to your store URL. For example, to see the entire **Featured Products** category, you can go to http://www.yourstoredomain.com/featured.html.

Managing products the customer-focused way

The heart of any online store is the selection of products offered to visiting customers. Yet, as simple as that may sound, creating online stores to present the vast array of products and product types has proven to be one of the most challenging quests for platform programmers.

If all stores sold each product as an individual item without different colours, sizes, or add-ons, e-commerce would be much simpler. But, that's not how the real world works. If you sell T-shirts (the classic example for this discussion), you might sell each colour as a separate item, especially if you only offer a few shirts. However, it would make shopping very cumbersome for your customers if you also had each size of each colour listed as a separate product.

People shop by product style, then decide upon variations, such as size and colour. To reflect this shopping "workflow," we need to create products in our store that are presented in the most convenient and logical manner possible.

Types of products

To accommodate the many different types of products that can be sold online, Magento groups products into five types, listed next.

Simple products

Simple products are those which are sold individually. For example, if you're selling watches, you may offer different types of bands for each watch; but in general, the product is sold as an individual unit.

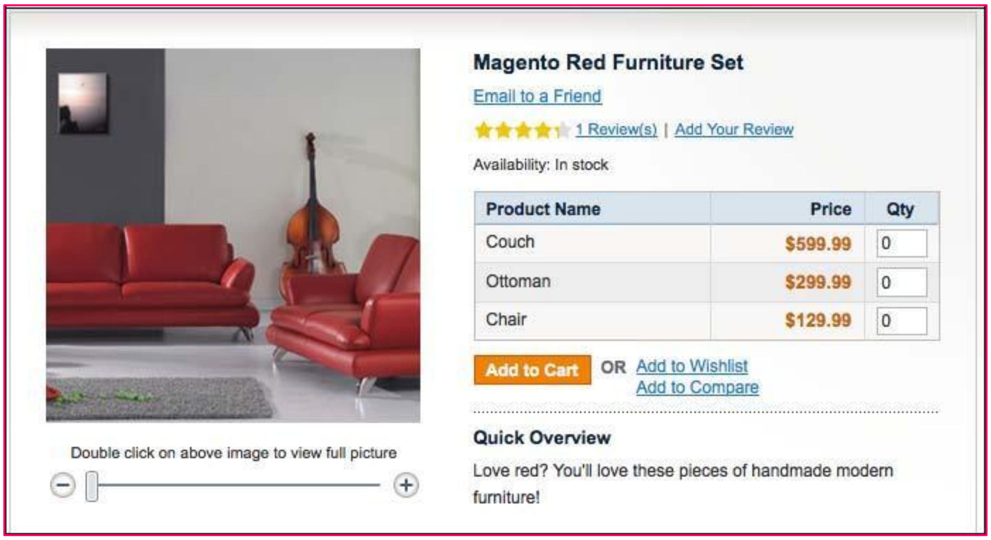
Complex products

In Magento, a complex product refers to a product that has to be configured differently, as it generally relies on other products in your store, or is not a physical product. To clarify, let's explore each type of complex product.

Grouped products

A grouped product is one that displays several similar products on a single product page. Shoppers can buy only those items in the group that they wish to add individually. Grouping products is a wonderful way to display coordinated items.

The sample data that you installed includes a grouped product for a furniture set, shown as follows:



Magento Red Furniture Set

Email to a Friend

1 Review(s) | Add Your Review

Availability: In stock

Product Name	Price	Qty
Couch	$599.99	0
Ottoman	$299.99	0
Chair	$129.99	0

Quick Overview

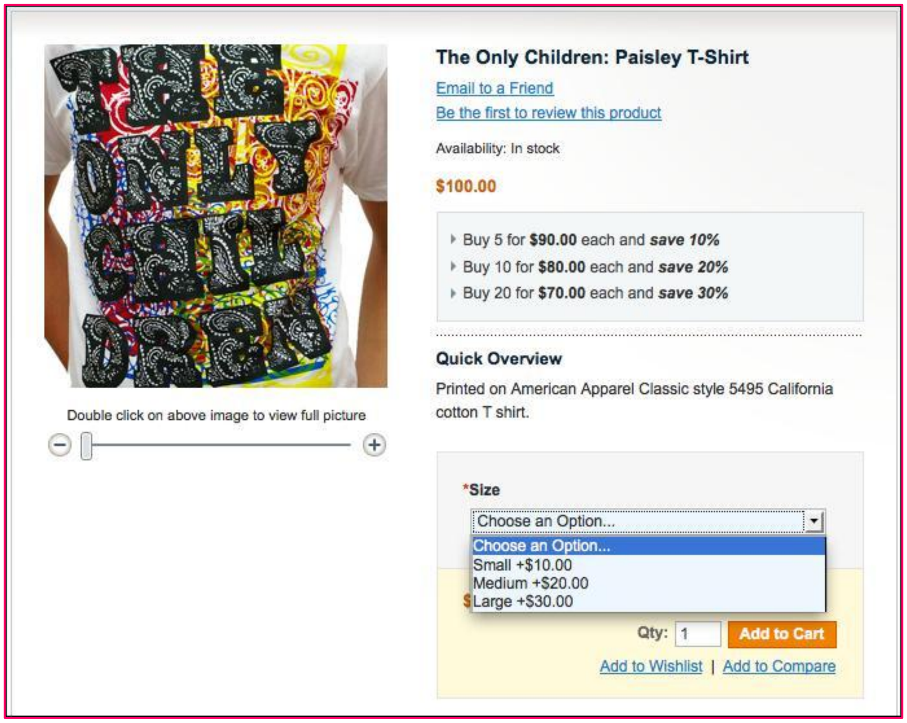
Love red? You'll love these pieces of handmade modern furniture!

Configurable products

Some products are offered with variants, such as colours and sizes. If the product variants do not require separate **SKUs** (**Stocking-keeping units**), you could use a simple product type, with an option to select a size. However, in most cases, each variant of a product requires a different SKU. Many also are priced differently. For example, if you're selling a basketball shoe, it might be one price for sizes 5-12, but cost an additional $15 for sizes 13 and above. To track inventory for each size, each individual product SKU needs to be entered as a separate simple product. But, you don't want to show 12 listings of the same shoe, one for each size; that would be cumbersome for your online customers.

With configurable products, you can create each individual SKU as a simple product, but show them all as one product with variants. You can even have more than one variant. Your basketball shoe might come in different sizes and colours.

The next screenshot shows a configurable product included in the sample data. Notice the **Size** drop-down menu and how the price changes for each variant included.



The Only Children: Paisley T-Shirt

Availability: In stock

$100.00

- Buy 5 for **$90.00** each and ***save 10%***
- Buy 10 for **$80.00** each and ***save 20%***
- Buy 20 for **$70.00** each and ***save 30%***

Quick Overview

Printed on American Apparel Classic style 5495 California cotton T shirt.

*Size: Choose an Option...

Qty: 1

Email to a Friend
Be the first to review this product
Double click on above image to view full picture
Choose an Option...
Small +$10.00
Medium +$20.00
Large +$30.00
Add to Cart
Add to Wishlist | Add to Compare

Configurable products are one of the most powerful product features of Magento. I am finding more and more ways to use this to better present products for our clients, as I'm sure you will, too.

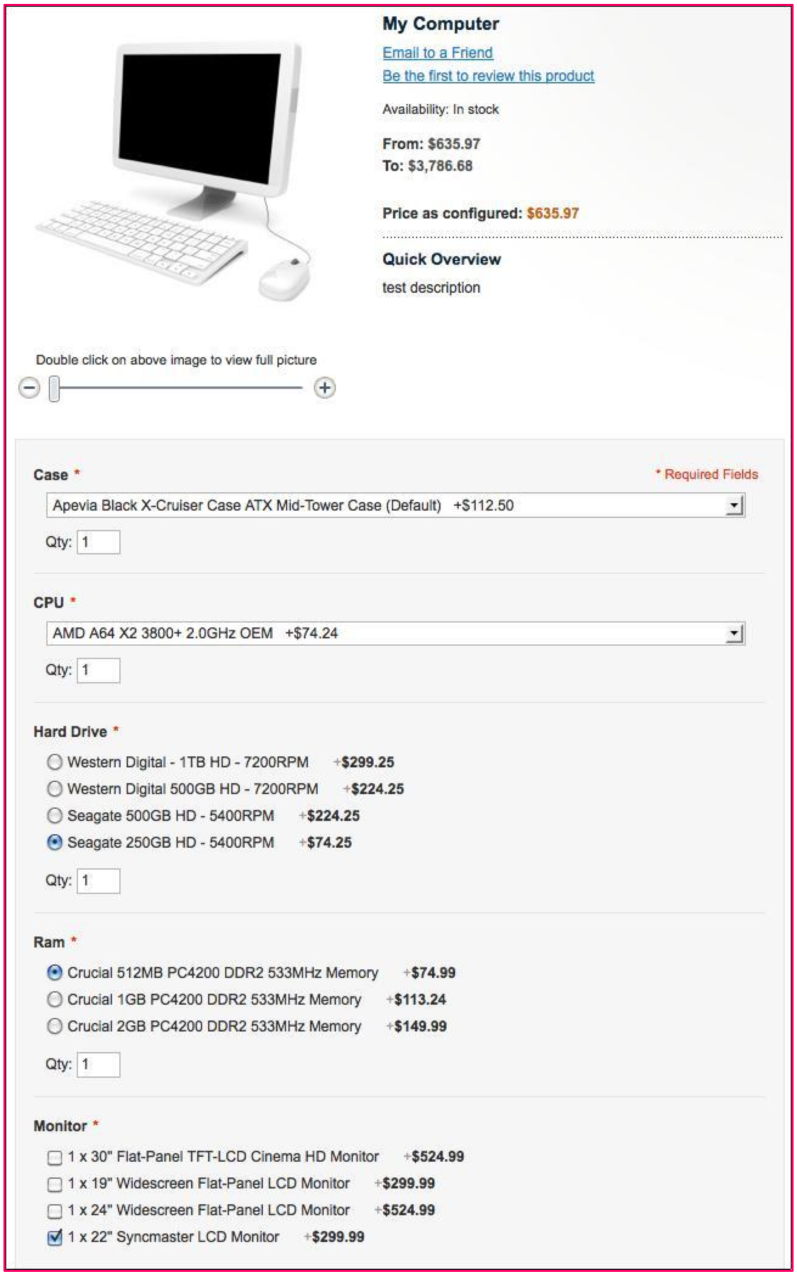
Virtual products

Products that don't require inventory or shipping, such as warranties or services, are called virtual products. As you'll see later in this chapter, the creation of a virtual product is very similar to creating a simple product.

Bundle products

Bundle products are perhaps one of the most interesting product types in Magento because their application as a product allows you to offer some amazing customization capabilities to your customers.

The classic example of a bundle product is that of the "Build-Your-Own" computer, as shown on the following page. By bundling many different products together, you can give your customers the ability to create their desired product simply by choosing from an array of possible choices.



My Computer

Email to a Friend
Be the first to review this product

Availability: In stock

From: $635.97
To: $3,786.68

Price as configured: $635.97

Quick Overview

test description

Case * 				* Required Fields

Apevia Black X-Cruiser Case ATX Mid-Tower Case (Default) +$112.50

Qty: 1

CPU *

AMD A64 X2 3800+ 2.0GHz OEM +$74.24

Qty: 1

Hard Drive *

○ Western Digital - 1TB HD - 7200RPM +**$299.25**
○ Western Digital 500GB HD - 7200RPM +**$224.25**
○ Seagate 500GB HD - 5400RPM +**$224.25**
● Seagate 250GB HD - 5400RPM +**$74.25**

Qty: 1

Ram *

● Crucial 512MB PC4200 DDR2 533MHz Memory +**$74.99**
○ Crucial 1GB PC4200 DDR2 533MHz Memory +**$113.24**
○ Crucial 2GB PC4200 DDR2 533MHz Memory +**$149.99**

Qty: 1

Monitor *

□ 1 x 30" Flat-Panel TFT-LCD Cinema HD Monitor +**$524.99**
□ 1 x 19" Widescreen Flat-Panel LCD Monitor +**$299.99**
□ 1 x 24" Widescreen Flat-Panel LCD Monitor +**$524.99**
☑ 1 x 22" Syncmaster LCD Monitor +**$299.99**

Amazingly, Magento handles all the pricing—including the **From** and **To** price range shown—automatically from the products you add into the bundle.

Downloadable products

Products that are distributed electronically, such as eBooks, music files, or computer programs, are configured in Magento as downloadable products. By using this product type, Magento helps you manage how the products are distributed and the number of times they can be downloaded.

Attributes and attribute sets

Before we can begin adding products, we have to learn about one very important aspect of Magento product management: attributes. Once you understand how attributes are used in creating and managing products you may, as I did, find new and creative ways to use attributes for your own store websites.

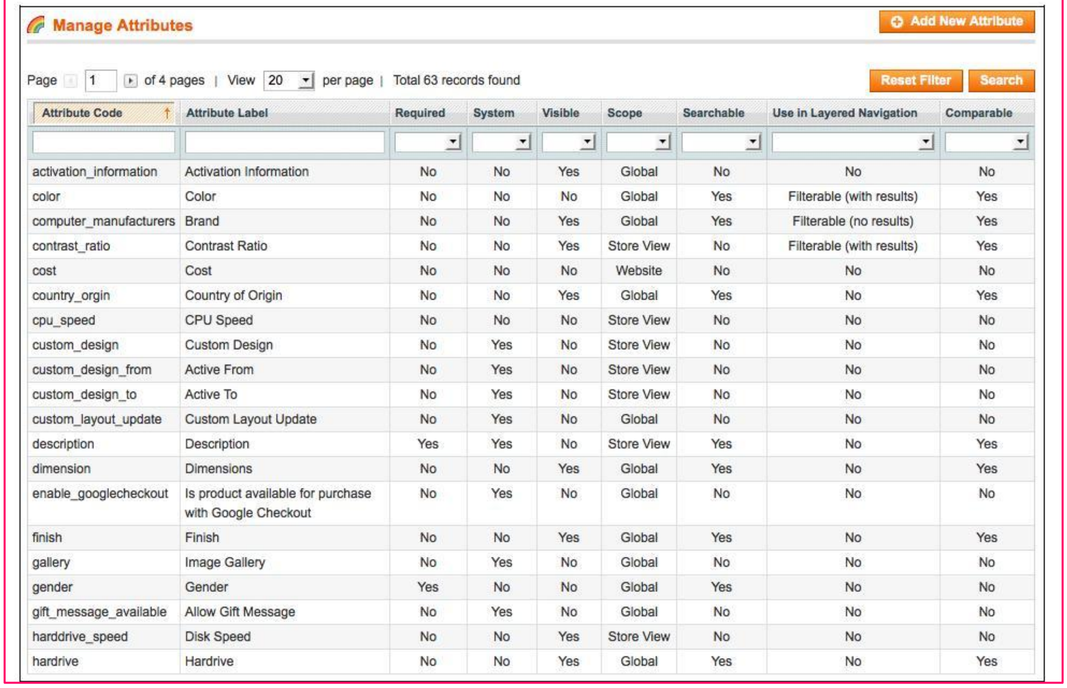
Product attributes

Every product in Magento contains a number of fields, such as price, SKU, name, and so on. Each of these can be considered as attributes of a product. Attributes, in essence, serve to describe the different features of a product.

Magento gives you the ability to alter some default attributes. More importantly, you can create new attributes that will help you provide better information to your customers. For example, if you sell furniture, you may want a field to describe the wood finish or upholstery cloth. Certain attributes can be used in the layered navigation or as comparison fields when comparing different products.

Attributes are added to attribute sets (see the next section) so that they will be included as part of the **Product** edit screen. If you want to enter sizes when creating products, for example, you'll need to have an attribute for this as part of the **Product** edit screen.

As shown in the next screenshot, this partial list of attributes, in part created from the sample data, shows a variety of different possible product features.


Manage Attributes
Add New Attribute
Page 1 of 4 pages | View 20 per page | Total 63 records found
Reset Filter
Search

Attribute Code	Attribute Label	Required	System	Visible	Scope	Searchable	Use in Layered Navigation	Comparable
activation_information	Activation Information	No	No	Yes	Global	No	No	No
color	Color	No	No	No	Global	Yes	Filterable (with results)	Yes
computer_manufacturers	Brand	No	No	Yes	Global	Yes	Filterable (no results)	Yes
contrast_ratio	Contrast Ratio	No	No	Yes	Store View	No	Filterable (with results)	Yes
cost	Cost	No	No	No	Website	No	No	No
country_orgin	Country of Origin	No	No	Yes	Global	Yes	No	Yes
cpu_speed	CPU Speed	No	No	No	Store View	No	No	No
custom_design	Custom Design	No	Yes	No	Store View	No	No	No
custom_design_from	Active From	No	Yes	No	Store View	No	No	No
custom_design_to	Active To	No	Yes	No	Store View	No	No	No
custom_layout_update	Custom Layout Update	No	Yes	No	Global	No	No	No
description	Description	Yes	Yes	No	Store View	Yes	No	Yes
dimension	Dimensions	No	No	Yes	Global	Yes	No	Yes
enable_googlecheckout	Is product available for purchase with Google Checkout	No	Yes	No	Global	No	No	No
finish	Finish	No	No	Yes	Global	Yes	No	Yes
gallery	Image Gallery	No	Yes	No	Global	No	No	No
gender	Gender	Yes	No	No	Global	Yes	No	No
gift_message_available	Allow Gift Message	No	Yes	No	Global	No	No	No
harddrive_speed	Disk Speed	No	No	Yes	Store View	No	No	No
hardrive	Hardrive	No	No	Yes	Global	Yes	No	Yes

Let's go over the various fields available when creating a new attribute. To begin, go to **Catalog | Attributes | Manage Attributes** in your backend. You should see a screen similar to the previous screenshot.

As an example, let's create a new attribute that we will use for our tennis wear to show the type of material used in our garments. As we go through this process, you'll get a better understanding of how attributes are created and the choices that you have to make while creating a new attribute. If you want to create our example attribute, enter the values shown in brackets ([ ]) after each field description.

Click on **Add New Attribute** at the upper right-hand side of your screen. If you look at the left sidebar, you'll see two tabs, **Properties** and **Manage Label / Options**. This means that this screen is actually two screens.

On the **Properties** screen:

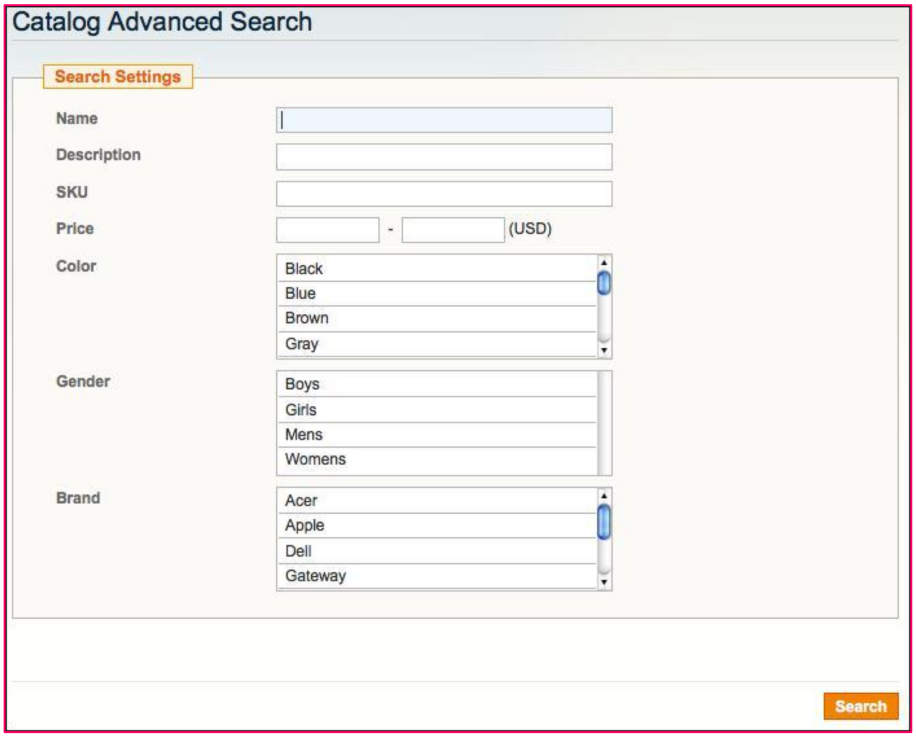
- **Attribute Code**: This is a unique value used by Magento in linking data together. Therefore, you can use any word or phrase, but substitute any spaces with underscores or hyphens. [Enter *garment_material*]

- **Scope**: This can be a bit confusing, so allow me to clarify how **scope** is used here. Think about how the attribute will be used for a product. In our example, the material of a garment will be the same at all scope levels. That is, we're unlikely to choose *cotton* for one store and *polyester* for another, for the same garment. Therefore, we would choose a **Global** scope. On the other hand, if you might offer a product in *red* in one store, but *green* in another, then the scope would be **Website** or **Store View**, depending on how you structure your stores. [*Global*]
- **Catalog Input Type for Store Owner**: In case you didn't already know— you're the Store Owner—the person who enters data into Magento, regardless of whether you actually own the store or not (you'll see this term used in various places throughout Magento). In this field, you will decide what type of entry can be made when the attribute is given a value. Each entry type has different consequences for how the attribute can be used. [*Dropdown*].
    - **Text Field**: A blank field in which you can enter whatever value you wish. This is useful for entering values such as voltage, length, or CPU speed. Values are limited to 255 characters.
    - **Text Area**: A larger blank field for data entry. You could use this field type for entering installation instructions or a list of minimum requirements.
    - **Date**: A date entry could be used to denote some future or past date, such as release date for a downloadable book.
    - **Yes/No**: If your attribute value, such as "Hardcover Available", can be answered as Yes or No.
    - **Multiple Select**: If you want to provide a set list of values for an attribute, but be able to select more than one for a given product, use a Multiple Select entry type. For example, you may want a list of available cloth types for your tennis wear. A garment might be available or may contain more than one cloth type. Therefore, with a Multiple Select field, you can select all the applicable types.
    - **Dropdown**: Basically, a Dropdown is the same as a Multiple Select, but you can only select one of the listed values.
    - **Price**: A new attribute using a Price field lets you create an alternative pricing option for a product. As we'll see later when we create products, Magento provides fairly extensive pricing options, such as Special Price, Tier Price, and Cost. However, you may want to include additional price values, such as Wholesale Cost and Distributor Cost for internal use.

- °° **Media Image**: This field type allows you to upload an image that could be used on the product's description page. You could use this to show a schematic or a brand logo for a product.
- °° **Fixed Product Tax**: Some governments impose fixed taxes, such as a VAT (value added tax), on certain goods. You could use a Fixed Product Tax type for entering this value for a product. This tax would be added to each purchase.

- **Default Value**: For appropriate field types, you can enter a default value that will appear whenever you create a new product that uses this attribute (not applicable for the Dropdown input type).
- **Unique Value**: If each product using this attribute is to have a unique value, set this to *Yes*. For example, a field for UPC code would be unique, as no two products can have the same UPC code. [*No*]
- **Values Required**: Set this to *Yes* if you want to force a value to be entered or selected for a product. [*No*]
- **Input Validation for Store Owner**: You can test the value entered to make sure that it conforms to certain criteria: *Decimal Number*, *Integer Number*, *Email*, *URL*, *Letters*, or *Letters and Numbers*. A UPC code, for example, would be an *Integer Number* (no decimals). [*Not applicable*]
- **Apply To**: Certain attributes may only be applicable certain product types. For example, an attribute for Assembly Time might only apply to Bundle products. [*All Product Types*]
- **Use To Create Configurable Products**: We'll learn more about configurable products later in this chapter when we create a configurable product, which will make this field more relevant. [*No*]
- **Use in Quick Search**: In Magento, a "Quick Search" is performed using the single search field shown on every page, such as the one in the next screenshot. When searching for a match, this attribute, if this value is *Yes*, will be used as a search matching value. [*Yes*]

- **Use in Advanced Search**: One of the default pages in Magento is an Advanced Search. If you set this field to *Yes*, this attribute will be added to the list of possible search criteria on the **Advanced Search** page. The next screenshot shows an **Advanced Search** page from the sample data store. [*Yes*]



- **Comparable on Front-end**: Magento includes the ability for customers to select two or more products and show them in a pop-up window in a comparison table. This allows customers to compare attributes across the selected products. If you want this attribute to be included in this comparison chart, choose *Yes*. [*Yes*]
- **Use in Layered Navigation**: The use of attributes in layered navigation can be a powerful shopping tool for your customers. Since Magento needs to set values in order to construct the layered navigation criteria, only the *Dropdown*, *Multiple Select*, and *Price* field types can be used in layered navigation. *Filterable (with results)* means that this attribute will only appear in layered navigation if there are actual products with attribute values matching one of the values in this attribute. *Filterable (without results)* will show this attribute in layered navigation at all times. [*Filterable (with results)*]

- **Use in Search Results Layered Navigation**: When customers do a product search on your site, the results include a layered navigation sidebar. As with the previous field, only the *Dropdown*, *Multiple Select*, and *Price* field types can be used. [*Yes*]
- **Use for Promo Codes**: As we'll cover in *Chapter 5*, *Configuring to Sell*, promotional codes can be used to provide discounts to customers. These codes are established using rules, such as "price over $100" or "shipping weight under 100 pounds". You could use an attribute to create promotions, such as "green coloured garments" (to promote St. Patrick's Day) or "XXL sizes" (to move an overstocked inventory of large shirts). [*No*]
- **Position**: If you want to manipulate the order in which layered navigation attributes are listed, you can use this field, entering numbers in ascending order for top-to-bottom positions. [*Leave blank*]
- **Allow HTML Tags on Frontend**: When entering text values, you may want to include HTML tags, such as <strong> for bold or <i> for italics. Magento, by default, filters these out to avoid the possibility of breaking your design.

  You can, however, allow HTML tags if you feel it necessary to use them in presenting an attributes, content. For example, you may want to be able to enter CPU speeds with "MHz" in italics for added emphasis. [Not applicable]
- **Visible on Product View Page on Front-end**: In most product detail page designs, there will be a section called "Additional Information". This section contains a table listing all attributes for the product in which this field is set to *Yes*. [*Yes*]
- **Used in Product Listing**: If *Yes*, this attribute will always be shown in a search results list, if your design is modified to show this attribute. I've found that most designs do not automatically add new attributes with this field set to *Yes*. [*No*]
- **Used in Sorting in Product Listing**: Earlier, we talked about changing the default sorting for a category, such as by *Best Value*, *Name*, or *Price*. By setting this field to *Yes*, you will create a new sorting criteria for lists that include products with this attribute. For example, you could add "CPU Speed" as a sorting criteria for computer products. [*No*]

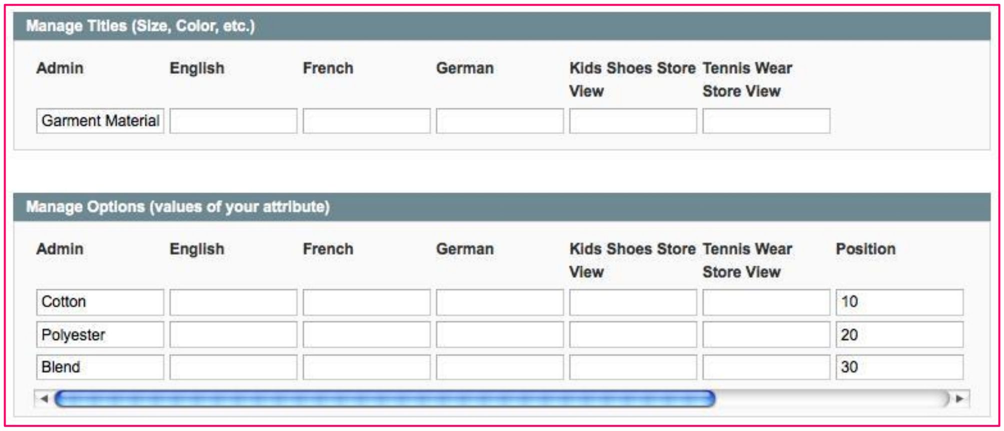
Before you can save your new attribute, you still need to add at least one more piece of data. Click on the left sidebar menu called **Manage Label / Options**.

For your attribute to be understood on the frontend, you need to provide a title, such as *Garment Material*. This value should be entered into the first space labelled **Admin**.

If you want the title to be different for your stores, enter the other values in the spaces shown. Any spaces left blank will automatically use the **Admin** value.

When you create an attribute with multiple selections, such as with our garment material example, an additional table called **Manage Options** will be visible, such as the one shown in the next screenshot. This chart is used to list the possible choices that you will use for this attribute when you create or edit a product.

To add a choice, click on **Add Option** and fill in the information, using the same process you did for the title. You can add as many options as you like. For our garment material, let's add **Cotton**, **Polyester**, and **Blend** in the **Admin** column, as shown in the next screenshot. In order that Magento list these in this particular order, enter **10** for Cotton, **20** for Polyester, and **30** for Blend in the **Position** column. Click on **Save Attribute** to save this example attribute, which we will be using in the next section.



Manage Titles (Size, Color, etc.)

Admin	English	French	German	Kids Shoes Store View	Tennis Wear Store View
Garment Material					

Manage Options (values of your attribute)

Admin	English	French	German	Kids Shoes Store View	Tennis Wear Store View	Position
Cotton						10
Polyester						20
Blend						30

After creating a new attribute, you may see a notice at the top of your screen that one or more of Magento's indexes need updating. Click on the link shown and re-index any index with a **REINDEX REQUIRED** message showing in the **Status** column.

Attribute sets

If you thought creating attributes was interesting (you have to admit that Magento attributes offer tremendous flexibility and power to your product management), hold on to your seat!

Before you can use attributes, you have to assign them to attribute sets. Each product that you create is first assigned to a set of related attributes for you to use. So, before you set out to add products to Magento, let's discuss some general concepts for using attribute sets:

- If you have a wide variety of product types (such as clothing, computers, shoes, and so on), consider creating attribute sets for each type, particularly if each product type will have different attributes. "Size" may be important for shoes, but not for computers. Alternatively, computers may include "voltage", which is not applicable to shoes. When segregating your attributes by product type, you will have better control over the attributes for each product. Additionally, any attributes for which **Visible on Product View Page on Front-end** is set to *Yes* will appear on the product detail page regardless of whether a value is provided or not. You don't want a shoe to show a N/A value for voltage; it might confuse your buyers ("Do your shoes have voltage?").
- If you have to import large quantities of products, assigning each to a different attribute set can be quite tedious and time-consuming. In this case, it may be wiser to create one attribute set for all products being imported, but make sure that attributes in the set are:
    - Not required for entry
    - Fairly generic (for example, "Power requirements" instead of "voltage", "watts", and "amps", separately)

I often have to import 20-30,000 products at a time; assigning different attribute sets for each product type would be too expensive in terms of time spent.

> One challenge with attribute sets is that once assigned to a product, you cannot re-assign the product to another attribute set. However, you can modify an attribute set, which will also update a product's attributes.
> Therefore, take some time to experiment with different attribute sets and add new products. View the results in your frontend to see how they appear to your customers. Before you get too far down the road in adding new products, it's best to get a firm handle on how attributes work and how they affect your stores. I can tell you from experience, it's no fun having to delete and re-add hundreds of products because you suddenly decide to assign the products to a new attribute set.

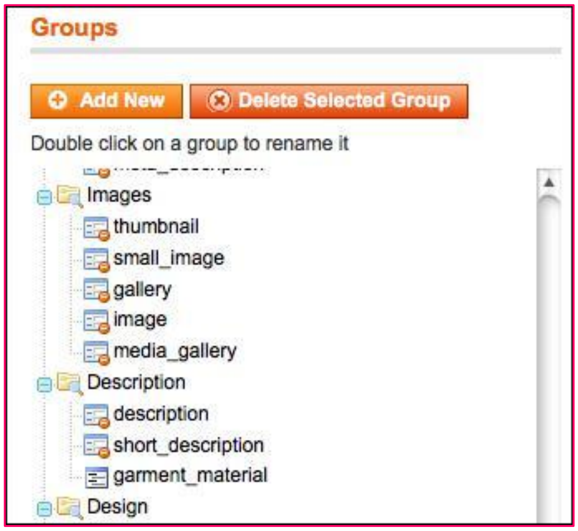
Creating an attribute set

To continue our example, let's create a new attribute set for our tennis wear. We'll plan on using this set for all tennis wear garments.

In your Magento backend:

1. Go to **Catalog | Attributes | Manage Attribute Sets**.
2. Click on **Add New Set** near the top of the screen.

3. Enter *Tennis Wear* for the **Name**.
4. To make creating an attribute set easier, Magento asks you to create your new set based on an existing set. For our example, choose *Shirts (General)* from the drop-down list (if you did not install the sample data, you have no other choice but *Default*).
5. Click on **Save Attribute Set**.
6. On the next screen are three columns. The first is the name of the attribute set. The second shows all the attributes assigned to this set, grouped into a folder hierarchy. The third column shows all unassigned attributes that can be added to the current set. For our attribute set, we want to add **garment_ material**. Scroll down the second column until you see the folder titled **Description** and the attributes within it. Using your mouse, drag **garment_ material** and position it anywhere within this folder, as shown in the
next screenshot:



7. To remove an attribute from an attribute set, drag it to the list in the third column.
8. Click on **Save Attribute Set**.
9. You should see a warning at the top of your screen asking you to re-index Magento. Click on the link shown and re-index any required indexes.

We have now created a new attribute set to use for tennis wear which will include an attribute for the material. Furthermore, this attribute will be shown on the product details screen on the frontend and will be shown if a customer chooses to compare two or more garments.

Managing inventory

Most of the time, the manner in which you manage your product inventory will be dictated by the settings you make in **System | Configuration | Inventory**. However, as with most system configurations, you can override your choices for each product.

Notice that inventory settings are Global in scope. Magento reasons that if you're managing inventory for a product in one store, you're also managing inventory in all stores for which the product might be assigned. Since some product inventory configurations are vague, let's go through each one. These are found in the Product Detail (**Catalog | Manage Products**) screen under the **Inventory** side tab:

- **Manage Stock**: Do you want Magento to keep track of products that are purchased and subtract them from the available inventory? If set to *No*, only **Minimum Qty Allowed in Shopping Cart** and **Maximum Qty Allowed in Shopping Cart** are available for configuration.
- **Qty**: A required field, this is the number of items you have on-hand to sell.
- **Qty for Item's Status to become Out of Stock**: If you want to keep some products in reserve, you can set a number below which, Magento will mark the product "Out of Stock." For example, if you want to keep five units in "reserve", Magento will show the item as "Out of Stock" by setting this field to 5 when the quantity remaining is 5 units.
- **Minimum Qty Allowed in Shopping Cart**: While *1* is a normal value, you might require a buyer to purchase a minimum number of a particular product.
- **Maximum Qty Allowed in Shopping Cart**: You may want to set a limit on the number of items that a buyer can purchase during a single transaction. There is no other limitation available, such as total purchased in one day or by one customer.
- **Qty Uses Decimals**: If you manage your product inventory in fractions of available inventory, set this to *Yes*.
- **Backorders**: If set to *Yes*, when a customer purchases a quantity of a product greater than available in inventory, they will be notified that some of the products will be on "backorder". If set to *No*, the customer cannot add more of a product to their shopping cart than is available in inventory.
- **Notify for Quantity Below**: As a store administrator, you may want to receive an e-mail alert whenever a product's inventory goes below a certain threshold; you might want to know when it's time to re-order stock.
- **Enable Qty Increments**: If you select *Yes*, the next field will appear, allowing you to set the increments at which a buyer can purchase your product.

- **Qty Increments**: In what quantity increments can the product be purchased?

> **Using "Quantity Increments" creatively**
>
> Let's assume you want to sell bottles of compressed air on your site, but in order to make enough margin to cover your overhead, you need a customer to buy a minimum of three bottles, although they can buy any quantity over that. In this case, you would enter for **Minimum Qty Allowed in Shopping Cart**, for **Enable Qty Increments**, and for **Qty Increments**. For another example, you might want to sell these bottles at a minimum quantity of *6*, with additional increments of *3* bottles at a time: set **Minimum Qty Allowed in Shopping Cart** to *6*, **Enable Qty Increments** to *Yes*, and **Qty Increments** to *3*.
>
> Unfortunately, you cannot configure Quantity Increments if you do not manage inventory. If you use a **drop shipper**, you may not be tracking inventory, as their inventory is available to other retailers as well. Still, you may want to require certain quantity increments. Hopefully, this will be added in future versions of Magento.

- **Stock Availability**: If you want the product saleable, select *In Stock*. When managing inventory, if the inventory level drops to zero (or below the floor you entered for **Qty for Item's Status to become Out of Stock**), the status for the product will automatically show as **Out of Stock** on the frontend, but will remain **In Stock** here. If you do not want the product to show as **In Stock** regardless of inventory level, choose *Out of Stock*.

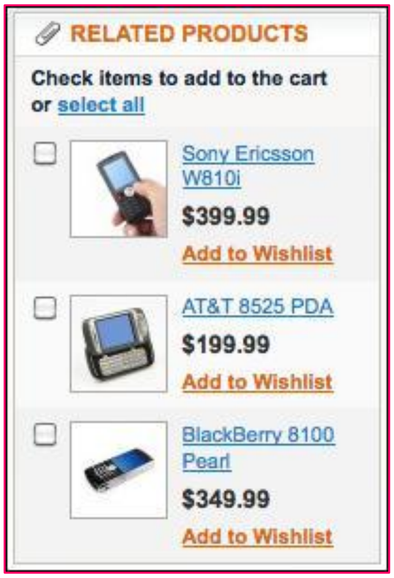
Related products, up-sells, and cross-sells

For each product that you create in Magento, you have the opportunity to attract additional purchases from customers by linking products together. These cross-links allow you to encourage shoppers to consider buying products other than what they reviewed. In the brick-and-mortar world, we're constantly bombarded with similar promotions. For instance, if you go into the grocery store, you'll often find small, red coupon machines hanging off the shelves, encouraging you by their flashing red light to pull a coupon out while you're reaching the canned peaches that you came for. Throughout the store, placards and banners promote "2-for-1" specials, or "buy 1, get 1 free". The area around the checkout lane is crowded with magazines, razor blades, batteries, and gum, put there to catch your eye just as you're about to leave the store.

In a similar fashion, Magento allows you, the store administrator, to encourage shoppers to spend more during their online visit using related products, up-sells, and cross-sells. All three types are managed within the Product Detail screen.

Related products

As shown in the next screenshot, **related products** are other products that you want to show on a product details page as possible additional purchases, products *related* to the main product displayed. The customer can check any related products that they wish to add to the shopping cart at the same time that they add the main product. Therefore, related products are usually those for which a purchasing decision could be made without leaving the current product.



RELATED PRODUCTS

Check items to add to the cart or select all

□ Sony Ericsson W810i
$399.99
Add to Wishlist

□ AT&T 8525 PDA
$199.99
Add to Wishlist

□ BlackBerry 8100 Pearl
$349.99
Add to Wishlist

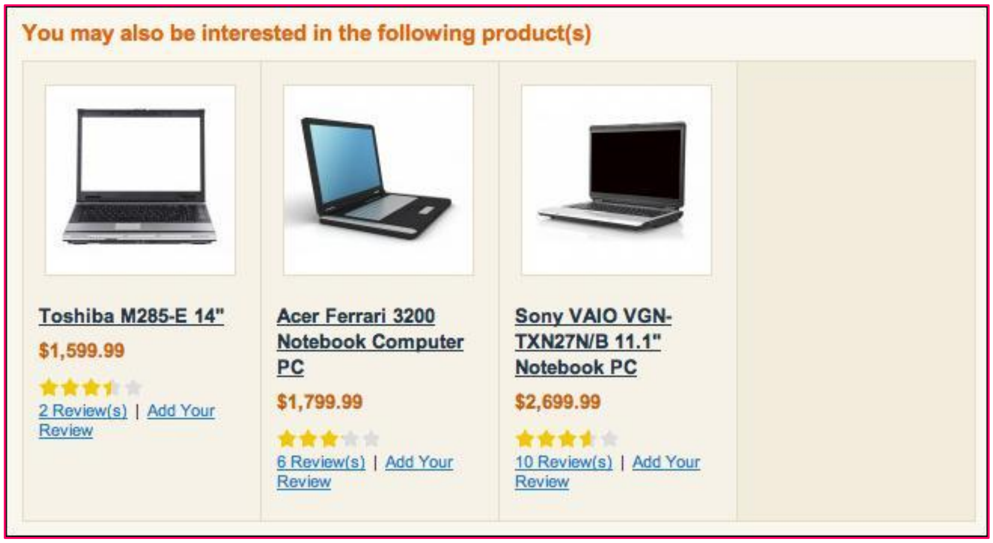
Some examples of related products might be:

- A carrying case for a smart phone
- An AC adapter for a portable speaker
- Leather cleaning kit for a sofa
- An extended warranty

A related product is one you want the customer to purchase *along with* the main product.

Up-sells

Like related products, **up-sells** are products that you wish to bring to your customer's attention. However, up-sells are generally considered to be products that may enhance the product purchased or help the customer notice other products to consider. Unlike related products, up-sells cannot be added to the shopping cart at the same time as the main product. Up-sells appear on the product details page, as shown in the next screenshot. They are generally considered products of a higher price or quality; products that you might want to encourage your customers to consider as an alternative to the currently viewed product.



You may also be interested in the following product(s)

Toshiba M285-E 14" $1,599.99 2 Review(s) | Add Your Review	Acer Ferrari 3200 Notebook Computer PC $1,799.99 6 Review(s) | Add Your Review	Sony VAIO VGN-TXN27N/B 11.1" Notebook PC $2,699.99 10 Review(s) | Add Your Review

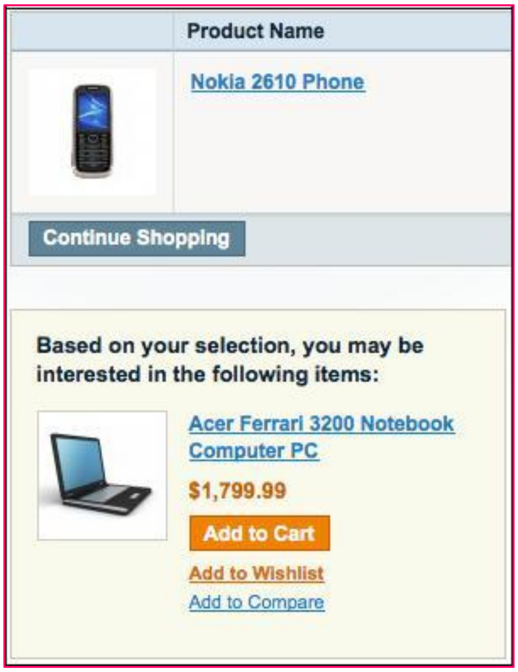
Consider these up-sell product examples:

- A king-sized candy bar (as opposed to the normal size shown)
- A brass version of a nickel-plated door knob
- A 17" laptop, as opposed to the 15" model

Just as fast-food restaurants offer "up-sells" on meals, you can entice customers to make larger—or at least alternative—purchases by showcasing select products.

Cross-sells

Yet another way to promote additional products! Cross-sells appear on the shopping cart as other possible products to consider. The next screenshot shows a portion of the shopping cart with a cross-sell product shown for the added item:



Based on your selection, you may be interested in the following items:

I consider cross-sells as products that we want to showcase to customers in order to introduce them to other product lines. For example, if we wanted to get camping enthusiasts to consider buying their backpacks from our store, we may offer select backpacks to those who buy camping stoves, sleeping bags, or hiking boots from us. Cross-sells are a creative way of encouraging shoppers to continue shopping after reaching the check-out page of your site.

Importing products

Creating products one-by-one is fine if you're only selling a few products, but many people choose Magento as a platform to sell hundreds and thousands of different products. Entering each product individually can take a long time, especially if you're writing unique product descriptions, uploading photos, and adding cross-linked products.

With Version 1.5, Magento has greatly improved the performance of product imports. In previous versions, importing thousands of products could take hours— literally. I once tried to import 19,000 products and the job took 20 hours to complete!
The improvements to Magento importing have greatly decreased that time.

Really fast importing

Magento's importing tool is still not the fastest alternative available.
While it works fine for a few thousand products, if you're building a store with tens of thousands of products, I highly recommend you take a look at **Magmi** [http://sourceforge.net/projects/magmi/]. Magmi is not so much an extension as it is a tool to help you rapidly import products—even complex types—in short order. In our use, Magmi imports products with photos at a rate of 2000 to 3000 per minute! Magmi is a freeware solution written by a French developer, and is constantly evolving. Go to the Magmi project [http://sourceforge.net/ projects/magmi/] for the latest downloads and documentation.

The shortcut to importing products

After spending hundreds of hours wrestling with Magento's importing schemas over several versions of the platform, I'm happy to say that from Version 1.5 there is a shortcut to help you successfully import products.

Magento imports CSV files according to a specific layout scheme. Trying to configure an import file to accurately import a variety of different products, product types, attribute sets, and so on, will lead to many, many failed attempts as you try different ways of configuring the .csv to meet your needs. Therefore, my best advice is to do the following:

1. Enter at least one product into your store, representing the different types (simple, complex, bundle, and so on) that you will be using.
2. Fully enter the data for each product, including all options. You don't have to upload images however.
3. Go to **System | Import/Export | Export** in your backend.
4. Export a product .csv file to your computer (in previous Magento versions, this export would be placed on your server and you would have had to FTP in order for your server to retrieve it).
5. Open the downloaded .csv file in Excel or Numbers.

Now, you can easily see how each type of product is configured in the export file.
Using this format, you can easily add additional products, simply by following a similar pattern.

Notice that for products with various options, particularly configurable products, only the differences between each iteration are included on the rows below the main product row. Never repeat any other fields that are the same between variations. A unique SKU is required of each product though.

To import images, you must upload the images to your server in the /media/import folder. If the /import folder doesn't exist, create it. Put the name of the image file in the image columns of your import .csv file (don't include any path). Magento will pick the image out of the /media/import folder, resize it as needed, and store it within the /images/catalog folder hierarchy. Never use capitals, spaces, or other strange characters in your image names; only the letters a through z, the numbers 0 through 9, and the underscore.

> The most important thing to remember is to remove any unused columns before importing. For example, if you are not going to include categories in your import, leave that column out. This is especially important if you are re-importing or importing product updates. If you leave a column blank, then Magento will enter a blank value for that column in your product record. Oops!

Summary

Selling online begins with your products. Your products have to be presented well, and in logical categories. Furthermore, Magento gives you additional powerful tools for promoting your products to potential customers.

In this chapter, we:

- Learned about how catalogs and categories relate to each other, and to your stores.
- Learned about the different types of products that can be sold in a Magento store.
- Went in-depth into how to create and manage attributes and attribute sets.
- Discussed inventory management configurations.
- Learned a shortcut for configuring product imports.

Getting a handle on how the complexity of categories and products in Magento is actually one of its greatest strengths, puts you in a better position to leverage the remaining topics in this book toward encouraging more sales from your visiting customers. Next, we turn our attention to *grabbing* attention: how to theme your Magento store.

2. Configuring to Sell

If you've been following the instructions in this book, chapter by chapter, then by now you will have created a working, accessible online store front. *Ready to take orders now?*

Not just yet. There's still more to do before you can swing open the virtual doors to your new Magento store. Specifically, we need to:

- Understand the Magento sales process
- Configure payment gateways to allow you to take online credit card payments
- Set up how your products will be shipped
- Configure sales tax rules
- Create customized outgoing e-mails

If you're the developer or designer of a Magento-powered site, this is usually the time when you consult with your client—the store owner—to learn how they want to take payments, charge for shipping, and offer promotional discounts. Once you understand the concepts in this chapter, you'll be well prepared to ask the right questions.

For store owners and administrators, this chapter will give you insights into what can be managed with Magento. Fortunately, there are few limitations to Magento. I am, however, consistently amazed at the various ways retailers price and vend their products. Hopefully, whatever unique selling programs you currently employ can be replicated online with your Magento store. I'm betting they can.

The sales process

If you've shopped online before, you no doubt have some understanding of the usual online sales process:

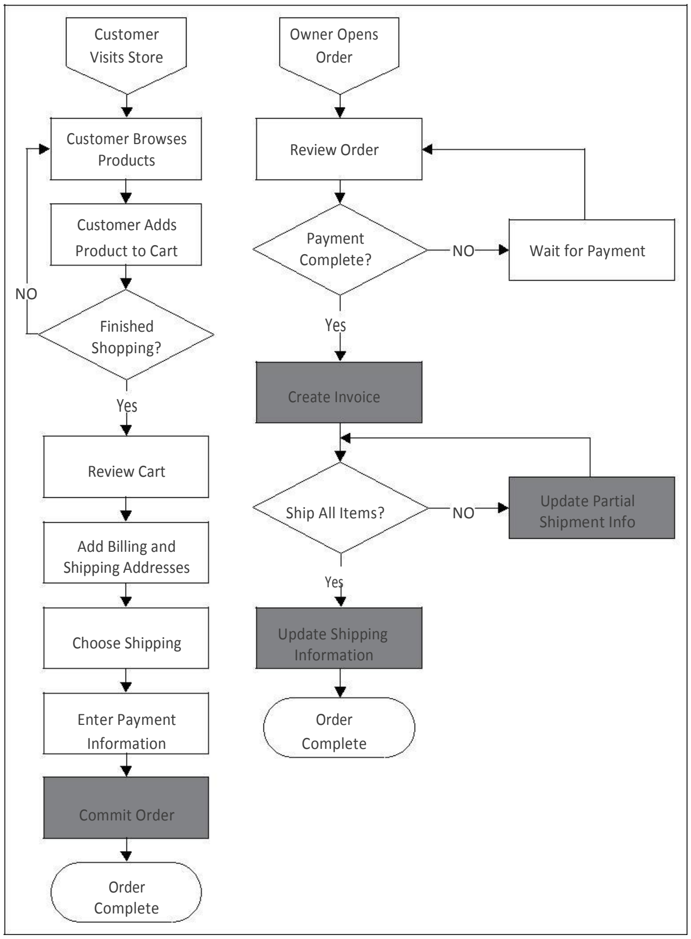
1. You browse and find a product that you want to purchase.
2. You "add" the product to your virtual shopping cart.
3. When you're finished shopping, you go to a *checkout* page.
4. In most online stores, you will first enter your billing and shipping addresses.
5. From this information, various shipping alternatives are presented, from which you choose the most appropriate for your needs and budget.
6. Next, you choose a payment method and enter your credit card information.
7. After reviewing your order details, you commit to the purchase, and moments later you receive confirmation that your order has been processed.
   You usually receive an e-mail receipt of your purchase.

8. After a day or so, you receive another e-mail announcing that your order has been shipped. This e-mail may also include package tracking numbers so that you can follow the progress of your package from "distribution to doorstep".

The Magento sales process

Magento duplicates this **sales process** in much the same way. For our purposes, though, we need to understand what happens after the customer commits to the order, for that is when the store administrator's participation is required.

The next chart illustrates both the **frontend** and **backend** steps of the Magento **sales process**:



The black squares with white type are steps that generally create an e-mail to the customer.

As we'll see in this chapter, there are occasional variations to this process, but in general, the Magento sales process is pretty straightforward. What you'll find impressive is the scope of Magento's ability to give you a wider latitude on adjusting the sales process to meet your particular needs. For instance, if you're selling downloadable products, such as e-books, music, or software, you don't need the shipping process steps. Yet, if you sell proprietary digital media, you will need to manage the distribution of the products to your customers. For instance, to prevent unlimited downloads.

Managing backend orders

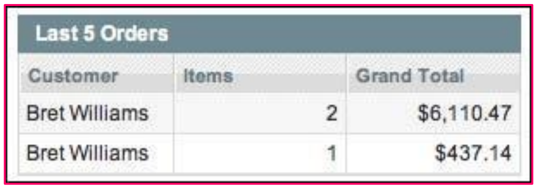
Before mapping out the business rules that you will use to configure Magento, it's helpful to see and understand how **orders** are processed in the Magento backend.

Many times, developers and administrators new to Magento rush to configure the myriad of settings (which we will be covering in this chapter) without fully realizing how those choices might impact the overall sales process. It's understandable because most will want to test the ordering system with all the settings in place. It's a bit of a catch-22: you have no orders to use to understand the configurations, yet without the configurations, you can't test the ordering process.

Fortunately, a basic Magento install with the sample data (again with the sample data? Yes!) gives you the basic configurations to allow you to place some sample orders and review the order process.

> **Give it a whirl!**
>
> If you've installed the sample data, or you already have a store configured to accept some type of test payments, you should spend some time placing and processing orders. Try any number of different combinations. Ask your colleagues to place dummy orders, imagining that they are actual shoppers. You'll soon get a real handle on the process, and if you're a developer, your client will certainly appreciate the added insight you have into the Magento ordering process. This is incredibly important to your client, so it should be important to you!

For this section, I have placed two sample orders to use for our journey through the **backend order process**. Logging into the Magento backend, we can see our latest orders listed in the left sidebar of the **Dashboard**:

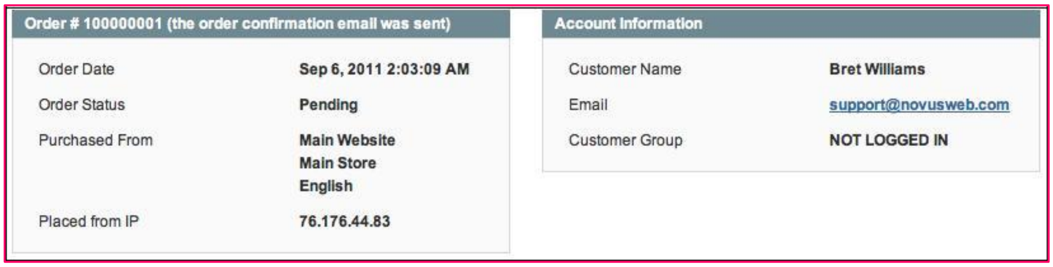


Last 5 Orders

Customer	Items	Grand Total
Bret Williams	2	$6,110.47
Bret Williams	1	$437.14

From here we can click on the order that we wish to process, or we can go to **Sales | Orders** in the top navigation bar and then select the order from the list of all orders. Either way, we end up with a detailed view of the order.

Let's take each section of this screen separately and explain what each contributes to the ordering process.



Order # 100000001 (the order confirmation email was sent)

Order Date	**Sep 6, 2011 2:03:09 AM**
Order Status	**Pending**
Purchased From	**Main Website** **Main Store** **English**
Placed from IP	**76.176.44.83**

Account Information

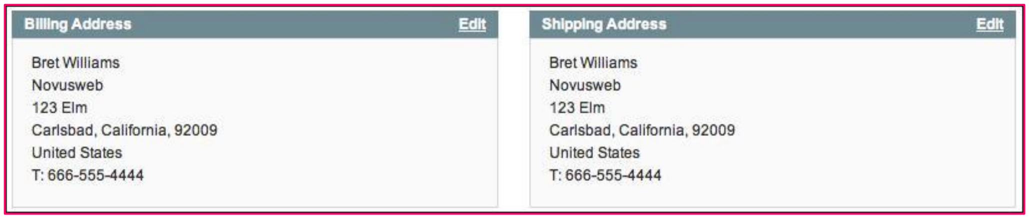
Customer Name	**Bret Williams**
Email	support@novusweb.com
Customer Group	**NOT LOGGED IN**

The first section, shown on the left-hand side, summarizes key order information, including the timestamp of the order (date and time), the current status (which by default is *pending*), from which Magento store the purchase was made, and the customer's IP. The IP number could be important if the customer makes a fraudulent purchase. Your merchant bank and law enforcement agencies can use this to track down suspected thieves.

New orders, by default, are marked as **Pending**. This means the order is awaiting your attention. The customer has already been charged and has received an e-mail confirmation of their order, but it's now up to you to complete the order, eventually taking it to a stage of *complete*.

The box on the right-hand side tells us the name of the customer, their e-mail address, and the fact that they checked out without registering (more on customer groups later in this chapter).

The next row of boxes shows the billing and shipping addresses of the purchaser. Notice that these are *editable*. Sometimes a customer, upon receiving their e-mail receipt, will see that they made an error in either one or both of these. If the customer contacts you with corrected information, you can easily make the changes here.



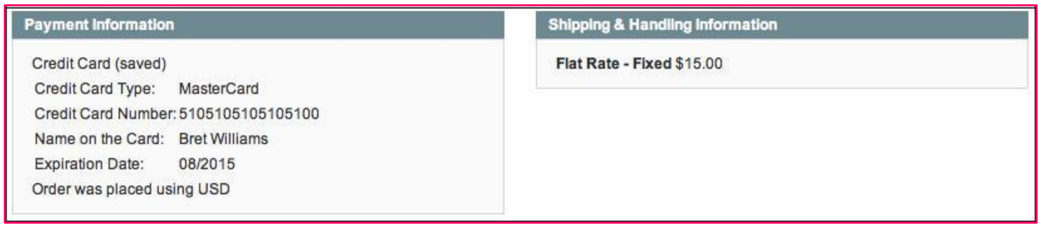
Billing Address Edit

Bret Williams
Novusweb
123 Elm
Carlsbad, California, 92009
United States
T: 666-555-4444

Shipping Address Edit

Bret Williams
Novusweb
123 Elm
Carlsbad, California, 92009
United States
T: 666-555-4444

The third row of boxes gives you information about the payment method and the buyer's choice of shipping for the order.



Payment Information

Credit Card (saved)

Credit Card Type:	MasterCard
Credit Card Number:	5105105105105100
Name on the Card:	Bret Williams
Expiration Date:	08/2015

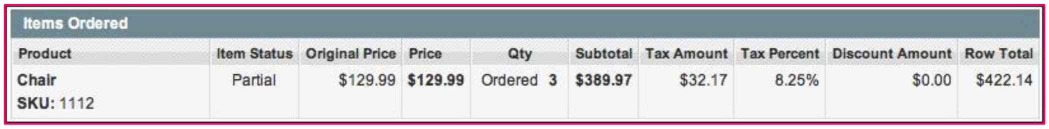
Order was placed using USD

Shipping & Handling Information

Flat Rate - Fixed $15.00

> Note that when using the sample data configuration, you are allowed to take and store credit card numbers in Magento. For **PCI Compliance**, this is a big no-no. Once you set up your payment gateways (later in this chapter), you won't see credit card numbers anywhere in Magento. And that's a good thing! You do not want to take on the liability of protecting your customer's credit card information from unscrupulous hackers and, sometimes, even employees.

On row four, you'll find the list of products ordered by the customer, the amount charged, and the amount of sales tax applicable for each line item.



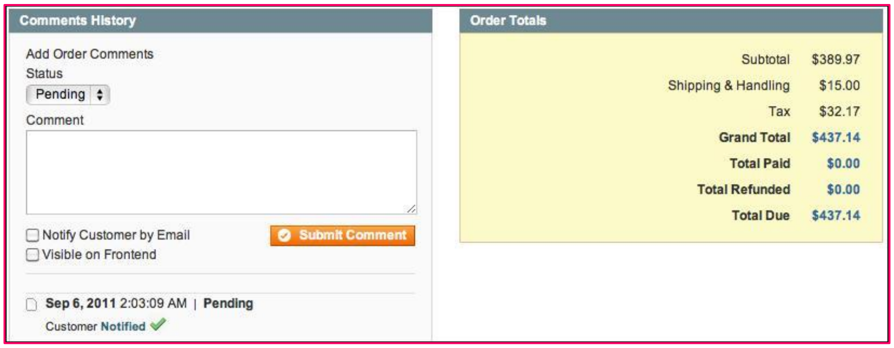
Items Ordered

Product	Item Status	Original Price	Price	Qty	Subtotal	Tax Amount	Tax Percent	Discount Amount	Row Total
Chair **SKU:** 1112	Partial	$129.99	**$129.99**	Ordered **3**	**$389.97**	$32.17	8.25%	$0.00	$422.14

In the previous screenshot, you'll see that the **Item Status** is **Partial**. In preparation for this example order, I reduced the available inventory for this item to "1" and allowed for backorders. While one of the missing pieces on this screen is the ability to see exactly what the inventory for a line item is, you are alerted that this product is not completely in stock.

The final row of boxes are quite interesting and important. First, on the left-hand side in the following screenshot, is how you can keep notes on an order and update the customer. Let's say, as in this case, that you have discovered that you only have one remaining chair and that more chairs won't arrive for another two weeks. By filling in this information here, and checking the **Notify Customer by E-mail** box, you can update the customer (*Would you like us to hold your order, or cancel it?*) and have the update sent automatically to the buyer. Furthermore, by checking the **Visible on Frontend** box, the customer—if they are registered (and this sample customer is not)—can view the update in the **Account Information** section of your store.

All comments entered will be appended to the list at the bottom of the box. The box on the right-hand side is the transaction summary of the order.



Comments History

Add Order Comments

Status

Pending

Comment

☐ Notify Customer by Email

☐ Visible on Frontend

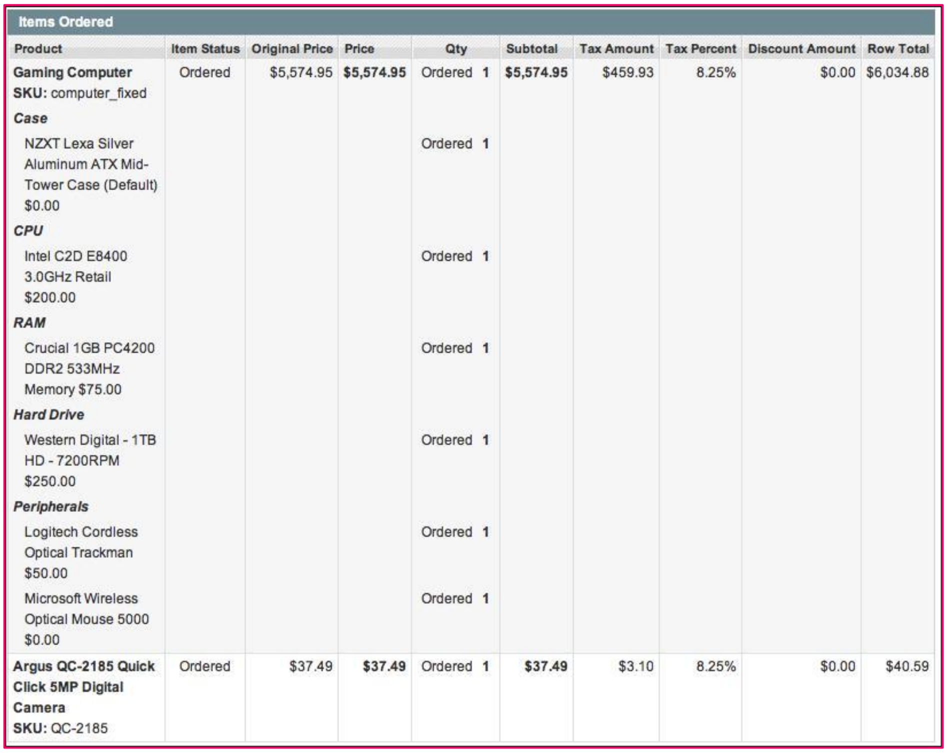
☐ **Sep 6, 2011** 2:03:09 AM | **Pending**

Customer **Notified** ✓

Order Totals

Subtotal	$389.97
Shipping & Handling	$15.00
Tax	$32.17
Grand Total	**$437.14**
Total Paid	**$0.00**
Total Refunded	**$0.00**
Total Due	**$437.14**

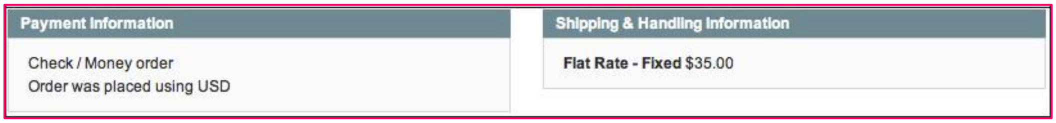
For the second sample order, there are two differences that I wanted to illustrate. The first was how a **configurable product** is represented. In the next screenshot, you can see that I ordered a **Gaming Computer** with additional choices for such things as **Case**, **RAM**, and **Hard Drive**:



Items Ordered

Product	Item Status	Original Price	Price	Qty	Subtotal	Tax Amount	Tax Percent	Discount Amount	Row Total
Gaming Computer **SKU:** computer_fixed	Ordered	$5,574.95	**$5,574.95**	Ordered **1**	**$5,574.95**	$459.93	8.25%	$0.00	$6,034.88
Case									
NZXT Lexa Silver Aluminum ATX Mid-Tower Case (Default) $0.00				Ordered **1**					
CPU									
Intel C2D E8400 3.0GHz Retail $200.00				Ordered **1**					
RAM									
Crucial 1GB PC4200 DDR2 533MHz Memory $75.00				Ordered **1**					
Hard Drive									
Western Digital - 1TB HD - 7200RPM $250.00				Ordered **1**					
Peripherals									
Logitech Cordless Optical Trackman $50.00				Ordered **1**					
Microsoft Wireless Optical Mouse 5000 $0.00				Ordered **1**					
Argus QC-2185 Quick Click 5MP Digital Camera **SKU:** QC-2185	Ordered	$37.49	**$37.49**	Ordered **1**	**$37.49**	$3.10	8.25%	$0.00	$40.59

The second difference is that the **Item Status** is **Ordered**, as I paid by check or money order, as shown below. This means that as a purchaser, I will be sending payment to the retailer before the product is shipped.



Payment Information

Check / Money order
Order was placed using USD

Shipping & Handling Information

Flat Rate - Fixed $35.00

Converting orders to invoices

The next step for you, as the person who is processing orders, is to **invoice** this order. In Magento, this means that you are confirming the order, and proceeding with processing.

> You can go straight to shipping for an order without having to generate an invoice. However, it is good practice to go from order to invoice to shipping. In this manner, you are tracking not only the products, but the payments, as well.

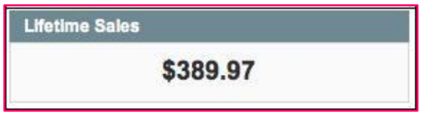
To convert an order into an invoice, click the button at the top of the page labelled **Invoice**. The resulting page is similar to the order page, except that it allows you to perform certain additional functions, as follows:

- **Create shipment**. In the box titled **Shipping Information**, you can check the box labelled **Create Shipment** and add any tracking numbers to the invoice.
- **Change the quantity of products to invoice**. As in our first example, if you have fewer products in stock than actually on hand, you may want to alter the number of products you are invoicing. Any remaining products will cause the order to remain open for future invoicing.
- **Add comments.** As on the order page, you can add a comment to the invoice and select whether the comment should be appended to the invoice. Otherwise, the comment will be added to the order history.

Once you have made any of the changes specified here, you can click on **Submit Invoice**, which will convert the order to an invoice, and record the order as an actual sale. This is key, as your analysis of sales for your store rely on the analysis of invoices, not orders. If you have not shipped the items yet, the status of the order is now shown as **Processing**.

Remember: *Orders are not sales*. Invoices are sales.

Once we have converted an order into an invoice, the box on the **Dashboard** screen titled **Lifetime Sales** is updated and shows the total sales for the chosen period, minus shipping and sales taxes, as shown in the following screenshot:



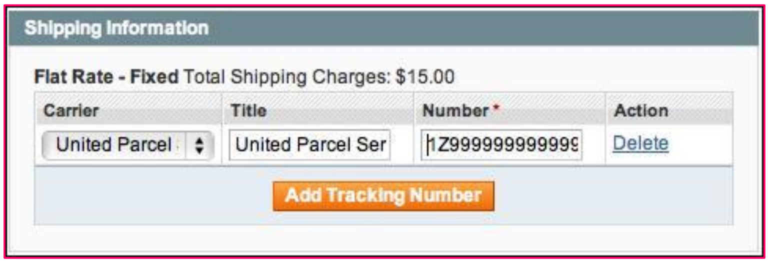
Lifetime Sales

$389.97

Creating shipments

Now that we have created our invoice, and once we have shipped the purchased products, we can create one or more shipments. To do this, open the order as before and click on the **Ship** button near the top of the screen.

On this screen, you can add **tracking numbers** for your shipments, as well as indicate the quantity of each product shipped. In the next screenshot, I have added a sample UPS tracking number. You can add as many as required (you may need to ship an order in more than one box, for instance).

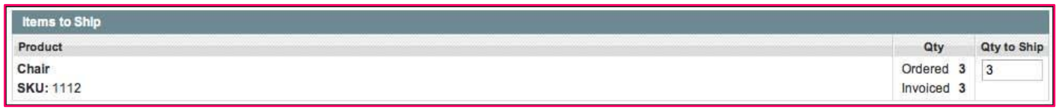


Shipping Information

Flat Rate - Fixed Total Shipping Charges: $15.00

Carrier	Title	Number*	Action
United Parcel	United Parcel Ser	1Z99999999999	Delete

Lower on the page, you will find each line item of the order with a field allowing you to change the number of products shipped, shown as follows:

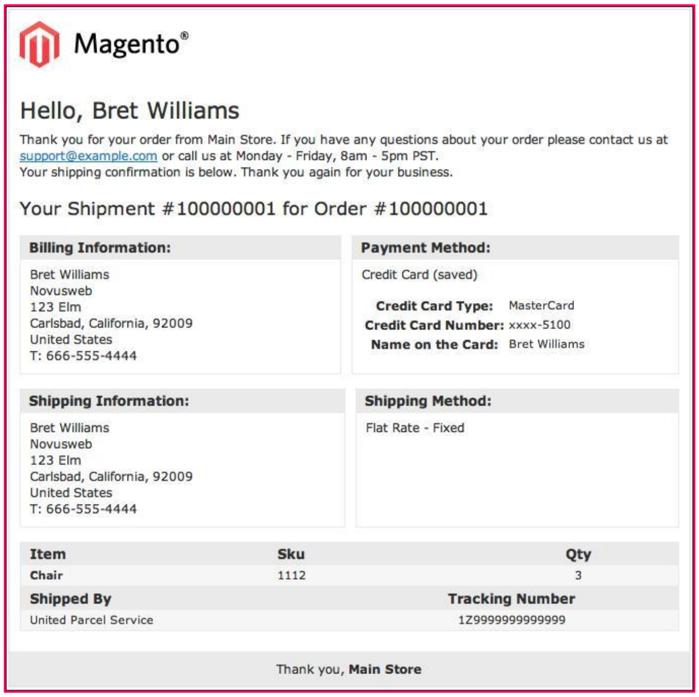


Items to Ship

Product	Qty	Qty to Ship
Chair SKU: 1112	Ordered 3 Invoiced 3	3

Once you have made any changes, including adding any comments, you can click on **Submit Shipment**. If you ship all ordered items, the status of the order will change to **Complete**; otherwise, the order will remain in a **Processing** state.

The following example is a typical e-mail sent to the buyer, showing that the order has been shipped, and includes its tracking information:



Hello, Bret Williams

Thank you for your order from Main Store. If you have any questions about your order please contact us at support@example.com or call us at Monday - Friday, 8am - 5pm PST.
Your shipping confirmation is below. Thank you again for your business.

Your Shipment #100000001 for Order #100000001

Billing Information:

Bret Williams
Novusweb
123 Elm
Carlsbad, California, 92009
United States
T: 666-555-4444

Payment Method:

Credit Card (saved)

Credit Card Type: MasterCard
Credit Card Number: xxxx-5100
Name on the Card: Bret Williams

Shipping Information:

Bret Williams
Novusweb
123 Elm
Carlsbad, California, 92009
United States
T: 666-555-4444

Shipping Method:

Flat Rate - Fixed

Item	Sku	Qty
Chair	1112	3

Shipped By	Tracking Number
United Parcel Service	1Z9999999999999

Thank you, **Main Store**

Once you become familiar with the sales process, you'll have a much better understanding of how various system configurations affect how orders are moved through Magento.

Payment methods

In today's online retailing world, most purchasers use credit cards (or debit cards) for payment. Nothing new there. However, the process—of taking someone's credit card details online, verifying the card for available purchasing limit, drawing the amount for the purchase from the buyer's account, and transferring it into your bank account—is one that remains a mystery to many. Of all the components that comprise online commerce, the process of moving money—in this case from the credit card account of the buyer to your bank account—remains one of the most complex of all.

Without the ease of credit cards, online e-commerce might well have been growing at a much slower pace. However, the use of credit cards—and the potential for misuse—concerns your shoppers, particularly when the press reports stories of hackers breaking into retailer databases. What is important is that online purchases have never been "hacked". That is, no one has been prosecuted for stealing credit card information used to buy online as long as the store is using SSL encryption. To ease consumers' fears, several payment systems have evolved over the past decade, each designed to help you process the financial transactions for your store, while providing increased security and processes necessary to give both you and your buyer a safer and easier transaction process.

As a Magento administrator, you have within Magento several default payment systems available based on your own needs. Each one requires that the store owner enroll and qualify, but, having done so, allows the store to provide buyers with a convenient, secure means of paying for their purchase.

In this section, we will cover the most common, popular payment systems and how they work with Magento. This is intended to familiarize you with how each system interacts with Magento—the buyer and the store owner. Once you understand how they work, you will be able to appropriately decide on which system(s) you want to employ, which also makes configuring Magento that much easier.

Classes of payment systems

The determination of which payment system to utilize in your Magento store is driven by a comparison of pros and cons (isn't everything in life?). In terms of credit card sales, there are two basic classifications of **payment systems**: *off-site* and *on-site*.

Off-site payment systems

Off-site systems allow buyers to make purchase choices, but pay for their order on another website which offers the buyer a sense of greater security and fraud protection. The buyer is actually paying the off-site payment provider, who in turn pays the store owner once there is sufficient verification that the order has been processed and shipped. Each system has different degrees of verification based on the type of products sold, the history of the merchant (that is, have there been previous problems with the merchant's reliability?), and the amount of the purchase.

The pros of using this system are as follows:

- Provides extra layers of protection to buyers against unscrupulous merchants.
- Quick merchant approval. No credit report is required.
- No PCI compliance requirements.

- Easy integration into almost any e-commerce platform.

Many buyers like these systems because of the added layer of protection against merchants who fail to deliver the expected results.

Additionally, the off-site system qualifies the merchant as opposed to a merchant account provider or bank. For first-time e-commerce merchants, this qualification is usually easier to obtain, as no credit report is required.

The downsides to these types of payment systems are as follows:

- Takes buyers off your e-commerce site
- May require the buyer to enroll in a third-party payment system
- The merchant has limited access to buyer information, including e-mail addresses

The dominant off-site systems are **PayPal Express**, **PayPal Standard**, and **Google Checkout**.

On-site payment systems

Almost any well-developed e-commerce store will allow buyers to pay directly on the site without having to go off-site to another payment system. While most will also provide off-site payment alternatives, by providing an on-site payment process, the merchant eliminates any reluctance the buyer may have to enroll in a third-party payment system.

The advantages of on-site payment systems are as follows:

- Keeps the buyer on the site, exposed to the merchant's own available merchandise
- Eliminates the need for a buyer to register or enroll with an outside payment system
- Gives the merchant access to all buyer information for follow-up, processing, and future marketing

In order to succeed with on-site payment systems, merchants need to consider design elements and payment system brands that will help buyers have confidence in the security of the payment process. Most buyers have no history with new merchants; therefore, merchants, if they wish to offer on-site payments, should pay special attention to methods of communicating the security of the buyer's information.

The cons of using on-site payment systems include the following:

- Requires a merchant banking account, which can be difficult to obtain for new businesses
- Site may be subject to PCI Compliance
- Integration with e-commerce platforms is more complex

On-site payments are processed through gateways. **Gateways** accept the customer payment information, as well as the order total, by means of a secure connection between your store server and the gateway's servers. The gateway validates the buyer's information and returns a result of success or error, which your store platform processes accordingly.

Currently, the most popular payment gateways include **Authorize.net**, **Moneybookers**, and **PayPal Pro**.

PayPal

In the early, heady days of e-commerce, many merchants lacked an easy way to implement credit card processing, particularly with the lack of sophisticated platforms such as Magento. To accommodate the growing legion of online buyers, and to provide a form of third-party protection for both transaction partners, **PayPal** was created. By signing up for a PayPal account, the buyer could pay for purchases at participating merchants, the money would be withdrawn from the buyer's bank account or credit card, and the merchant would be paid, less a processing fee. Both were protected against fraud by PayPal.

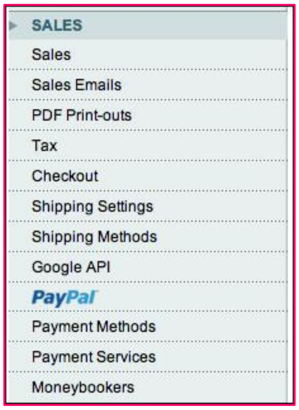
Today, PayPal remains one of the most popular payment systems in the world because it does allow for global purchases. You can sell to buyers in other countries, as long as they have a PayPal account, knowing that you will receive payment. Most regular merchant accounts, such as those used by brick and mortar retailers, restrict sales to only buyers with cards issued by US banks.

The downside to using PayPal used to be that buyers would have to sign up for PayPal if you, the merchant, offered it as a payment system. That changed some years ago: today your buyers don't have to sign up for PayPal. They can purchase using a credit card without enrolling. However, you as the merchant have to pay higher fees than a regular merchant account, and PayPal does everything it can to encourage enrollment, which may be a turnoff to some shoppers.

If you are considering PayPal as a payment system, the store owner must enroll with PayPal and choose one of three variations:

- **PayPal Express**: This is the simplest and fastest way to implement PayPal. In essence, your store's **Add to Cart** buttons become links directly to PayPal. Items are placed into a PayPal shopping cart, not yours, and the buyer must return to your store after each selection. It is the least integrated method, meaning that it does not provide your users with a consistent shopping experience.
- **PayPal Standard**: This is perhaps the best implementations of PayPal. It provides PayPal subscribers with the ease of buying with their PayPal account, and also allows others to use their own credit card without enrolling in PayPal; and it does not come into play until the shopper has completed all other steps on your checkout page. Instead of creating the shopping cart on PayPal, as with PayPal Express, the shopper creates the cart within Magento and then goes to PayPal only during the step of final payment. This gives the store owner greater access to information about their shoppers (for example, abandoned carts), and retains much more of the intended brand experience that you intended.
- **PayPal Pro**: Consider this an alternative to Authorize.net, which we will discuss a bit later. Shoppers enter credit card information right on your site, never being taken to PayPal. PayPal users cannot use their PayPal account, although you can use one of the other methods in conjunction with PayPal Pro. The downside is that the store owner must qualify in the same manner as if they were applying for a regular merchant account. Furthermore, your Magento installation is subject to PCI compliance scans.

PayPal is well integrated into Magento. It's no surprise, since eBay, the owner of PayPal, purchased Magento in 2011. In fact, it's quite evident that Magento would like store owners to consider using PayPal, as shown by the emphasis of PayPal in the system configuration sidebar menu, which you can see as follows:



Authorize.Net

Owned by CyberSource, **Authorize.Net** is the most popular payment gateway in the United States and Canada. In Magento, you have the opportunity to configure your store to use two different versions of Authorize.net's gateway:

- **Authorize.Net AIM**. Labelled simply Authorize.Net in the Payment Methods configuration section, AIM stands for Advanced Integration Method. AIM is an on-site payment system.
- **Authorize.Net Direct Post**. This is a fairly new service for Authorize. net. Direct Post works very similarly to PayPal Standard, in that once the customer is ready to complete their order, they are taken to the Authorize.Net website to enter their credit card information. Unlike PayPal, you still need a merchant account in order to use Direct Post.

Google Checkout

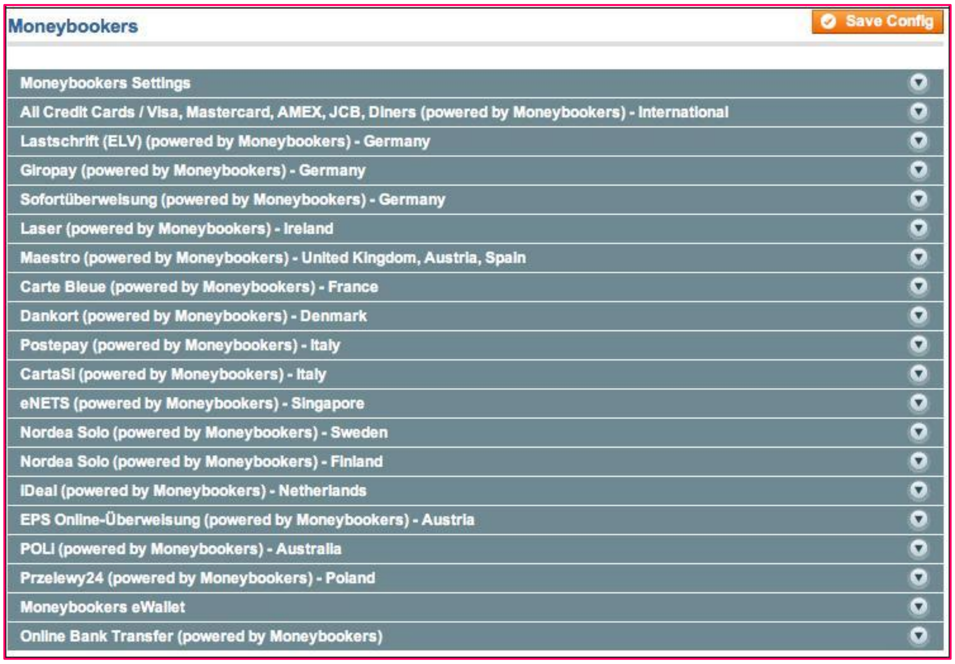
The behemoth that we call Google started offering payment services to merchants some years ago, perceived as a competitor to PayPal. Like PayPal, buyers can register with Google Checkout to store credit card and address information. As with PayPal, participating merchants allow buyers to use their Google accounts to buy products without having to disclose payment information to the merchant.

Google Checkout is an off-site payment system. Buyers who choose Google Checkout are taken to Google as the final step in the checkout process where they log in and verify their purchase.

The big difference between Google Checkout and PayPal Standard—the PayPal method most like Google Checkout—is how the merchant interacts with Google. Google places more responsibility on the merchant in terms of configurations for shipping and sales taxes, and in how they update Google when items ship or orders are canceled. Merchants do have to go through a slightly more complex verification process, as well.

MoneyBookers

For merchants selling internationally, and who want an on-site solution, Magento includes integration with MoneyBookers. **MoneyBookers** is not as well known in the US; but globally, it is the most popular payment processing gateway. The more impressive aspect of MoneyBookers is how complex the system must be to accommodate all the variations of their gateway based on location. As you can see from this next screenshot, there are different configurations required based on the countries in which the merchant wants to sell:



Moneybookers

Moneybookers Settings
All Credit Cards / Visa, Mastercard, AMEX, JCB, Diners (powered by Moneybookers) - International
Lastschrift (ELV) (powered by Moneybookers) - Germany
Giropay (powered by Moneybookers) - Germany
Sofortüberweisung (powered by Moneybookers) - Germany
Laser (powered by Moneybookers) - Ireland
Maestro (powered by Moneybookers) - United Kingdom, Austria, Spain
Carte Bleue (powered by Moneybookers) - France
Dankort (powered by Moneybookers) - Denmark
Postepay (powered by Moneybookers) - Italy
CartaSi (powered by Moneybookers) - Italy
eNETS (powered by Moneybookers) - Singapore
Nordea Solo (powered by Moneybookers) - Sweden
Nordea Solo (powered by Moneybookers) - Finland
iDeal (powered by Moneybookers) - Netherlands
EPS Online-Überweisung (powered by Moneybookers) - Austria
POLi (powered by Moneybookers) - Australia
Przelewy24 (powered by Moneybookers) - Poland
Moneybookers eWallet
Online Bank Transfer (powered by Moneybookers)

How Magento integrates payment systems

One of the more enjoyable aspects of Magento is that developers have gone to great lengths to integrate each of their default payment systems as fully as possible into Magento. That means that for each off-site system, where possible, the buyer is returned to your site after completing their off-site payment and the order information is sent back to your Magento store so that you can manage the order in the backend just as with any on-site purchase.

Additionally, Magento includes links within each payment system configuration to give you the information that you need to complete the configuration process.

The more important concern—and the purpose of this section—is to understand the various alternatives and consult with the store owner to determine which one(s) to configure. There are no right and wrong answers; it's a matter of merchant ability and preference, and customer expectations and needs.

Shipping methods

Magento includes, by default, integrations with the most popular shipping carriers: **UPS**, **USPS** (**United States Postal Service**), **FedEx**, and **DHL** (no longer shipping in the US, DHL is available in other countries). As with the payment methods, all but UPS require that you establish a customer account with each carrier in order to complete your configuration.

In addition to the common carriers, merchants can simply offer flat rate shipping costs or set up a table of rates based on weight, price, or number of items ordered. Merchants can also provide a site-wide free shipping condition, based on a minimum order amount.

Let's now explore some of the more complex configuration issues relating to shipping.

Common configuration choices

Every shipping method configuration shares some common configuration choices:

- **Enabled**: If you want the particular shipping method available at the Website level and below (that is, all Store Views), select **Yes**.
- **Title**: You may want to use a name that is more recognizable by your customers, or is translated for a different language Store View.
- **Ship to applicable countries**: You can choose to offer a particular method to all allowed countries, as defined under **System | Configuration | General | General**, or only to countries which you select in the field just below.
- **Sort order**: By entering numbers into these fields, you can control the order in which various enabled shipping methods are listed on the store checkout page.
- **Calculate handling fee**: If you wish to add an additional **handling fee** to the fixed shipping cost, you can do so by adding a fixed per order fee or one calculated as a percentage of the total product prices. Handling fees are not shown separately, but are added into whatever shipping costs are displayed to the buyer.

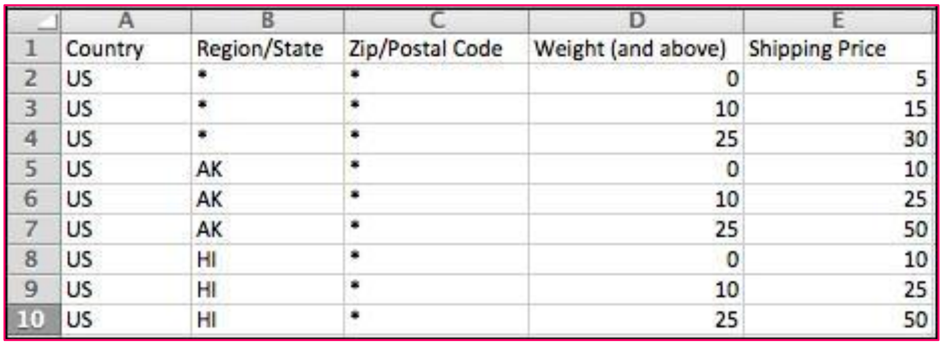
Flat rate shipping

This is perhaps the most straight-forward shipping configuration as it involves the fewest configuration choices. The more important ones is **Type**. You can charge a fixed shipping cost on a per-order or per-item basis. For instance, you might charge $5.00 for shipping any order of any size, or you might change $2.50 shipping on every item in the order.

Table rates

Setting up a rate table for shipping is by far the most complex of all shipping method configurations, as it requires that you prepare your shipping rate table separately, saving it as a CSV file, then importing it into Magento at the Website level.

The key to successfully implementing a rate table is to click on the **Export CSV** button (you must be at a Website level in the **Current Configuration Scope**). Open the downloaded file in a spreadsheet program or text editor and enter in your desired values according the columns shown in the file. Given next is a portion of a spreadsheet with rates entered according to the column names at the top:



	A	B	C	D	E
1	Country	Region/State	Zip/Postal Code	Weight (and above)	Shipping Price
2	US	*	*	0	5
3	US	*	*	10	15
4	US	*	*	25	30
5	US	AK	*	0	10
6	US	AK	*	10	25
7	US	AK	*	25	50
8	US	HI	*	0	10
9	US	HI	*	10	25
10	US	HI	*	25	50

Let's go over each column and explain what should be entered in each:

- **Country**: Enter the two-letter abbreviation of the country for which the rate applies. An asterisk (*) means that the rate applies to all allowed countries.
- **Region/State**: Use the two-letter code for the state or region of the country. In the case of US states, an asterisk means the rate applies to the "lower 48", the states except for Alaska and Hawaii. Since it commonly costs more to ship to these states, Magento wisely makes a provision for setting rates only for the continental US without having to list each state separately. If you intend to ship to Alaska and Hawaii, you will need to include line items for each, as shown in the table.

- **Zip/Postal Code**: Usually, you would include all codes within a given state or region. However, you can be quite granular by entering specific postal codes in this column.
- **Weight (and above)**: Table rates work by taking the total weight of the products in the order. Use this column to provide different shipping rates based on this weight total.
- **Shipping Price**: This is the rate that will be applied to the order.

The previous chart shows the columns for *weight-based* calculations. For *price-based* or *quantity-based* calculations, the Weight column is named differently to reflect these alternatives. You can choose either of these alternatives using the drop-down menu called **Condition** in the **Table Rates** panel.

Once you have your table rates set, use the **Import** configuration function to upload your CSV file to Magento.

Free shipping

All the common carrier methods allow you to set one method (for example, ground, overnight) as a free method based on a minimum order amount. The **Free Shipping** method gives you the global ability to offer free shipping based on a minimum order amount, as well. The difference is that with the **Free Shipping** method, you cannot specify or display a particular method such as ground, overnight, and so on.

Generally, you would use this method in conjunction with the **Flat Rate** or **Table Rate** methods. If you're now asking, "why not just specify no shipping cost in your table rate CSV file?", here's one scenario where you would still find value in using Free Shipping with a Table Rate configuration.

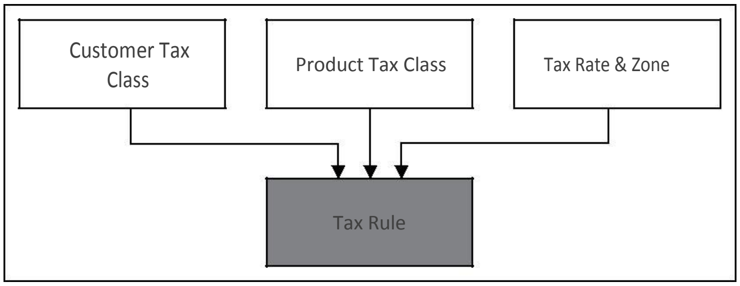
Let's say you've set up your Table Rates to charge based on weight. The heavier the order, the more the shipping costs, much like in our earlier example. However, for customers who order a total of $500 or more, you'd like to give them free shipping.
Since your Table Rates are based on weight, the **Free Shipping** method gives you the opportunity to offer free shipping once your profit margin is sufficient.

Configuring sales tax rules

As you know, when selling products to consumers, you may need to charge and collect sales tax. Some states don't require sales tax, but most due. Moreover, the rules for calculating and collecting sales tax varies from state to state, country to country.

Before configuring your Sales Tax Rules, you may need to consult with your accountant. The examples in this book are purposely simple; your actual sales tax configuration may be more complex depending on your situation.

Before we dive into creating Sales Tax Rules, let's understand a bit about the relationship Magento creates between Sales Tax Rules, Customer Tax Classes, Product Tax Classes, and Tax Rates & Zones. The following chart illustrates this relationship:



The tax rule that calculates sales tax on any particular item is based on a combination of the type of customer, the type of product and the applicable tax rate based on the geographic location of the shipping destination.

Creating a sales tax rate

As an example, let's assume you're located in Massachusetts. Since your physical location is in the Bay State, you're obligated to charge customers a 6.25% tax on all applicable purchases (again, *please* consult your accountant for advice; the tax rates used in this example are for illustrative purposes only).

Therefore, to create the appropriate Sales Tax Rate:

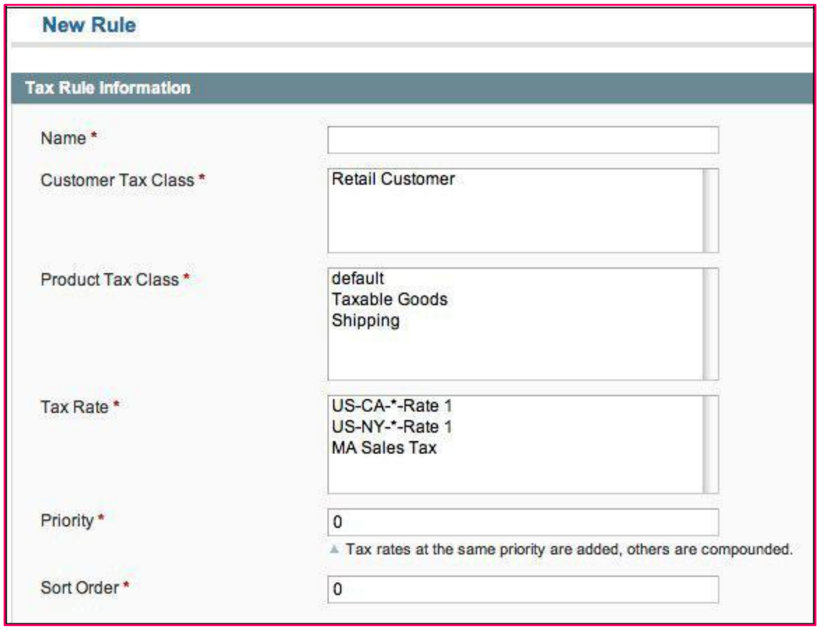
1. Go to **Sales | Tax | Manage Tax Zones & Rates** in your Administrative backend.
2. Click on **Add New Tax Rate** near the top-right of your screen.
3. For **Tax Identifier**, enter a value that is descriptive of the rule. In this case, we'll use **MA Sales Tax.**
4. Of course, choose **Massachusetts** as the **State** in the drop-down menu.
5. Since your tax will apply to all zip codes, leave **Zip/Post is Range** set to **No**.
6. Enter **6.25** as the **Rate Percent**.

7. Finally, if you wish to use a name other than the **Tax Identifier** for any of your **Store Views**, enter it in the **Tax Titles** section.
8. Click on **Save Rate** to save your changes.

Creating the sales tax rule

Creating the Sales Tax Rate doesn't yet mean that purchases by customers in Massachusetts will have a tax levied. Referring to the previous diagram, you now must combine the **Tax Rate** with a **Customer Tax Class** and a **Product Tax Class**.

1. Go to **Sales | Tax | Manage Tax Rules**.
2. Click on **Add New Tax Rule**.
3. On this screen, as shown in the next screenshot, you combine the various components that together create the **Sales Tax Rule**:



New Rule

Tax Rule Information

Name *	
Customer Tax Class *	Retail Customer
Product Tax Class *	default Taxable Goods Shipping
Tax Rate *	US-CA-*-Rate 1 US-NY-*-Rate 1 MA Sales Tax
Priority *	0 Tax rates at the same priority are added, others are compounded.
Sort Order *	0

4. Enter a descriptive name in the **Name** field, such as *MA-Retail-Taxable*.
5. To continue our example, click and select **Retail Customer**, **Taxable Goods**, and **MA Sales Tax** in the fields shown.
6. Click on **Save Rule**.

Now, whenever a retail customer purchases a taxable product to be shipped to a Massachusetts address, 6.25% sales tax will be added to the price of the product.

> **Charging tax on shipping**
>
> In most cases, you will not be charging sales tax for shipping costs. However, if you are required to do so, you would designate that requirement in the **Shipping Methods** configurations.

Outgoing e-mails

In past years, one of the banes of using Magento was customizing outgoing e-mails sent to your customers for orders, invoices, site registrations, and so on. It was truly a painful experience.

With Magento today, administrators have it much, much easier. You can quickly and easily create multiple versions of e-mails with different styles and content. Let's go through the process of creating an alternative version of the order confirmation e-mail and assigning it to one of our Stores.

1. Go to **System | Transactional E-mails** in your Magento backend. Although you'll see an empty list, this screen only shows customized e-mails once they are created.
2. Click on **Add New Template**.
3. The first step in the process is to select an existing Magento e-mail as a template for your new e-mail. Let's select **New Order** and click on **Load Template**.
4. You'll see that **Used as Default For** shows you where in the system configuration this particular e-mail is assigned. In this case as the **New Order Confirmation Template** for all Websites and Store Views, although any Website or Store View can override this by assigning a new e-mail template as we will do in this example.
5. Fill in an applicable name for your new e-mail in the **Template Name** field. We'll use *Electronics New Order*, since we're going to use this new e-mail for our Electronics Store.
6. In most cases, and until you become more familiar with Magento variables, you're not likely to need to change the **Template Subject** field. That said, you may want to include something like *Thank You* before the shown content.

> There are several variables that you can use in an e-mail template. If you find one which you wish to use in the subject, add it to the **Template Content** field, then cut and paste it into the **Template Subject** field.

Before we continue with the e-mail editing process, let's talk about **designing e-mails**. E-mails which are constructed poorly or improperly will most often times end up in the recipients spam filter or, at best, be unreadable by their e-mail program. If you're not familiar with designing e-mails that get past spam filters and look good to recipients, go online and look for recent articles regarding HTML e-mail design. While this is a topic which deserves much more space and attention than what we can provide here, in general HTML e-mails should be table-based. I know, everyone wants table-less web pages, but e-mail's different. CSS styling should be inline; never as a separate file or stuffed into the header area. You should also pay attention to width, as most e-mail readers—especially those on smart phones—have limited viewing widths.

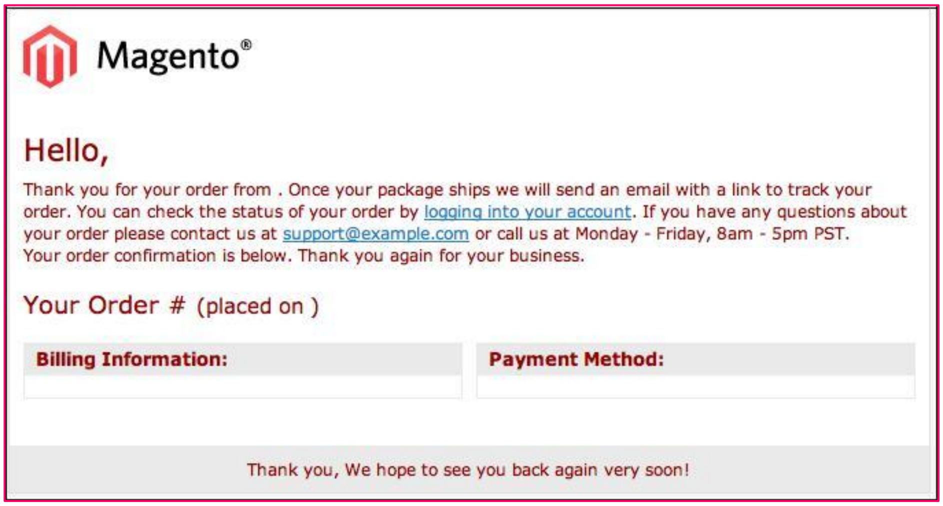
For now, use the Magento templates as a guide and if you know your way around HTML, you'll easily find ways of modifying the template to your liking without breaking anything. For our example:

1. Scroll to the bottom of the **Template Content** field and find the HTML that contains **Thank you**. Add, before the </p> tag, *(space)We hope to see you back again very soon!* This line should look something like this:

```
<td    bgcolour="#EAEAEA"
        align="center"
        style="background:#EAEAEA; text-align:center;"> <center><p
    style="font-size:12px; margin:0;">
        Thank you, <strong>{{var store.getFrontendName()}}</strong> We hope to see you back
        again very soon!</p>
    </center>
</td>
```

2. The **Template Styles** field contains the only CSS styling not set as an inline style. Here you can change the overall, default values for the body and other elements. For this example, we're going to change our body colour value to #990000 just to illustrate this change. This will set the text colour of the e-mail to red.

3. Click on **Preview Template** to view your changes. In this example, our changes are shown in the following screenshot:



Magento®

Hello,

Thank you for your order from . Once your package ships we will send an email with a link to track your order. You can check the status of your order by logging into your account. If you have any questions about your order please contact us at support@example.com or call us at Monday - Friday, 8am - 5pm PST. Your order confirmation is below. Thank you again for your business.

Your Order # (placed on)

Billing Information:	Payment Method:

Thank you, We hope to see you back again very soon!

While the change in text colour may not be evident, you can see that the bottom thank you line has changed. What is not shown in this preview are some of the dynamic values, such as the name of the Store, which is supposed to appear right after **Thank you**.

4. Click on **Save Template** to save your work.

While it may seem a bit tedious, if you need distinct differences in the e-mails sent by your different **Stores**, Magento does give you tremendous flexibility to craft your e-mails to suit your brand and purpose.

Now that we have created a unique e-mail for our Electronics Store, we need to assign it for use in that store.

1. Go to **System | Configuration | Sales | Sales E-mails**.
2. Change your **Current Configuration Scope** to **Electronics/English**.
3. Open the center panel titled **Order**.
4. De-select the checkbox to the right-hand side of **New Order Confirmation Template**. This will activate the drop-down for you to change.
5. Select **Electronics New Order** from the drop-down menu.
6. To save your new configuration, click on **Save Config**.

From now on, when a customer orders from the Electronics Store, they will receive the new, customized e-mail we created in this example.

Changing address formats

Customer addresses, as you can imagine, are formatted differently in different countries of the world. Fortunately, you can set the format of addresses for a given Store View by going to **System | Configuration | Customer | Address Templates**. The format will be the same for all customers in the given store, but if your Store Views are language/ region specific, this can help you provide proper formatting for your customer e-mails.

Summary

Magento is, above all things, a selling tool, a means of presenting products, taking orders and payments, and managing the fulfillment of those orders. Even though there are common steps to ordering online, Magento provides one of the most comprehensive and versatile ordering mechanisms to consumers among open source platforms. If nothing else, this chapter has demonstrated the impressive scope of its functionality.

As you begin to move closer to launching your new Magento store, you will re-visit the various configurations and processes relating to the sales cycle. Don't be discouraged by this. In fact, I find that as I discover more of Magento's power, I am empowered to create more ways of promoting and selling to customers.

In this chapter, we:

- Learned about the Magento sales process
- Went through a typical order management process
- Examined the various payment systems available in a default Magento installation
- Explored shipping method configurations
- Created a sales tax rule
- Customized an outgoing e-mail

You should also keep in mind that there are over 4,000 extensions to Magento, many of which add additional functionality for all the areas we covered in this chapter.
If Magento doesn't exactly provide what you need, chances are some third-party developer has created an add-on that meets your needs.

The next item on our agenda is to manage the various non-product content elements of a Magento store, which we'll cover in the next chapter.

3. Managing Non-Product Content

As important as products and product information are to e-commerce, successful online stores need more in order to attract customers and fortify the store's brand. Even printed catalogs often contain information about the seller, including hours of operation, return policies, company history, and more. This **non-product content** is essential.

To gain an understanding of how to create and manipulate non-product content, we will:

- Review how Magento incorporates content
- Learn how to create content pages in your Magento store
- Create and use static blocks
- Utilize built-in content widgets
- Review the principles of layout customization

Once you know how to interweave non-product content throughout your online store, you will no doubt discover innovative ways to increase customer engagement.

The Magento content management system

As with most e-commerce platforms, Magento's management of **non-product content** lacks some of the more robust features of a dedicated **Content Management System**—or **CMS**—such as *WordPress*, *Business Catalyst*, or *Joomla*. However, to its credit, Magento does provide a versatile system that takes into account the possible need for unique content for each of your stores.

As you would expect, the backend area for the management of CMS functions in Magento is under the **CMS** menu. This area includes sections for managing pages, static blocks, widgets, and polls. We will cover the first three in this chapter, reserving polls for *Chapter 7*, *Marketing Tools*.

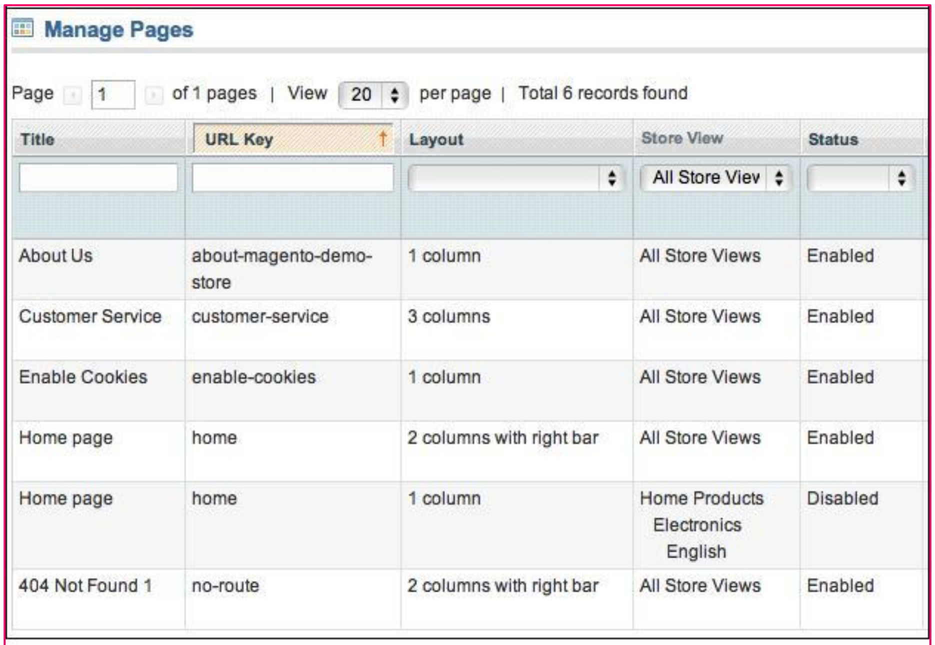
Before we dig into each of these items, it's important to recall our discussion about blocks in *Chapter 4*, *Designs and Themes*. Whether configuring a page, or placing a static block or widget, Magento builds the final results by assembling blocks of information.

Pages

The pages that we will use in the CMS are not actually complete pages as they lack controls for the overall template items of the header, navigation, and footer. These pages actually refer to the central content of a page—that which lies within the overall page template. Within this page, we can add text, images, static blocks, and widgets to give the page it's core content. As we will examine in this section, you can add some of the same code as you did in the local.xml file to even manipulate elements outside the core content area, including blocks within the header, navigation, and footer.

To begin, let's look at and alter a default page provided by the sample data installed into a new Magento store.

Go to **CMS | Pages** in the Magento backend. With the sample data installed, you should see a list of pages, much like those shown in the following screenshot:



Manage Pages

Page 1 of 1 pages | View 20 per page | Total 6 records found

Title	URL Key	Layout	Store View	Status
			All Store Viev	
About Us	about-magento-demo-store	1 column	All Store Views	Enabled
Customer Service	customer-service	3 columns	All Store Views	Enabled
Enable Cookies	enable-cookies	1 column	All Store Views	Enabled
Home page	home	2 columns with right bar	All Store Views	Enabled
Home page	home	1 column	Home Products Electronics English	Disabled
404 Not Found 1	no-route	2 columns with right bar	All Store Views	Enabled

> There are two pages on this list that you should always keep available: **Enable Cookies** and **404 Not Found 1**. These pages are used by Magento to display error messages to visitors if they do not have cookies enabled in their browser or they reach a page that does not exist, respectively. You can always modify the content of these pages to suit your need, or even duplicate and modify them for different stores, but each store should have both of these pages enabled and assigned.

Customizing a CMS page

The first place that most designers want to begin modifying a store's design is the home page. The sample data installation installs two versions of the home page, one enabled for all stores, the other disabled and assigned to only one store. Both are created based on the base package layout and design.

As we discussed in *Chapter 4*, *Designs and Themes*, we don't ever want to modify the base package. If we really liked the design, we could duplicate the **layout** and **skin** files, place them within another package, such as the default package, and modify them there. However, 99.9% of the time you understandably don't want your new store looking like the Magento default.

So, let's begin our quest by assigning one of the other default themes installed in Magento and modify the resulting home page according to our tastes. By following this example process, you'll learn a lot about how page modifications work.

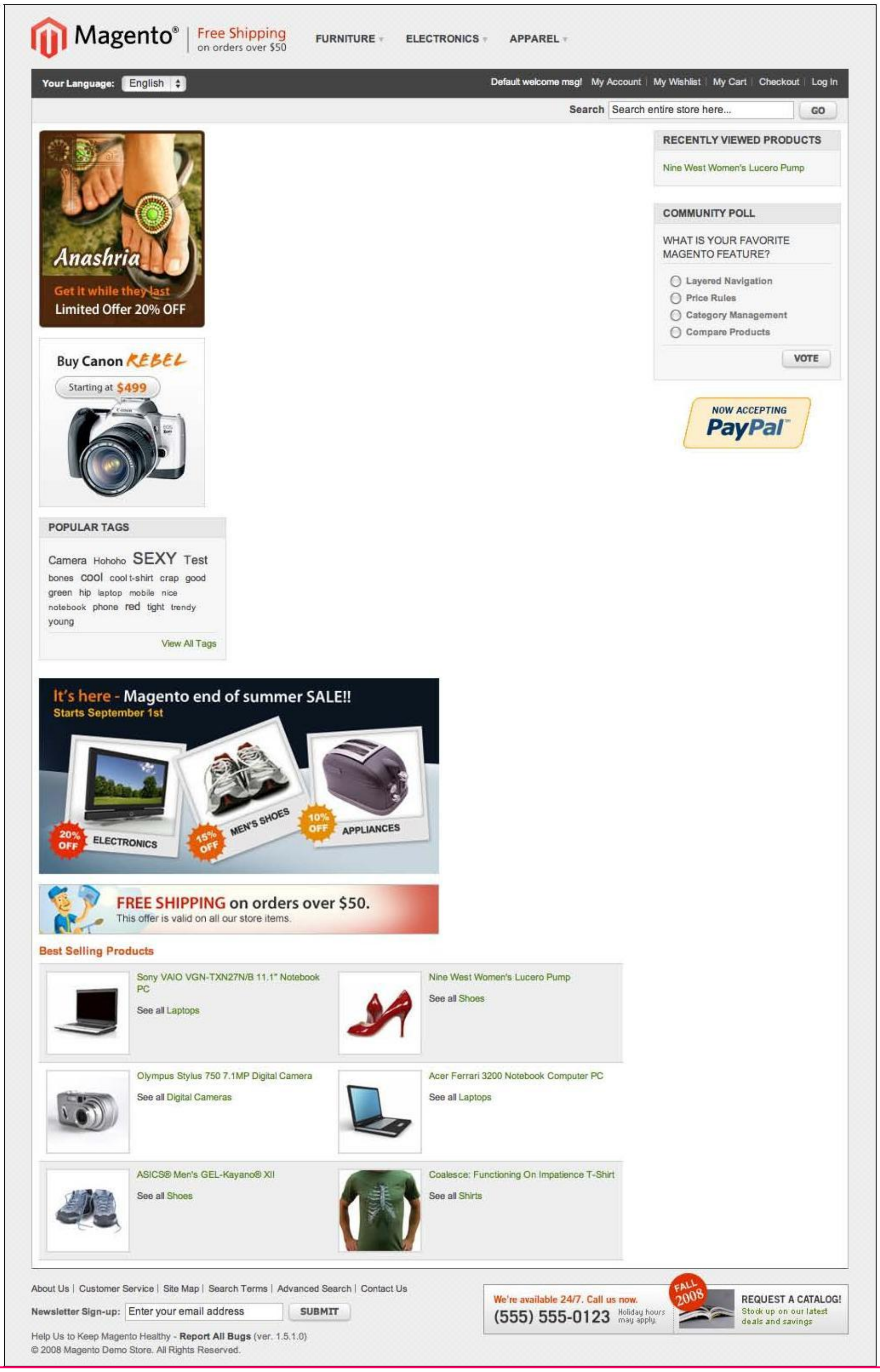
> Before doing any design modifications on a site, it's a good idea to turn off the caches. Go to **System | Cache Management** and select all the various cache types, and disable them. This way you won't have to continually refresh the cache after each change. Once you're finished updating your design, you can return and enable the caches again. If you have enabled compilation on your site, you should also disable it.

Assigning a theme

First, let's set our store to use the *modern theme*. It's a nice, simple store design that's not bad looking and easier on the eyes.

1. Go to **System | Configuration | General | Design** in the Magento backend.
2. For now, our changes will effect all the stores, so leave your **Current Configuration Scope** set to *Default*.
3. In the center panel, enter the following attributes:
    - **Templates**: *modern*
    - **Skin (Images / CSS)**: *modern*
    - **Layout**: *modern*
4. Click on **Save Config**.

If we now take a look at our home page, we should see something like the following screenshot :



Search entire store here...

POPULAR TAGS

Camera Hohoho SEXY Test bones cool cool t-shirt crap good green hip laptop mobile nice notebook phone red tight trendy young

View All Tags

RECENTLY VIEWED PRODUCTS

Nine West Women's Lucero Pump

COMMUNITY POLL

WHAT IS YOUR FAVORITE MAGENTO FEATURE?

○ Layered Navigation
○ Price Rules
○ Category Management
○ Compare Products

VOTE

Best Selling Products

Sony VAIO VGN-TXN27N/B 11.1" Notebook PC See all Laptops	Nine West Women's Lucero Pump See all Shoes
Olympus Stylus 750 7.1MP Digital Camera See all Digital Cameras	Acer Ferrari 3200 Notebook Computer PC See all Laptops
ASICS® Men's GEL-Kayano® XII See all Shoes	Coalesce: Functioning On Impatience T-Shirt See all Shirts

About Us | Customer Service | Site Map | Search Terms | Advanced Search | Contact Us

Newsletter Sign-up: Enter your email address

Help Us to Keep Magento Healthy - **Report All Bugs** (ver. 1.5.1.0)

© 2008 Magento Demo Store. All Rights Reserved.

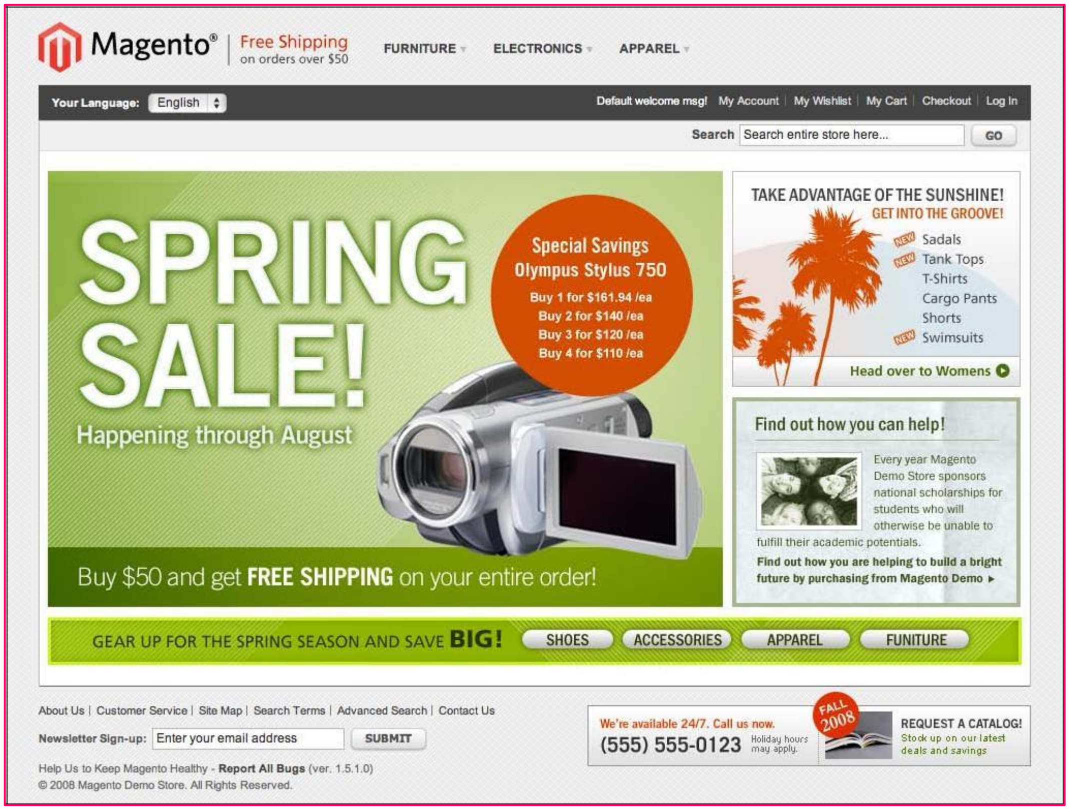
Kind of messy, eh? Basically, it's because the home page is configured in the CMS to display certain elements that have not been styled or appropriately sized for this theme. Remember, the Magento **Fallback Method** says that if an element does not exist within the designated theme, Magento looks to the **default theme** of the **design package**, then into the **Base Package**. Many of the components that make up this page are not part of the *modern theme* and exist only in the **Base Package**.

Modifying the home page layout

Fortunately for us, the sample data installation provides an alternative home page that works very well with the modern theme:

1. Go to **CMS | Pages**.
2. Click on the first **home page** listed (with the **2 columns with right bar** layout).
3. Change the **Status** to *Disabled* and click on **Save**.
4. Now, click on the second **home page** listed (with the **1 column** layout).
5. Select **English**, **French**, and **German** as the store views under **Main Website** in the **Store View** list.
6. Change the **Status** to *Enabled* and click on **Save**.

Now look at your home page:


Magento®
Free Shipping on orders over $50
FURNITURE
ELECTRONICS
APPAREL
Your Language: English
Default welcome msg!
My Account | My Wishlist | My Cart | Checkout | Log In
Search
Search entire store here...
GO
SPRING SALE!
Happening through August
Special Savings
Olympus Stylus 750
Buy 1 for $161.94 /ea
Buy 2 for $140 /ea
Buy 3 for $120 /ea
Buy 4 for $110 /ea
Buy $50 and get FREE SHIPPING on your entire order!
TAKE ADVANTAGE OF THE SUNSHINE!
GET INTO THE GROOVE!
NEW Sadals
NEW Tank Tops
T-Shirts
Cargo Pants
Shorts
NEW Swimsuits
Head over to Womens
Find out how you can help!
GEAR UP FOR THE SPRING SEASON AND SAVE BIG!
SHOES
ACCESSORIES
APPAREL
FUNITURE
About Us | Customer Service | Site Map | Search Terms | Advanced Search | Contact Us
Newsletter Sign-up: Enter your email address
SUBMIT
We're available 24/7. Call us now.
(555) 555-0123
Holiday hours may apply.
FALL 2008
REQUEST A CATALOG!
Stock up on our latest deals and savings

Every year Magento Demo Store sponsors national scholarships for students who will otherwise be unable to fulfill their academic potentials.

Find out how you are helping to build a bright future by purchasing from Magento Demo ▸

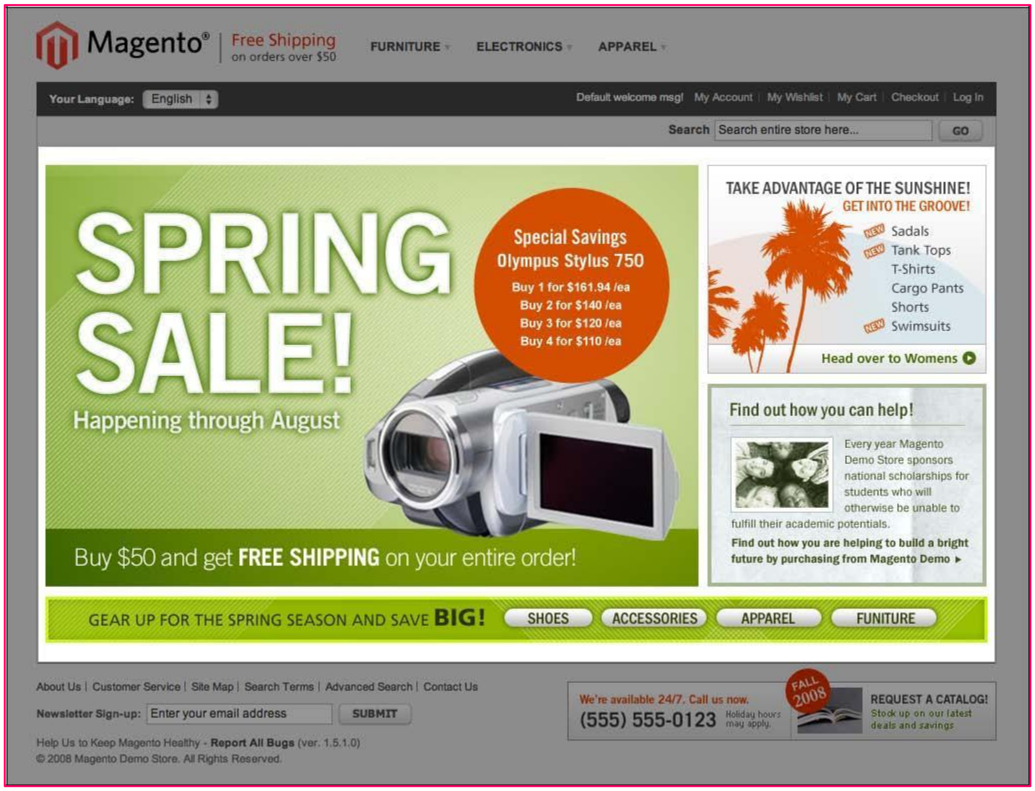
Help Us to Keep Magento Healthy - **Report All Bugs** (ver. 1.5.1.0)

© 2008 Magento Demo Store. All Rights Reserved.

Much better. However, it does lack some of the dynamic elements which we might want on this page, such as a list of product categories, recently viewed products, and a summary of a visitor's shopping cart.

Return to your Magento backend, and if you're not there, go to **CMS | Pages**. Click on the active **home page** item and let's take a closer look at what dictates what is on this page.

First, we need to understand that pages created in the Magento CMS primarily dictate what appears in the content section of the page. This is the area highlighted in the next screenshot, apart from the header, top navigation, and footer regions:



Every year Magento Demo Store sponsors national scholarships for students who will otherwise be unable to fulfill their academic potentials.

Find out how you are helping to build a bright future by purchasing from Magento Demo ▸

Help Us to Keep Magento Healthy - **Report All Bugs** (ver. 1.5.1.0)

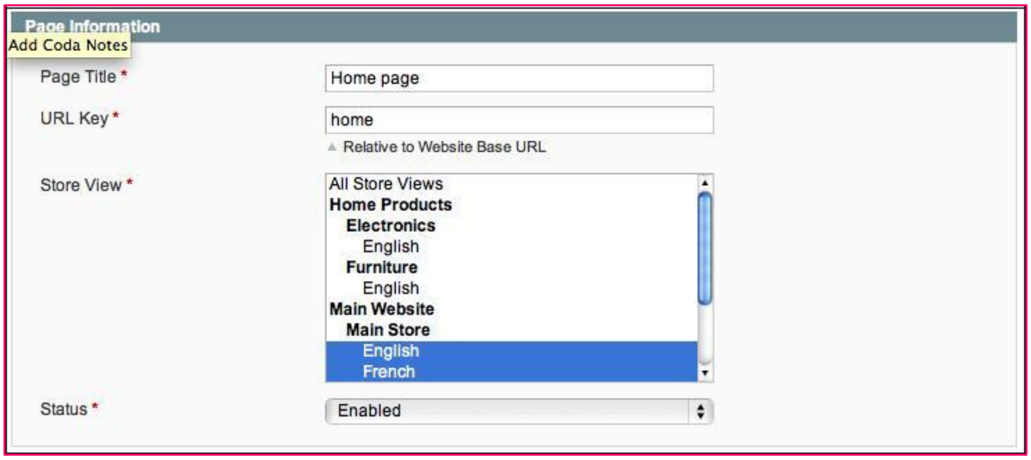
© 2008 Magento Demo Store. All Rights Reserved.

As we'll see however, we can add XML statements to a page's design to manipulate other non-content areas of the page.

Remember the local.xml file

In *Chapter 4*, *Designs and Themes*, we introduced a new feature of Magento: the local.xml file. By using various XML handles, you can manipulate an entire template for all pages. The same principles apply when working with pages, although your changes here will only affect the edited page.

On the **Page Information** panel of the **Edit Page** screen, you should see the following menu:



Page Information

Add Coda Notes

Page Title * Home page

URL Key * home

▲ Relative to Website Base URL

Store View *

All Store Views
Home Products
Electronics
English
Furniture
English
Main Website
Main Store
English
French

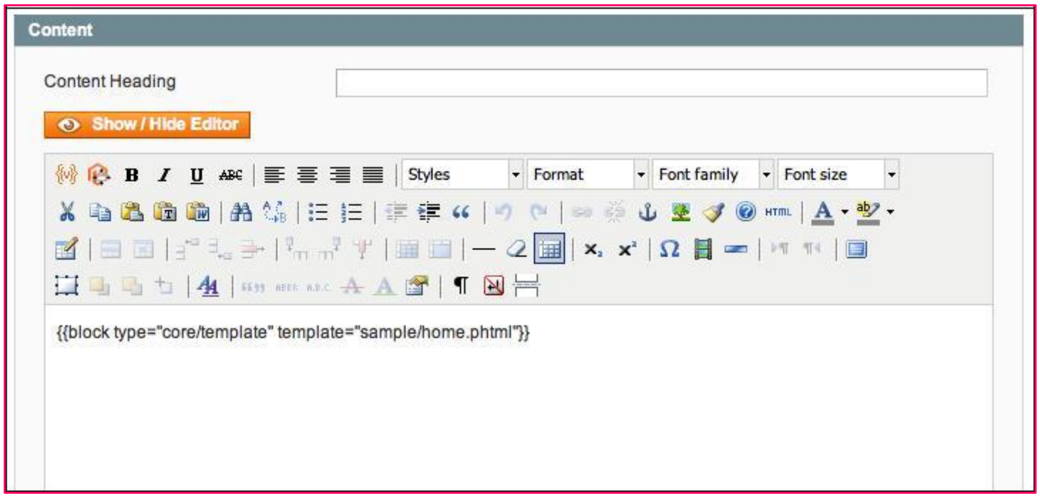
Status * Enabled

- **Page Title** is the title that will appear in the top of your browser, as well as the default name of any link you create to this page.
- **URL Key** is how the page is accessed in a URL path. For example, this page is accessible by going to http://www.storedomain.com/home.html. Magento assumes that your home page will have a URL Key of *home*. You should not change this unless you know what you're doing.
- **Store View** allows you to select which stores your page is applicable for. When you save your page, Magento does check to make sure another page with the same URL Key has not already been assigned to the same stores.
- **Status**, of course, indicates whether the page is active or not.

The Content screen

In the left-hand side tab, click on **Content**. This is the screen where you can add text, images, or dynamic content that will appear in the central content area of your page. You're also free to use HTML code, if you wish.

If you notice in the next screenshot, this sample data home page contains a **block reference**:



{{block type="core/template" template="sample/home.phtml"}}

Just as we use blocks in the XML files that define our page layouts, we can also use them in this content area. However, the format is a bit different: instead of using XML, we use "handle bars" to denote dynamic code:

```
{{block type="core/template" template="sample/home.phtml"}}
```

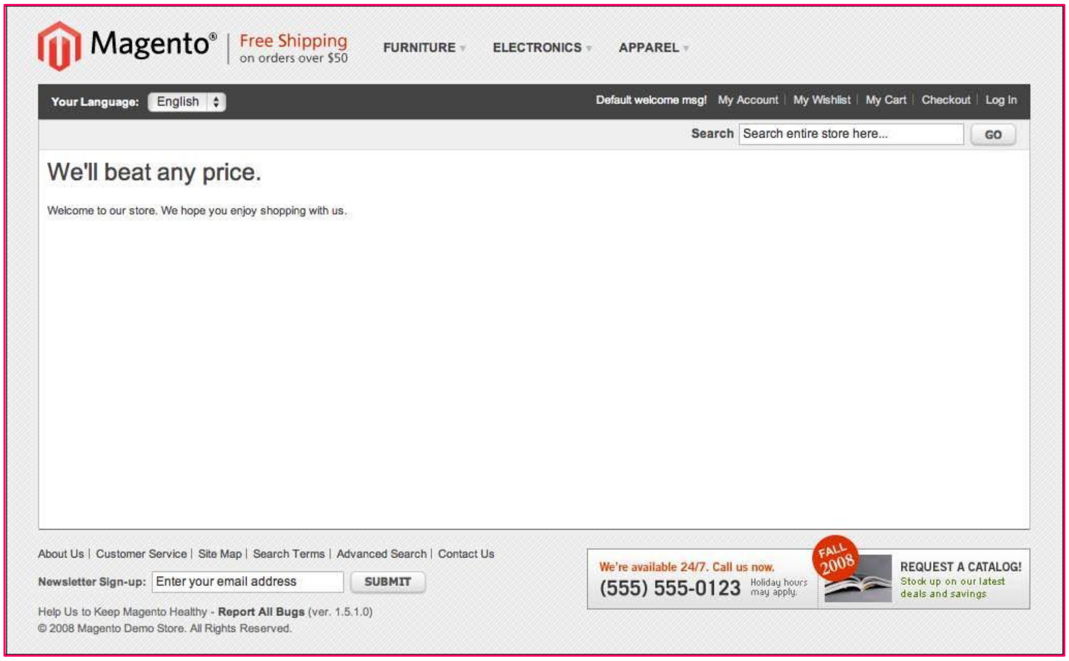
This example tells Magento to include the content of an HTML page that we created and stored within a folder called sample. This folder lies within the directory of our theme (in this case, modern). This file, home.phtml, contains the HTML code to display the graphics showing on the home page, as shown earlier.

For our exercise, let's remove this code and replace it with the following:

Welcome to our store. We hope you enjoy shopping with us.

In the field at the top, labelled **Content Heading**, enter: *We'll beat any price.*

Click on **Save and Continue Edit** (this will let you remain on this screen) and open your home page in another browser window or tab. It should look something like the following screenshot:



Your Language: English

Search: Search entire store here...

We'll beat any price.

Welcome to our store. We hope you enjoy shopping with us.

Newsletter Sign-up: Enter your email address

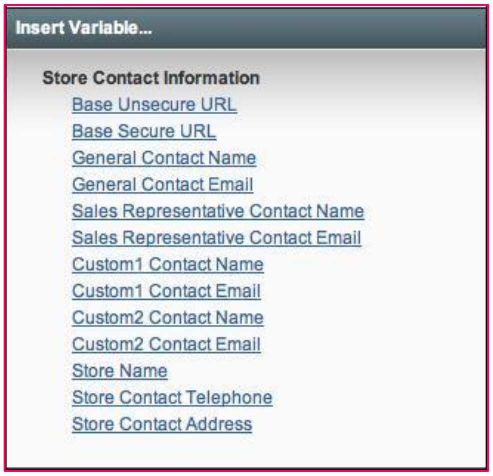
Help Us to Keep Magento Healthy - **Report All Bugs** (ver. 1.5.1.0)
© 2008 Magento Demo Store. All Rights Reserved.

The text you put into the **Content Heading** field appears at the top of the content section automatically styled with an <h1> tag.

Now you've created your first custom home page content, but even you have to admit, it's not very engaging. Let's return to the **Edit Page** screen.

Adding variable content

At the upper right-hand side of the WYSIWYG bar at the top of the editing area of the panel, the first icon is for inserting dynamic information into your content. If you click on this icon, a pop-up dialog box will appear allowing you to select from several items, as shown in the following screenshot:



Insert Variable...

Store Contact Information

- Base Unsecure URL
- Base Secure URL
- General Contact Name
- General Contact Email
- Sales Representative Contact Name
- Sales Representative Contact Email
- Custom1 Contact Name
- Custom1 Contact Email
- Custom2 Contact Name
- Custom2 Contact Email
- Store Name
- Store Contact Telephone
- Store Contact Address

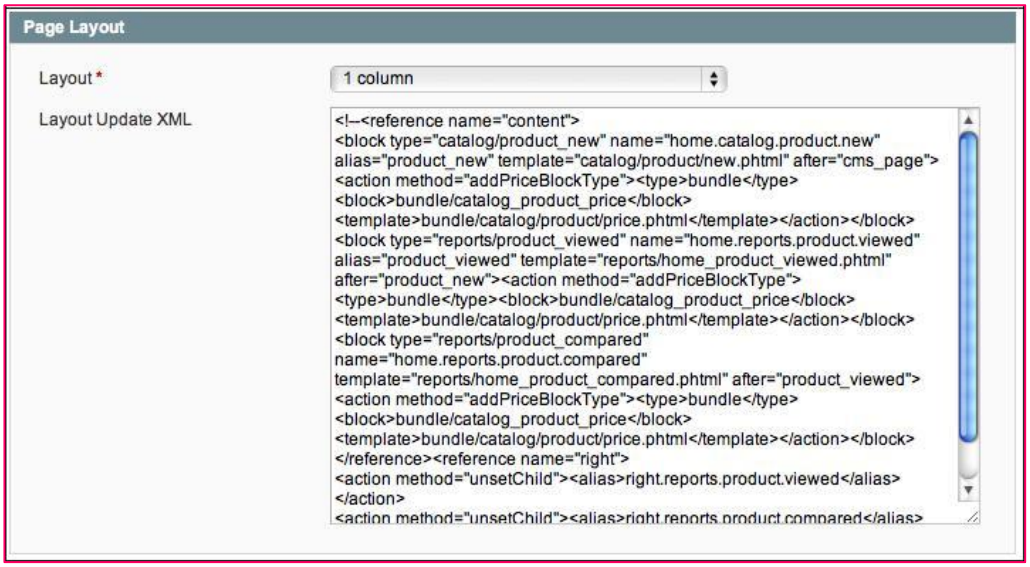
These variables can come in handy! For instance, if you're using the same home page layout for several stores, but want to show the name of the store in the content, click on **store name** in this list and a variable will be placed into the content editing field. This variable will show the name of the store as you configured it in the store management screen (**System | Manage Stores**). This is much easier than building multiple home pages, one for each store, just to show the current store name.

> You can create your own custom variables to use by going to **System | Custom Variables**. I use this feature to create custom variables for displaying a store's phone number or business hours.

The design screen

Before we put more content into the **Content** panel, click on the **Design** tab on the left-hand side. The design screen allows you to do much more in terms of altering the layout and surrounding elements of a page. As I suggested before, you can apply the same principles as you did with the local.xml file described in *Chapter 4*, *Designs and Themes* at the individual page level. These changes command the highest control over your page, overriding layout handles in the theme layout and local. xml files.

The next screenshot depicts the **Page Layout** panel for the same home page that we're using in this section:



Page Layout

Layout * 1 column

Layout Update XML

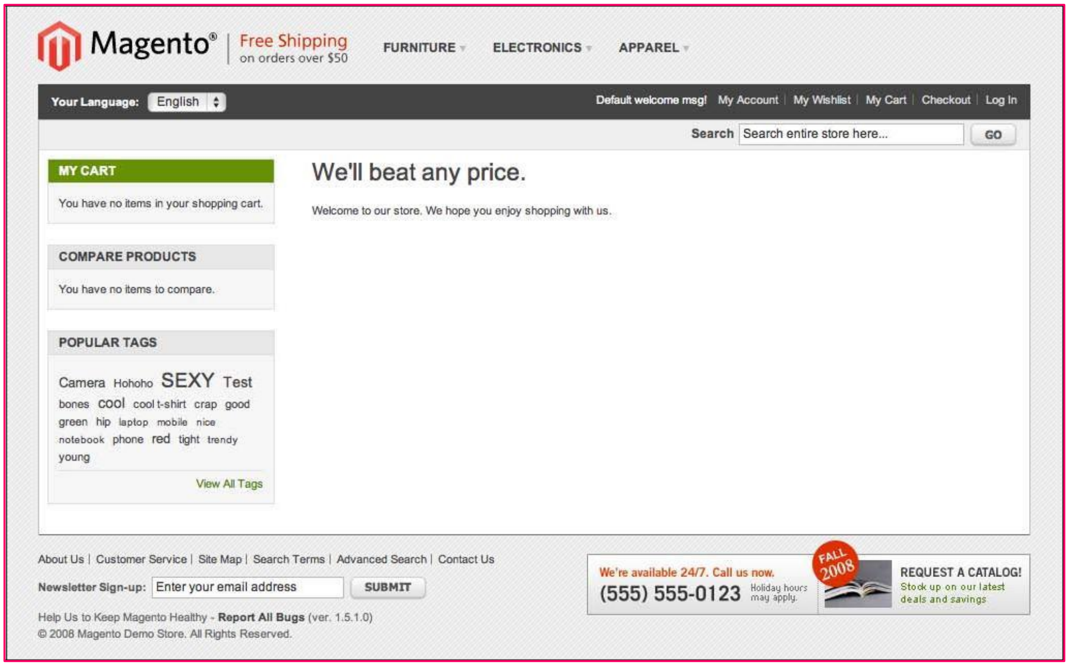
```
<!--<reference name="content">
<block type="catalog/product_new" name="home.catalog.product.new"
alias="product_new" template="catalog/product/new.phtml" after="cms_page">
<action method="addPriceBlockType"><type>bundle</type>
<block>bundle/catalog_product_price</block>
<template>bundle/catalog/product/price.phtml</template></action></block>
<block type="reports/product_viewed" name="home.reports.product.viewed"
alias="product_viewed" template="reports/home_product_viewed.phtml"
after="product_new"><action method="addPriceBlockType">
<type>bundle</type><block>bundle/catalog_product_price</block>
<template>bundle/catalog/product/price.phtml</template></action></block>
<block type="reports/product_compared"
name="home.reports.product.compared"
template="reports/home_product_compared.phtml" after="product_viewed">
<action method="addPriceBlockType"><type>bundle</type>
<block>bundle/catalog_product_price</block>
<template>bundle/catalog/product/price.phtml</template></action></block>
</reference><reference name="right">
<action method="unsetChild"><alias>right.reports.product.viewed</alias>
</action>
<action method="unsetChild"><alias>right.reports.product.compared</alias>
```

Every theme automatically includes four basic layouts:

- 1 column
- 2 columns with left sidebar
- 2 columns with right sidebar
- 3 columns

You also have the choice of selecting *Empty* in the **Layout** drop-down menu. This renders a page with nothing but what is directly specified in this page: no header, navigation, or footer. Go ahead and give it a try; I know you want to!

As for the layouts listed here, let's try each one and see how it immediately affects our page. Since we're already using the *1 column* layout, we'll start with *2 columns with left sidebar*:



You have no items in your shopping cart.

You have no items to compare.

We'll beat any price.

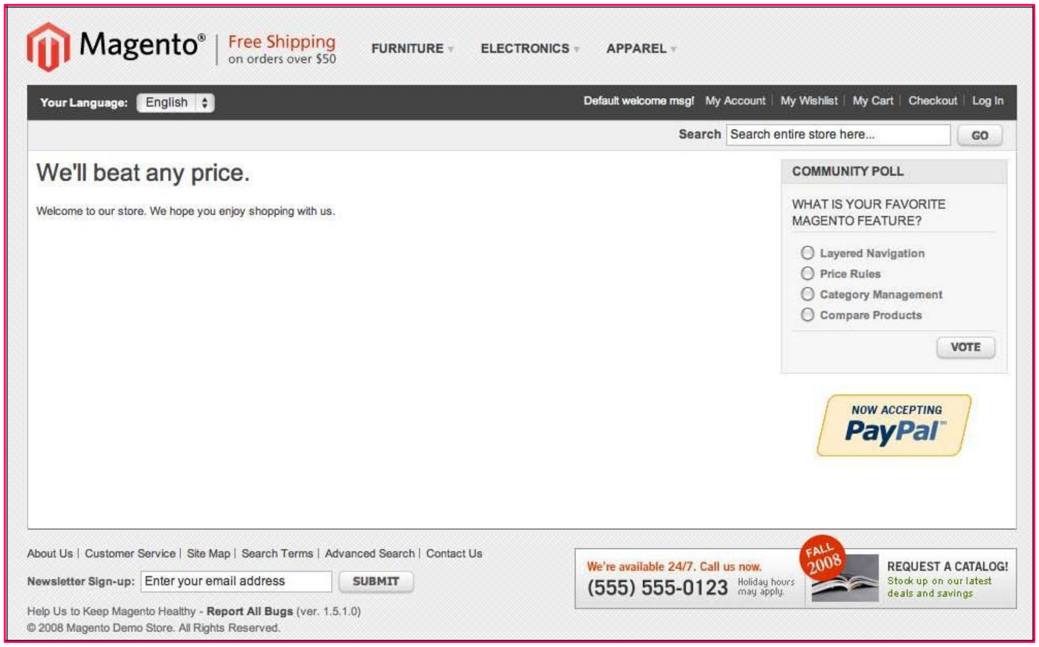
Welcome to our store. We hope you enjoy shopping with us.

We're available 24/7. Call us now.
(555) 555-0123 Holiday hours may apply.

Help Us to Keep Magento Healthy - **Report All Bugs** (ver. 1.5.1.0)
© 2008 Magento Demo Store. All Rights Reserved.

The elements, or blocks, showing in the left-hand sidebar are defined by the page layout XML file within the theme.

Compare this with the *2 columns with right sidebar* layout:


Magento®
Free Shipping on orders over $50
FURNITURE
ELECTRONICS
APPAREL
Your Language: English
Default welcome msg! My Account My Wishlist My Cart Checkout Log In
Search
GO
COMMUNITY POLL
VOTE
NOW ACCEPTING PayPal
About Us | Customer Service | Site Map | Search Terms | Advanced Search | Contact Us
SUBMIT
FALL 2008
REQUEST A CATALOG!

Search entire store here...

We'll beat any price.

Welcome to our store. We hope you enjoy shopping with us.

WHAT IS YOUR FAVORITE MAGENTO FEATURE?

○ Layered Navigation
○ Price Rules
○ Category Management
○ Compare Products

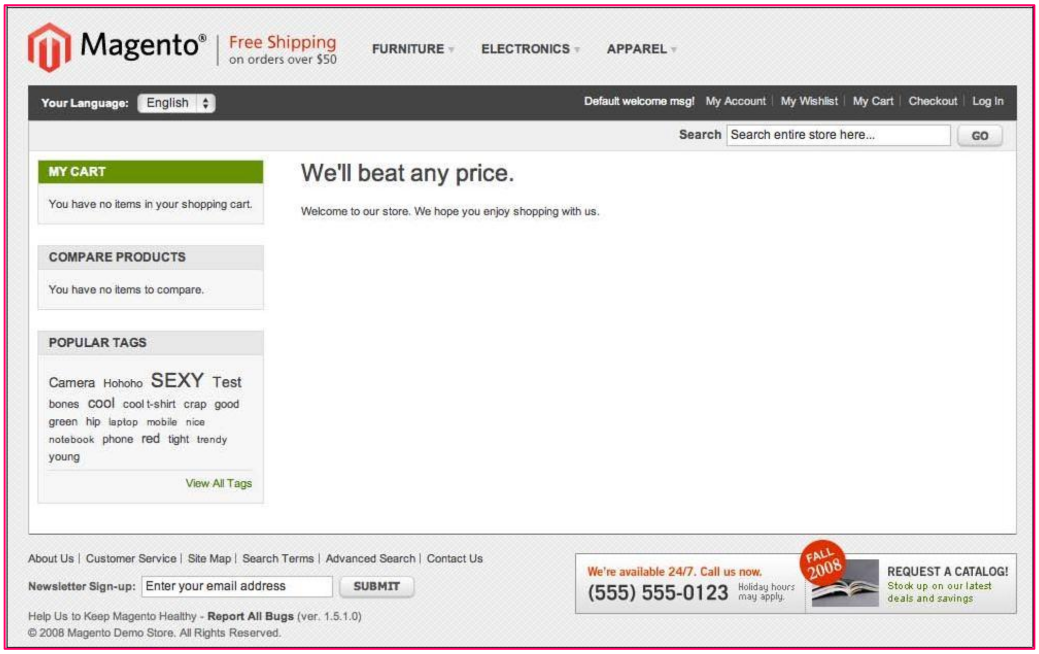
Newsletter Sign-up: Enter your email address

We're available 24/7. Call us now. (555) 555-0123 Holiday hours may apply.

Stock up on our latest deals and savings

Help Us to Keep Magento Healthy - **Report All Bugs** (ver. 1.5.1.0)
© 2008 Magento Demo Store. All Rights Reserved.

And finally, let's see what the *3 columns* layout looks like:


Magento®
Free Shipping on orders over $50
FURNITURE
ELECTRONICS
APPAREL
Your Language: English
Default welcome msg! My Account My Wishlist My Cart Checkout Log In
Search
GO
MY CART
COMPARE PRODUCTS
POPULAR TAGS
Camera Hohoho SEXY Test bones cool cool t-shirt crap good green hip laptop mobile nice notebook phone red tight trendy young
View All Tags
About Us | Customer Service | Site Map | Search Terms | Advanced Search | Contact Us
SUBMIT
FALL 2008
REQUEST A CATALOG!

Search entire store here...

You have no items in your shopping cart.

You have no items to compare.

We'll beat any price.

Welcome to our store. We hope you enjoy shopping with us.

Newsletter Sign-up: Enter your email address

We're available 24/7. Call us now. (555) 555-0123 Holiday hours may apply.

Stock up on our latest deals and savings

Help Us to Keep Magento Healthy - **Report All Bugs** (ver. 1.5.1.0)
© 2008 Magento Demo Store. All Rights Reserved.

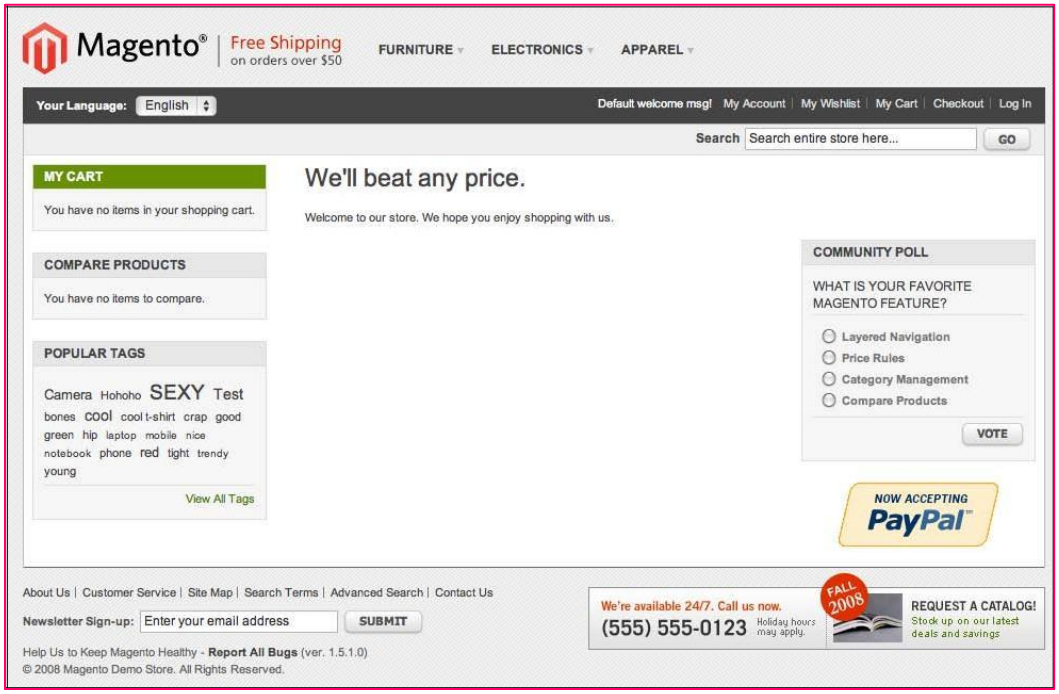
Wait a minute! This looks just like *2 columns with left sidebar*, right? Yes, because the template HTML file, template/page/3columns.phtml, used for this layout has the right sidebar commented out:

```
<?php /*<div class="col-wrapper">*/ ?>
    <div class="col-main">
        <?php echo $this->getChildHtml('global_messages') ?> <?php echo $this-
        >getChildHtml('content') ?>
    </div>
    <div class="col-left sidebar">
        <?php echo $this->getChildHtml('left') ?></div> </div>
    <?php /*<div class="col-right sidebar">
    <?php echo $this->getChildHtml('right') ?></div>*/ ?> <?php /*</div>*/ ?>
```

To see what all three columns might look like, we need to *un*-comment this code so it now looks as follows:

```
<div class="col-wrapper">
    <div class="col-main">
        <?php echo $this->getChildHtml('global_messages') ?> <?php echo $this-
        >getChildHtml('content') ?>
    </div>
    <div class="col-left sidebar">
        <?php echo $this->getChildHtml('left') ?></div>
</div>
<div class="col-right sidebar">
    <?php echo $this->getChildHtml('right') ?></div> </div>
```

Now when we look at our home page, we see the following:


Magento®
Free Shipping on orders over $50
FURNITURE
ELECTRONICS
APPAREL
Your Language: English
Default welcome msg! My Account | My Wishlist | My Cart | Checkout | Log In
Search
GO
MY CART
COMPARE PRODUCTS
POPULAR TAGS
Camera Hohoho SEXY Test bones cool cool t-shirt crap good green hip laptop mobile nice notebook phone red tight trendy young
View All Tags
COMMUNITY POLL
○ Layered Navigation
○ Price Rules
○ Category Management
○ Compare Products
VOTE
NOW ACCEPTING PayPal
About Us | Customer Service | Site Map | Search Terms | Advanced Search | Contact Us
Newsletter Sign-up:
SUBMIT
FALL 2008
REQUEST A CATALOG!

Search entire store here...

You have no items in your shopping cart.

You have no items to compare.

We'll beat any price.

Welcome to our store. We hope you enjoy shopping with us.

WHAT IS YOUR FAVORITE MAGENTO FEATURE?

Enter your email address

We're available 24/7. Call us now. (555) 555-0123 Holiday hours may apply.

Stock up on our latest deals and savings

Help Us to Keep Magento Healthy - **Report All Bugs** (ver. 1.5.1.0)
© 2008 Magento Demo Store. All Rights Reserved.

In the **Page Layout** panel, the larger field, called **Layout Update XML**, allows you to use XML code to manipulate the layout of the page. The sample code that's shown here is commented out (by using <-- before the code and --> after the code), but remains to help you understand how to use XML to change the layout. Refer to *Chapter 4*, *Designs and Themes* for more information on using handles, blocks, and actions to manipulate page layouts.

The bottom panel on the design screen—**Custom Design**—allows you to set an alternative layout for the page that takes effect during the time span indicated in the two date fields.

The Meta Data screen

The final **Edit Page** screen is for adding meta keywords and a meta description for the page. Entering values here will override any default values that you set in the system configuration settings.

Static blocks

When managing a website, it's often nice to have snippets of code or design elements that you can use in one or more places, yet be able to edit without digging into the template files of your theme.

Static blocks allow you to do just that. From creating blocks for footer links to setting up a special promotional block, static blocks are styled in the same manner as you used in the **Edit Page Content** screen.

Once you create a static block, you can insert it into the **Content** panel of any page. You can also refer to a static block in a layout XML file. For example, the sample data installation includes a static block entitled *Footer Link*, with an identifier of *footer_ links*. In the base package cms.xml layout file, this static block is referred to with the following code (the comments are included by default):

```
<reference name="footer"> <block
   type="cms/block"
      name="cms_footer_links" before="footer_links"> <!--
      The content of this block is taken from the database by its block_id.
      You can manage it in admin CMS -> Static Blocks -->
   <action method="setBlockId">
      <block_id>footer_links</block_id></action>
   </block>
</reference>
```

Once this block is defined, it can be called upon in a template HTML file, such as, in this case, footer.phtml, where the following PHP code pulls the *footer_links* static block content stored in the Magento database into this page component:

```
<?php echo $this->getChildHtml('cms_footer_links') ?>
```

Widgets

If you look at the WYSIWYG editor bar at the top of any content editing field, the second icon from the left is for inserting widgets into the content area. There are several types of widgets you can use in Magento.

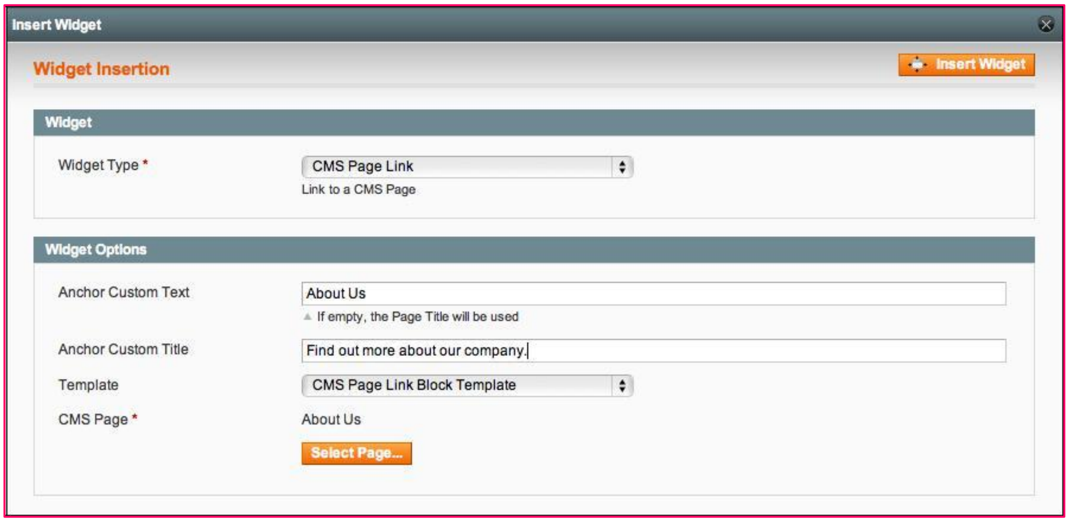
While you can insert Widgets within any **Page Content** area, under **CMS | Widget**, you can configure widgets that will automatically appear on one or more pages in your store. It's a powerful way of adding customized features to your Magento store without doing a lot of heavy programming and configuration.

The next few sections are overviews of the default Magento widgets (some third-party extensions will add additional widgets). Whether you insert a widget into a **Page Content** area, or configure it to appear automatically, the configuration principles remain the same.

CMS page link widget

A **CMS page link widget** allows you to insert a link to one of your CMS pages. This is convenient if you have, for example, a page showing product brands that you want to insert into a static block or a CMS page.

As shown in the next screenshot, this widget gives you control over how the link will appear on your page. You can also choose between two layout templates, a **Block** style or an **Inline** style, depending on how you want the link to be controlled using CSS rules.



Insert Widget

Widget Insertion

Widget

Widget Type * CMS Page Link
Link to a CMS Page

Widget Options

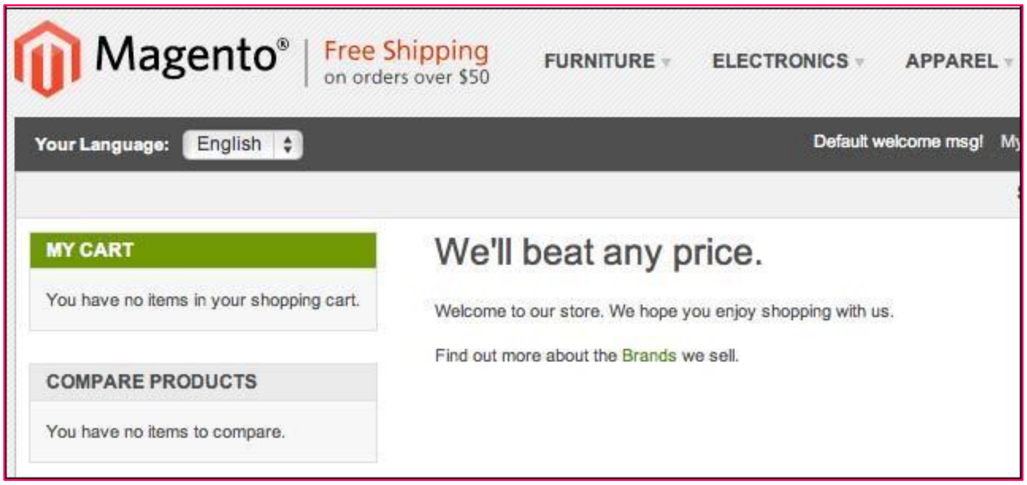
Anchor Custom Text: About Us
If empty, the Page Title will be used

Anchor Custom Title: Find out more about our company.

Template: CMS Page Link Block Template

CMS Page *: About Us

After saving the CMS page, the front end will show the link you specified, as shown in the following screenshot:



We'll beat any price.

Welcome to our store. We hope you enjoy shopping with us.

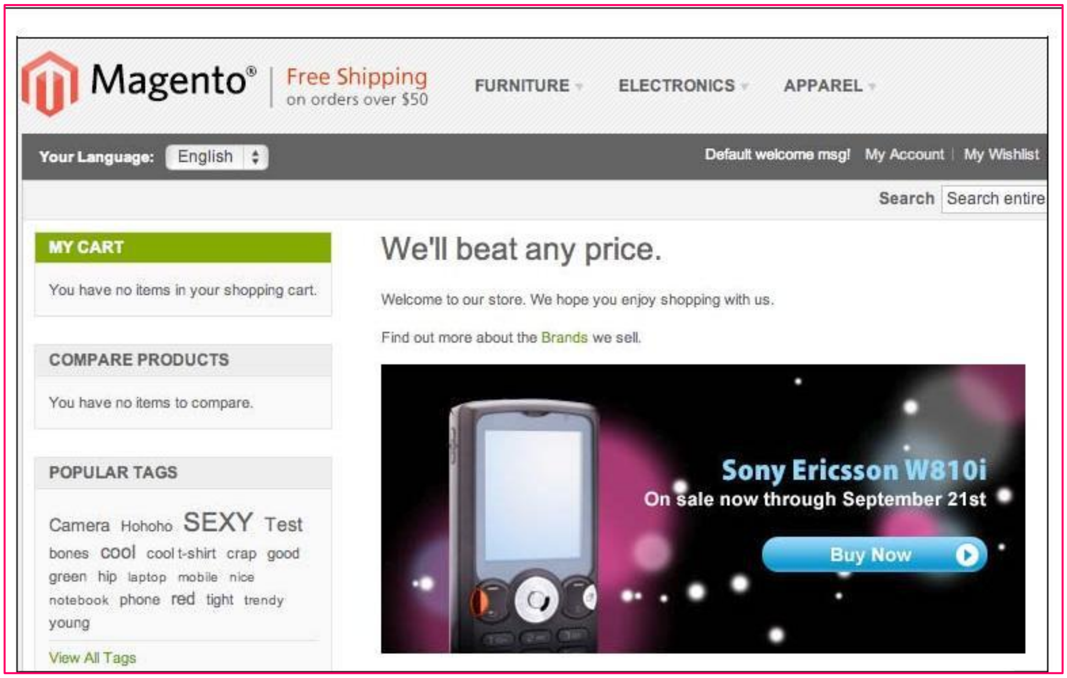
Find out more about the Brands we sell.

You have no items in your shopping cart.

You have no items to compare.

CMS static block

If you have static blocks that you have created, you can insert them into a page using the **CMS static block widget** instance. Simply choose the static block after clicking on **Select Block…** In the next screenshot, I added a static block that contained the cell phone image shown:



You have no items in your shopping cart.

You have no items to compare.

We'll beat any price.

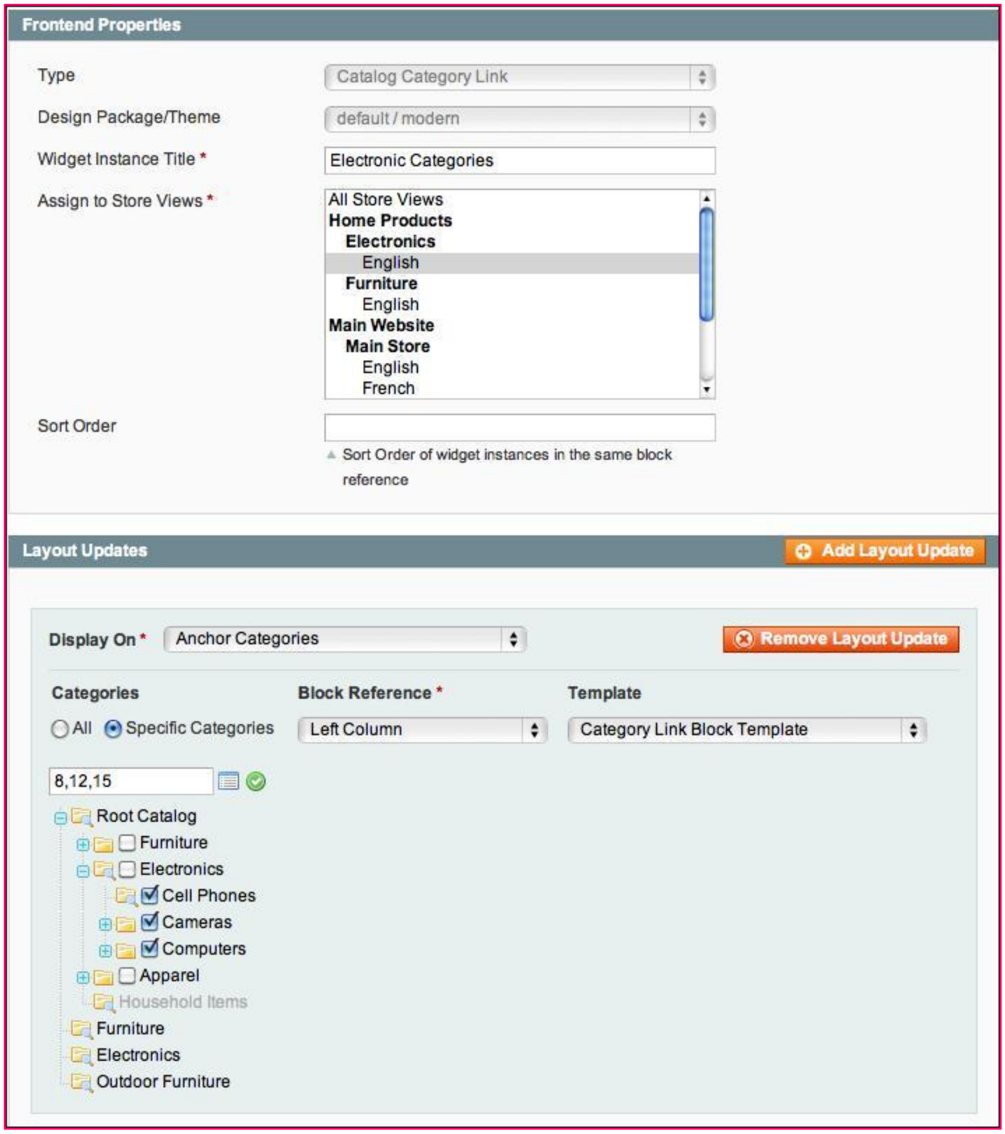
Welcome to our store. We hope you enjoy shopping with us.

Find out more about the Brands we sell.

There is only one default template with a Magento installation. If you wish to change how static blocks are displayed in your theme, copy the template file from app/design/frontend/base/default/template/cms/widget/static_block/ default.phtml and place it in the same directory hierarchy in your theme (for example, app/design/frontend/[design package]/[design theme]/template/ cms/widget/static_block/). This might be handy if you want to add a new CSS style to your static block layouts Catalog a category link

The **catalog category link widget** gives you the ability to create custom links that will take the visitor to a list of all products within a designated category. You'll find the configuration choices described here apply to other similar widgets.

In the example shown next for instance, we have set up a widget to appear on all **Anchor Category** pages in the *Left Column* of the *English Language* **Store View**. Furthermore, we have chosen only three categories—**Cell Phones**, **Cameras**, and **Computers**—from the list of all categories. Once saved, this widget will automatically create a link only on those three category pages.



Frontend Properties

Type: Catalog Category Link

Design Package/Theme: default / modern

Widget Instance Title *: Electronic Categories

Assign to Store Views *:
All Store Views
Home Products
Electronics
English
Furniture
English
Main Website
Main Store
English
French

Sort Order:

Sort Order of widget instances in the same block reference

Layout Updates

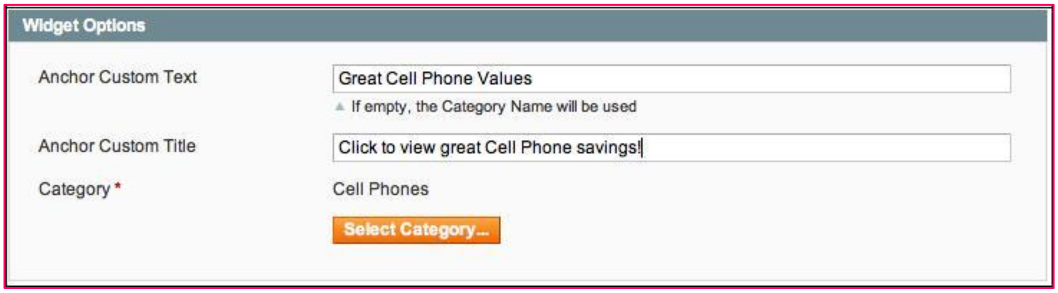
Display On *: Anchor Categories

Categories	Block Reference *	Template
○ All ● Specific Categories	Left Column	Category Link Block Template

8,12,15

Root Catalog
□ Furniture
□ Electronics
☑ Cell Phones
☑ Cameras
☑ Computers
□ Apparel
Household Items
Furniture
Electronics
Outdoor Furniture

Now, if you click on the **Widget Options** tab on the left-hand side of the screen, you have the opportunity to name the link created by this widget, the link's title tag value, and select the **Category** page for the link destination, as shown in the following screenshot:



Widget Options

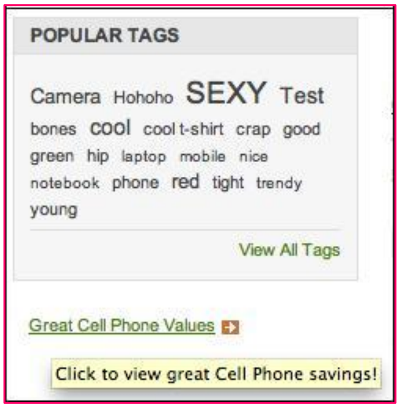
Anchor Custom Text: Great Cell Phone Values

If empty, the Category Name will be used

Anchor Custom Title: Click to view great Cell Phone savings!

Category *: Cell Phones

Using the values shown here, let's look now and see what this link looks like on one of our store pages:



POPULAR TAGS

Camera Hohoho SEXY Test bones cool cool t-shirt crap good green hip laptop mobile nice notebook phone red tight trendy young

View All Tags

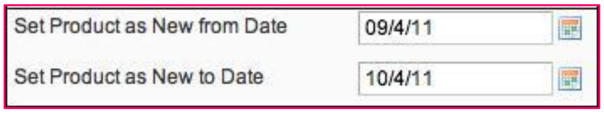
Great Cell Phone Values

Click to view great Cell Phone savings!

In this example, you can see our new link appearing beneath the **POPULAR TAGS** block, with the title tag value we specified when a user hovers their mouse pointer over the link. By clicking on this link, the visitor is taken to the same category page as if they clicked on **Cell Phones** in the upper navigation panel on the page.

Creating a new products list

Some e-commerce platforms designate new products using a simple checkbox. While this may be quick, Magento approaches new products with its usual degree of increased functionality. To set a product as *new*, you simply enter a **Set Product as New from Date** and, optionally, a **Set Product as New to Date**, as shown in the following screenshot from the **General** tab of the **Product Information** screen:



Set Product as New from Date 09/4/11

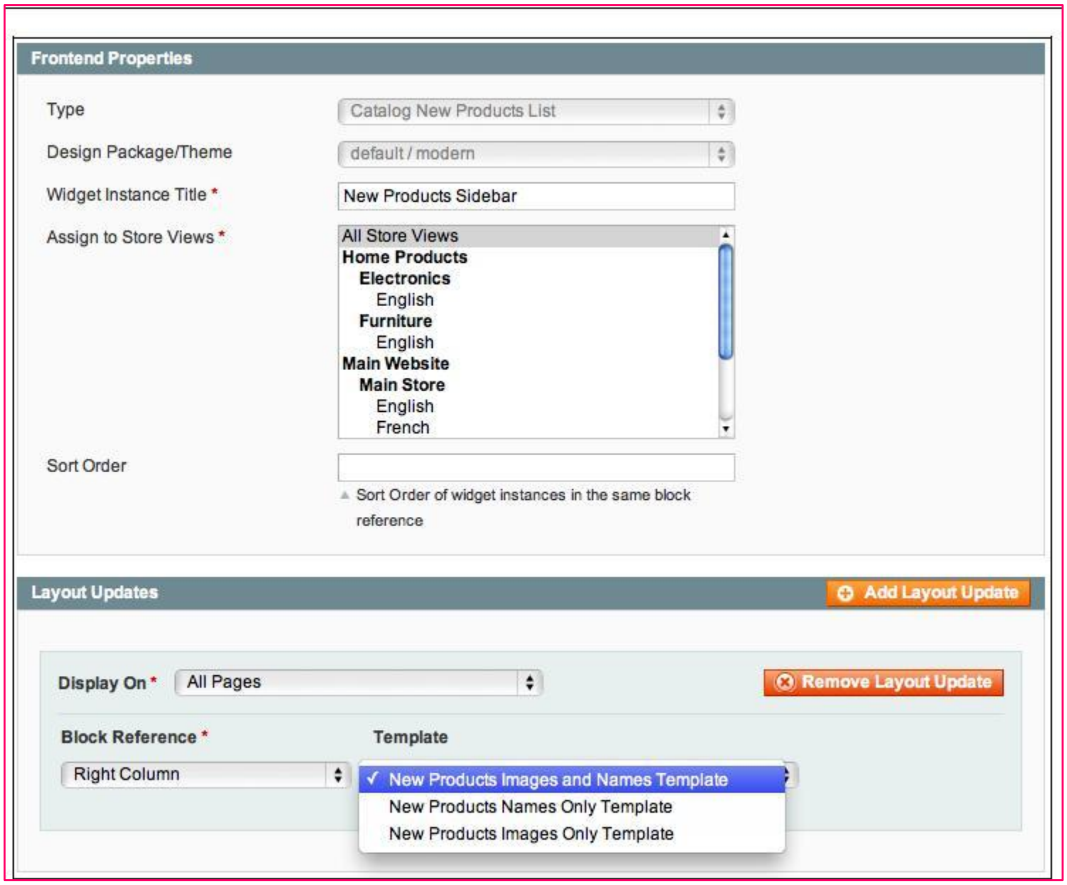
Set Product as New to Date 10/4/11

In this example, the edited product would be considered by Magento as *new* from September 4, 2012 through October 4, 2012. I like the ability to pre-determine when a **new product** would no longer be considered *new* without having to manually go through potentially hundreds or thousands of products and manually de-select a checkbox.

Set New When You Import

If you regularly import products from lists provided by distributors or manufacturers, many such feeds include a field that flags new products. When you import these products into Magento (see *Chapter 3*, *Managing Products*), you could set the **New from Date** field as the date of the import. You can either calculate the **New to Date** field (for example, add 30 days to the **New from Date** value), or leave it blank if you want the product to remain new for an indefinite period of time. This process keeps your new products changing, fresh, and up-to-date without continued interaction on your part.

Similar in setup to the aforementioned, the **Catalog New Products List Widget**, shown next, is likewise configured by setting the applicable store views, on what pages the widget is to appear, the desired block reference and, in this case, one of three display templates.



Frontend Properties

Type: Catalog New Products List

Design Package/Theme: default / modern

Widget Instance Title *: New Products Sidebar

Assign to Store Views *: All Store Views

Sort Order:

Sort Order of widget instances in the same block reference

Layout Updates

Display On *: All Pages

Block Reference *: Right Column

Template: ✓ New Products Images and Names Template

Catalog a product link

The **Catalog product link widget** works very much like the **catalog category link widget**, except, of course, you're choosing a specific **Product Detail Page** for the visitor instead of a **Category Page**.

Recently compared and recently viewed products

These two widgets do just as you would expect: display **Recently Compared** and **Recently Viewed** products in blocks you configure to appear in select pages in your store. Most Magento themes include these as sidebar blocks. However, I've found value in including these in such distinct locations as the Cart Page, the Checkout Page, and the Contact Us Page.

Principles of customizing layouts

After installing and configuring many Magento stores, I'm convinced that if you master the process of managing layouts, your stores will produce better sales results and give you better opportunities to customize a visitor's experience based on the needs of both the retailer and the customer.

That being said, the complexity of the Magento architecture demands your dedicated time and effort to achieve a level of true mastery. I suggest you carefully review *Chapter 4*, *Designs and Themes* and this chapter, truly examining the examples. As you do that, remember the following keys:

- At their core, page layouts are determined by XML files which specify the location of various blocks.
- Template files contain the HTML and PHP code that creates the final output of the pages.
- The graphic design of pages are greatly influenced by the CSS, JavaScript, and image files located in the skin/ directory. By altering these, you can do quite a lot of brand customization.
- The Magento CMS functionality works best when you understand how blocks are used to build pages, even if you don't actually dig into the layout and template files.
- Above all, experiment. Especially with the CMS, you really can't break anything. The more you try different configurations, the more opportunities you'll discover to give your Magento store the added punch it deserves.

Summary

The Magento CMS provides you with the capability to customize your store without directly altering the XML and HTML files of your theme. You can build productive pages, static blocks, and widgets, adding increased functionality and buyer conveniences.

That being said, we should never shy away from rolling up our sleeves and diving into the actual theme files. By combining both disciplines, you have greatly elevated Magento's potential to meet your retailing needs. In

this chapter, we:

- Learned about Magento's CMS functionality
- Saw how to create and customize pages
- Explored the creation of static blocks
- Reviewed the many widgets available for your use

Now that our Store has products and content, it's time to let the world know that your store exists as a premiere online shopping destination.

4. Marketing Tools

The online marketplace is crowded. Thousands of new stores are going online each day. With Magento, you have one of the most powerful open source platforms for presenting and selling your products and services. But to be noticed—and to create valuable repeat business—you have to do more than just point a domain name to your store.

The key is marketing—presenting a selling proposition to potential customers that will entice them to at least visit your online showroom.

In this chapter, we will explore some of Magento's key marketing tools, including:

- Customer groups
- Promotions
- Newsletters
- Sitemaps
- Search engine optimization

As you prepare your store for launch, you need to spend some time becoming familiar with how these tools can help you attract and close more sales.

Customer groups

If you're new to e-commerce, you may not quite have a handle on how to group your potential customers, outside of retail and wholesale. Most of you will build retail stores, fewer will build wholesale businesses, and even fewer will do both.

Yet, if you give some careful thought to how you intend to market to your customers, you may find that there are more customer segments you need to identify. Here are examples of some other possible customer groups:

- Distributors who will re-sell your product to retailers
- Tax-exempt organizations, such as charities or government agencies
- Groups or teams, such as associations or athletic leagues
- Professionals, as opposed to hobbyists or amateurs
- Certified or approved buyers meeting a licensing or other qualification

Customer groups in Magento are not to be confused with customer profiles, such as age, sex, or preferences. A customer can only belong to one Magento group, so profile attributes cannot really be accommodated.

As you contemplate your **groups**, consider the following:

- Groups should be segments of your customer base that might have different pricing, benefits, or product focus
- Groups can be used (as we'll see later in this chapter) for specific marketing promotions using newsletters and promotional codes
- Groups can be assigned different tax classes for the purpose of charging (or not charging) sales tax

By default, Magento creates three Customer groups: General, Retailer, and Wholesale. When a customer registers in your store, they are automatically put into the General group. You can manually assign a customer to another group, either after they have registered or if you create the customer yourself in the administrative backend to your store.

> Strangely enough, the Sample Data installation only creates one **Customer Tax Class**. Therefore, all the default Customer groups are assigned to the **Retail Customer Tax Class**. If you use other groups, you may want to create additional Customer Tax Classes (under **Sales** | **Tax** | **Customer Tax Classes**).

Creating a Customer group

Creating a new Customer group is perhaps the simplest operation in Magento!

1. Go to **Customers** | **Customer Groups** in your administration menu.
2. Click on **Add New Customer Group**.
3. Enter a **Group Name**.
4. Assign a **Tax Class** as created in *Chapter 5*, *Configuring to Sell*.
5. Click on **Save Customer Group**.

And that's all there is to it. Now that you understand how Customer groups are created in Magento, let's see how we can use these for marketing.

Promotions

Every retailer, with whom we work, finds it beneficial to offer promotions from time to time. From free shipping to coupons, promotions can help consumers choose your products over competitors, particularly if they feel they're getting a great bargain in the process.

As with Sales Tax, promotions are rule-based, meaning that the application and calculation of a promotion is based on rules you create. In Magento, there are two types of **Promotion Rules**:

1. **Catalog Price Rules** apply to any product that meets certain criteria. The discount is applied automatically.
2. **Shopping Cart Rules**, on the other hand, can be set to be only applied when a valid coupon is entered by the customer during Checkout. Shopping Cart Rules can also be applied automatically. Furthermore, Shopping Cart Rules— or **Coupons**—can be publicized by adding them to an RSS feed for your site.

Creating a Catalog Price Rule

As described, a Catalog Price Rule is a discount that is applied to selected products automatically. For example, using the installed Sample Data, let's assume you wish to provide a 5 percent discount on all Dell® brand products priced over 500 dollars sold between November 25, 2011 through November 27, 2011, as a Black Friday special.

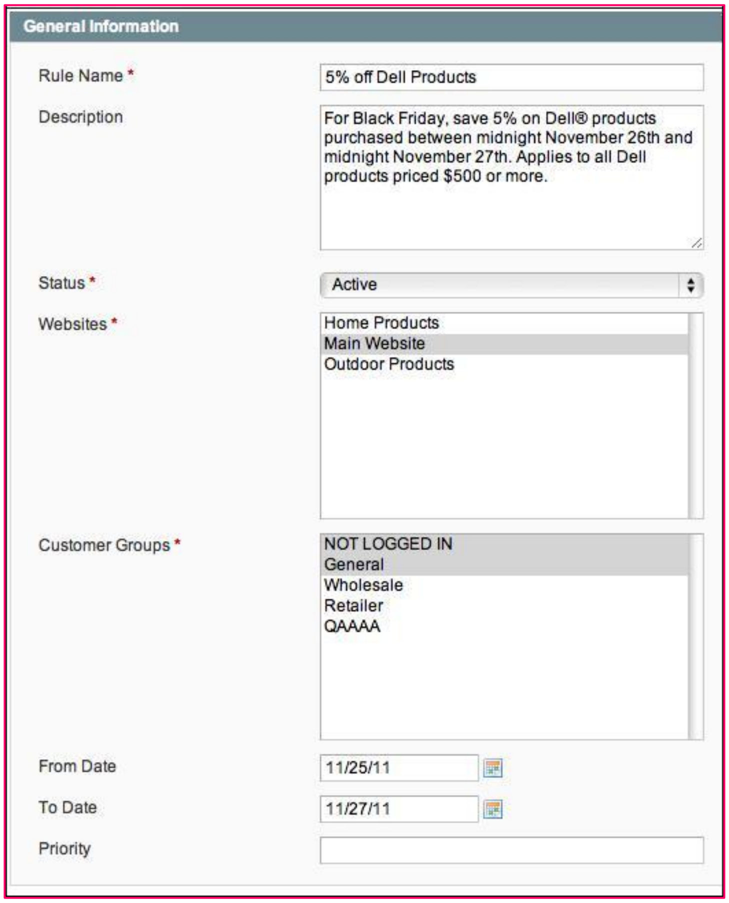
Black Friday and Cyber Monday

For years, in the United States, the day after Thanksgiving—the last Friday of November, Black Friday—has been one of the busiest shopping days of the Christmas season. Retailers offer huge discounts and shoppers begin lining up well before dawn just to be the first one in the door. To counter the brick and mortar retailers, online retailers created Cyber Monday,

the Monday following Black Friday, likewise offering huge discounts and special offers. Personally, I like the sound of Cyber Monday over the morbid moniker of Black Friday. I also like e-commerce!

Let's begin creating a new Catalog Price Rule:

1. Go to **Promotions** | **Catalog Price Rules**.
2. Click **Add New Rule**.
3. Enter the following in the **General Information** screen:



General Information

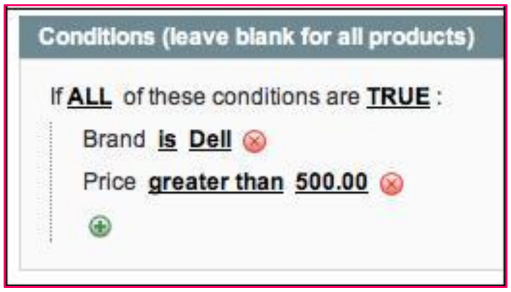
Rule Name *	5% off Dell Products
Description	For Black Friday, save 5% on Dell® products purchased between midnight November 26th and midnight November 27th. Applies to all Dell products priced $500 or more.
Status *	Active
Websites *	Home Products Main Website Outdoor Products
Customer Groups *	NOT LOGGED IN General Wholesale Retailer QAAAA
From Date	11/25/11
To Date	11/27/11
Priority	

4. Click on **Conditions** in the left tab menu. Now, here's where it gets interesting!
5. Click the green plus sign to add a condition to your rule.

> Any underlined items in these screens can be clicked to reveal additional choices. Don't hesitate to click and discover!

6. In the drop-down menu that appears, choose **Brand**.
7. Click the ellipsis (**…**).
8. In the drop-down menu, choose **Dell**.
9. Click on the green plus sign again, and choose **Price** from the drop-down menu.
10. Click on **is**. Select **greater than** from the drop-down menu.
11. Click the following ellipsis. Enter **500.00** in the field and click outside the field.

Your conditions screen should look as follows:



Conditions (leave blank for all products)

If ALL of these conditions are TRUE :

Brand is Dell

Price greater than 500.00

This promotion will now only apply to Dell brand products priced 500 dollars or more.

> The amount of the discount is based on the default currency for the website. If your site's default currency is Euros, then the discount would be 500 Euros.

Now, we need to create the discount calculation: 1.

Click the **Actions** menu tab.

> The four choices for **Apply** give you tremendous flexibility, but may not be clearly understood.

2. **By Percentage of the Original Price** reduces the price of the product by the percentage specified, such as 15 percent off the price.
3. **By Fixed Amount** will reduce the product's price by a specified amount, such as 50 dollars off.
4. **To Percentage of the Original Price**, by contrast to **By Percentage of the Original Price**, sets the price at the percentage you enter. For example, if you enter 75, the price of the product would be 75 percent of its original price.
5. Likewise, **To Fixed Amount**, sets the price of the product at the price you enter instead of reducing the price by a fixed amount.
6. Since we're going to apply a percentage discount to the price of the product, we can leave **Apply** as is.
7. In **Discount Amount**, enter **5**.
8. If you do not want to apply any other Promotion Rules to these applicable products, choose **Yes** for **Stop Further Rules Processing**. For example, you may not want a customer to receive multiple discounts on the same product.
9. Click on **Save Rule**.

You now have a new rule that will automatically take effect at 12:01 AM on Friday, November 25 and run until 11:59 PM on Sunday, November 27. However, there is one final action to take!

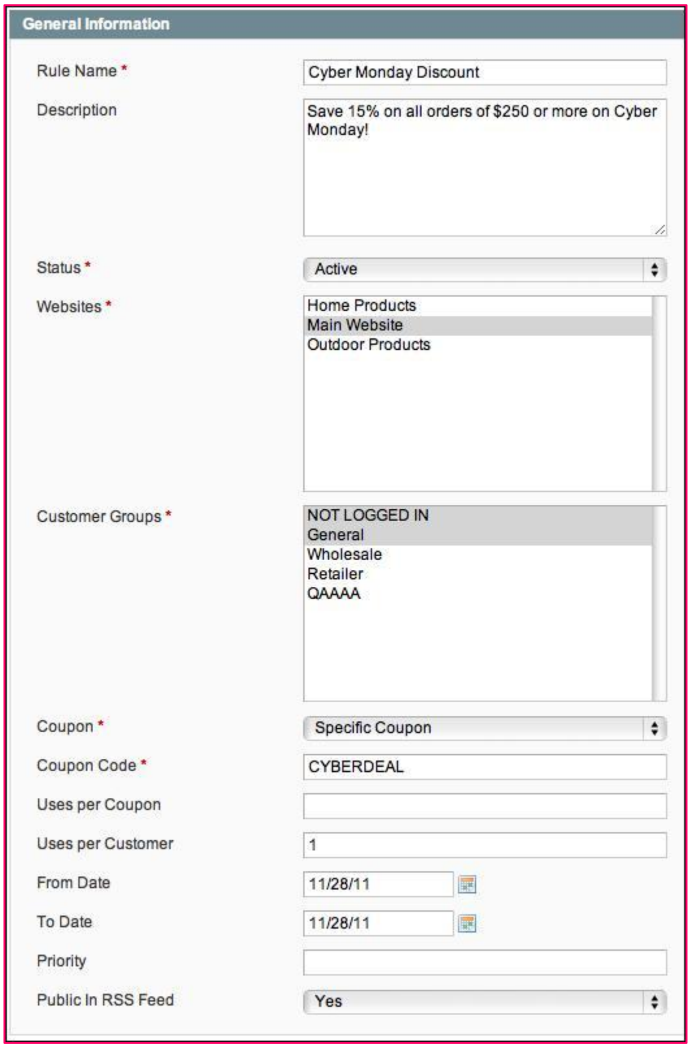
Before your new rule can take effect, click **Apply Rules** at the top of the **Catalog Price Rules** page. Magento needs to set itself to apply the rules to the applicable products.

> For Catalog Price Rules to be effective, the Magento cron script must be running. I discuss triggering cron jobs in ,
> *Advanced Techniques*.

Creating a Shopping Cart Rule

As a capitalizing entrepreneur, you've decided that while you're offering 5 percent off on Dell products for Black Friday, you want to give loyal customers a special coupon they can use on Cyber Monday only that will give them 15 percent off all orders of 250 dollars or more. What a generous retailer you are!

1. Go to **Promotions** | **Shopping Cart Price Rules**.
2. Click **Add New Rule**.
3. Enter the following in the **General Information** screen:



General Information

Rule Name *	Cyber Monday Discount
Description	Save 15% on all orders of $250 or more on Cyber Monday!
Status *	Active
Websites *	Home Products [✓ Main Website] Outdoor Products
Customer Groups *	[✓ NOT LOGGED IN] [✓ General] Wholesale Retailer QAAAA
Coupon *	Specific Coupon
Coupon Code *	CYBERDEAL
Uses per Coupon	
Uses per Customer	1
From Date	11/28/11
To Date	11/28/11
Priority	
Public In RSS Feed	Yes

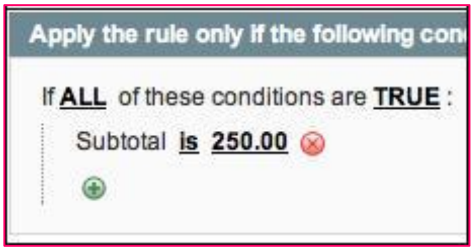
Let's talk a bit about some of the items on this screen, as there are differences between this and the one you completed for the previous Catalog Price Rule:

- You can use anything for the **Coupon Code**. However, you should use something that is easy to enter, yet unique enough not to be guessed (unless you don't mind that).
- If you want to restrict the number of times a coupon can be used, enter a value for **Uses per Coupon**. For instance, if you're only offering the discount to the first 100 people who use the coupon, enter **100**. Otherwise, leave blank if there is no limit.
- In many cases, you may want to limit the number of times a customer may use a coupon. In our example, we have limited the coupon use to once per customer by entering **1** for **Uses per Customer**.

Let's now create our conditions:

1. Click the **Conditions** menu tab.
2. Click the green plus sign. Choose **Subtotal**.
3. Click the ellipsis and enter **250.00**. Click outside the field.

Your condition should look as follows:



Apply the rule only if the following con

If ALL of these conditions are TRUE :

Subtotal is 250.00

Next, we will create the discount action:

1. Click the **Actions** menu tab.
2. Select **Percent of product price discount** for **Apply**.

> As with the Catalog Price Rules, the application methods for Shopping Cart Rules give you increased flexibility:
>
> 1. **Percent of product price discount** reduces the product price by the specified amount as a percentage of the product's price.
> 2. **Fixed amount discount** subtracts the amount from the product price, such as 10 dollars off.
> 3. **Fixed amount discount** for whole cart applies a fixed discount to the entire cart, not just selected products.
> 4. **Buy X get Y free (discount amount is Y)** really gives you a cool feature, allowing you to provide customers with a free product with the purchase of a qualifying product.

3. Enter **15** for the **Discount Amount**.
4. Select **Yes** for **Stop Further Rules Processing** if you want this to be the only discount applied to the shopping cart.

You'll also notice that on this screen you can also apply further rules specific to the products in the Shopping Cart. For instance, if you wanted this rule to only apply to Apple products, you could select Apple as a brand.

Also different from Catalog Price Rules, you can create a specific **Rule Label** for all Store Views, which you can also modify for specific Store Views. Click the **Labels** menu tab to enter what you want to appear to the customer after they enter their Coupon Code in their Shopping Cart.

> On the **General Information** screen, at the bottom, is the choice to publicize your discounts in your site's RSS feed. Unless you really don't want people to know of a special discount, using your RSS feed will help push your discounts out to search engines and customers who may be subscribing to your RSS feed. This is an easy way to increase awareness, and search sites love RSS feeds.

Click **Save Rule** when you're finished setting up your rule. Unlike Catalog Price Rules, you do not have to apply the rules in order to have them activated.

Newsletters

One way to keep customers reminded about your store is to ask them to subscribe to receive periodic e-mail newsletters from you. By informing your customers about new products, special discounts, and more, you can many times generate repeat sales, which is the least expensive cost-per-sale you can have.

Reduce your cost-per-sale

Unless you have an established brand, you will need to advertise to attract buyers to your store. From pay-per-click online ads (for example, Google AdWords) to offline print ads, there are many ways of buying customer attention. However, you should always do whatever you can to encourage buyers to return again and again to buy from your store. You should also encourage referrals. These sales don't cost you any more in terms of advertising expenses. This is one reason social networks such as Facebook and Twitter have become so popular; they're free (actually, you can purchase advertising on both now). The idea is that by getting Facebook and Twitter users to mention their purchases, their friends may be inclined to shop with you. Again, this is free advertising and a zero cost-per-sale.

Effective use of newsletters, however, requires a certain degree of specialized knowledge and expertise—more than you may have if you're new to e-commerce, and certainly more than we can distill within this book. With social networking, we're seeing some shifts in newsletters, as well. More customers are opting out of newsletters, instead subscribing to Twitter feeds and Facebook pages to receive promotional information. Newsletters that contain more in-depth information and articles remain quite active, though.

That said, you should not hesitate to use newsletters, even if they are only minimally requested by your customers. The cost of production is minimal and there is no cost of delivery. Any sales they generate will certainly be worth the effort!

Designing a newsletter template

The most challenging aspect of setting up to send newsletters is creating your newsletter template. Personally, I would suggest you consider buying a template design you find suitable for your brand (we like Theme Forest (http://www. themeforest.net) for design templates), or hiring a designer with specific newsletter experience. If you do want to take a stab at designing your own newsletter template, here's some key advice:

- Keep the width of your newsletter to no more than 600 pixels, as many e-mail programs have narrow viewing windows.

- While you can put some CSS styles in the header to control overall styles, any specific styles for things such as paragraphs, spans, and so on should be in-line. That is, within the HTML tag itself.
- E-mail programs have trouble interpreting <div> tags properly. Unlike building contemporary web pages, use tables to lay out your design.

> **Back to 1999**
>
> If you're a web designer, you may be cringing over these guidelines for e-mail layouts. I do, as well. However, e-mail client programs don't all have the rendering capabilities of today's browsers.
>
> The variety of tools people use to read e-mail also contributes to this dilemma. As a designer, you're already faced with browser compatibility among four or five primary browsers and versions. Imagine trying to debug a complex layout for hundreds of different e-mail viewing possibilities?
>
> I know it's not as fun as using stylesheets and <div> tags. Just consider it your opportunity to go down memory lane.

The use of templates for newsletters is a bit of a misnomer. The templates you create are not really used as you might imagine. You create a newsletter template in Magento to use for a particular e-mail event. If you wish to re-use the template for another mailing, but with some different content, you can save the original template as a new template.

Be that as it may, let's create a simple first newsletter template. Once you learn the basics of creating a Magento newsletter, you'll be prepared to apply more advanced design techniques or third-party newsletter design templates.

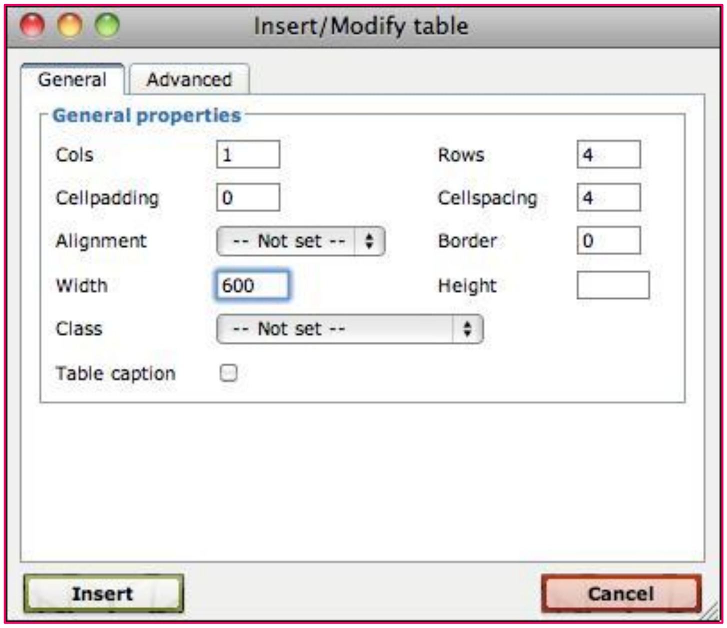
Let's put together a newsletter to send to customers announcing our Cyber Monday sale:

1. Go to **Newsletters** | **Newsletter Templates** in your administration backend.
2. Click on **Add New Template**.
3. For **Template Name**, enter **Cyber Monday**.
4. The **Template Subject** is what will appear as the subject of the e-mail your customers will receive. For this example, enter **Cyber Monday Sales Event!**
5. We'll leave **Sender Name** as is for now.
6. Enter your return e-mail address for the **Sender Email**. For our customers, we would enter **support@novusweb.com**.

7. In **Template Content**, I first add a couple of lines above the default content. What you see in this field, when you first create a new template, is so you'll know what Magento code to use to add an **Unsubscribe** link. This link allows recipients to unsubscribe from your newsletter list, and helps you meet anti-spam rules.

> Every country—and even some states in the US—have different anti-spam rules. If you're not familiar with the ones in your locale, do some research. That said, the main thing is to avoid having your server blacklisted or blocked by various **internet service providers** (**ISPs**). The best way to avoid blacklisting is to prevent complaints by recipients: only send to customers who have agreed to receive your newsletter, have a clear and easy to find unsubscribe link, and include your company name and address. A link to your privacy policy page would also be advised.

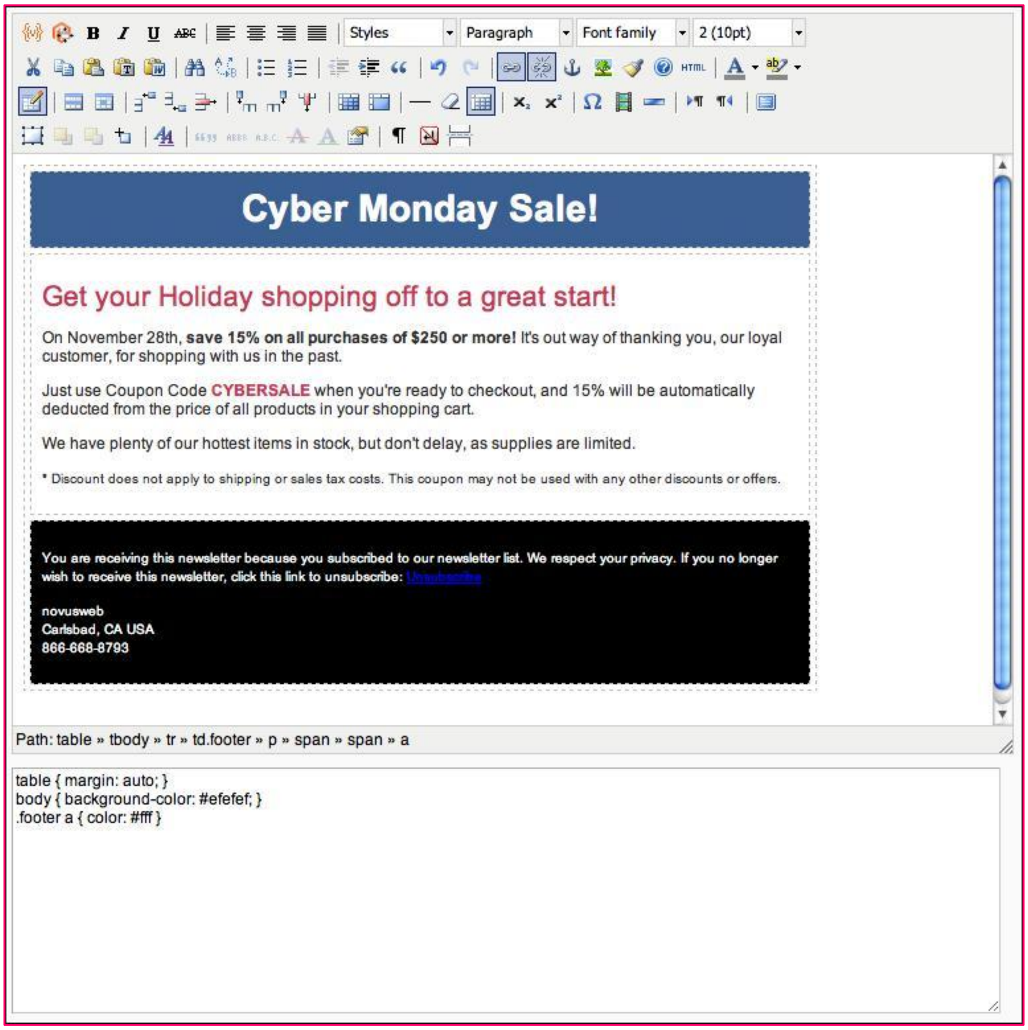
8. Next, I create a table 600 pixels wide, one column in width, that has three rows. This way, I can include a header, a main content area, and a footer. If I need to further subdivide any of these areas, I can always nest a table within a table cell. By clicking the **Inserts New Table** icon in the editor tool bar, a pop-up window, shown in the following screenshot, allows me to specify the table I want to insert:



Cols	1	Rows	4
Cellpadding	0	Cellspacing	4
Alignment	-- Not set --	Border	0
Width	600	Height	
Class	-- Not set --		
Table caption	□		

In the following screenshot, you can see the very simple newsletter design I created for this example. By using the tools in the editor, I styled the different table cells.

Additionally, as you can see in the bottom pane, I added some CSS styling that will be placed in the header of the e-mail. Remember, we want to use as little styling in the header as necessary.



Cyber Monday Sale!

Get your Holiday shopping off to a great start!

On November 28th, **save 15% on all purchases of $250 or more!** It's out way of thanking you, our loyal customer, for shopping with us in the past.

Just use Coupon Code **CYBERSALE** when you're ready to checkout, and 15% will be automatically deducted from the price of all products in your shopping cart.

We have plenty of our hottest items in stock, but don't delay, as supplies are limited.

* Discount does not apply to shipping or sales tax costs. This coupon may not be used with any other discounts or offers.

You are receiving this newsletter because you subscribed to our newsletter list. We respect your privacy. If you no longer wish to receive this newsletter, click this link to unsubscribe: Unsubscribe

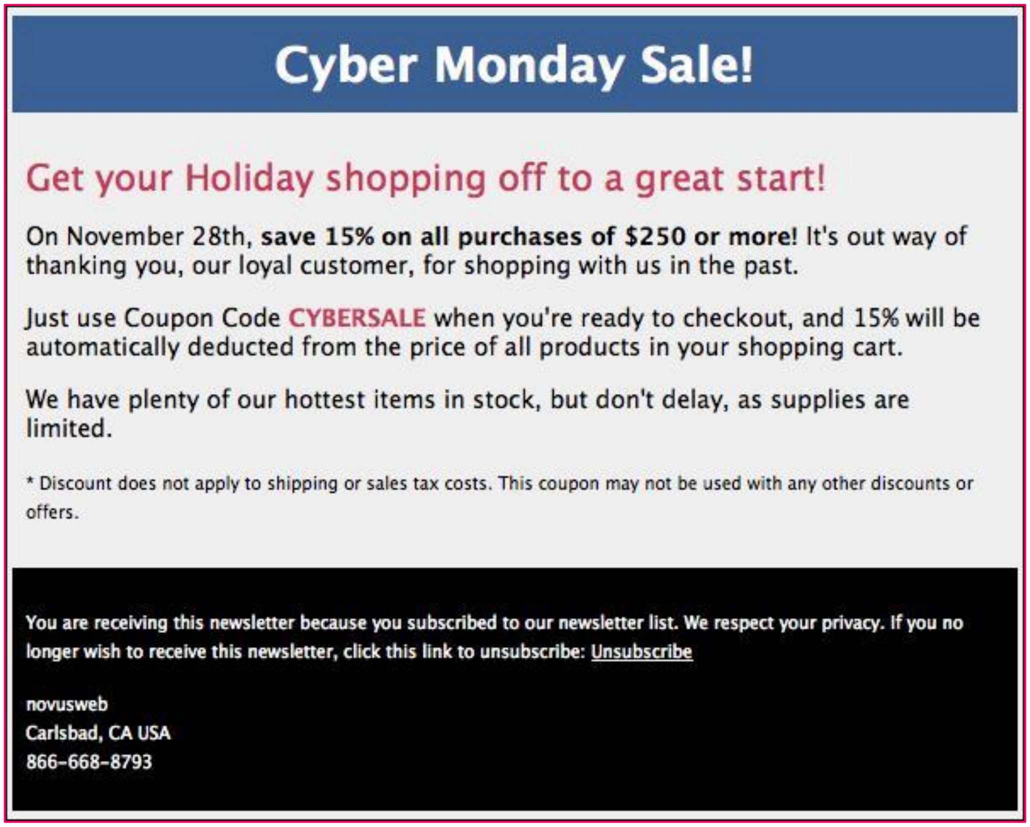
novusweb
Carlsbad, CA USA
866-668-8793

Path: table » tbody » tr » td.footer » p » span » span » a

```
table { margin: auto; }
body { background-color: #efefef; }
.footer a { color: #fff }
```

Notice the blue link in the footer area. While I have added a CSS statement—**.footer a { colour: #fff; }**—in the Template Styles pane, Magento won't render these styles until we preview the template.

To do that, click **Save Template** at the top of the screen. You'll be returned to the list of newsletter templates. To the right of your template, choose **Preview** in the **Action** drop-down menu. A new browser window will open, and you can see the fruits of your labor. In my case, the preview looks as follows:



Cyber Monday Sale!

Get your Holiday shopping off to a great start!

On November 28th, **save 15% on all purchases of $250 or more!** It's out way of thanking you, our loyal customer, for shopping with us in the past.

Just use Coupon Code CYBERSALE when you're ready to checkout, and 15% will be automatically deducted from the price of all products in your shopping cart.

We have plenty of our hottest items in stock, but don't delay, as supplies are limited.

* Discount does not apply to shipping or sales tax costs. This coupon may not be used with any other discounts or offers.

You are receiving this newsletter because you subscribed to our newsletter list. We respect your privacy. If you no longer wish to receive this newsletter, click this link to unsubscribe: Unsubscribe

novusweb
Carlsbad, CA USA
866-668-8793

This looks pretty much as we expected.

You can also insert Static Blocks (see *Chapter 6*, *Managing Non-product Content*) into the body of your newsletter using the following format: **{{block type="cms/block" block_id="block-id"}}**.

One more thing. To create the **Unsubscribe** link as shown in the footer, simply create a text link as you normally would using the built-in editor. For the link URL, enter **{{var subscriber.getUnsubscriptionLink()}}**. This will dynamically create a link your subscribers can click on to go to a special page on your site to unsubscribe.

Sending a newsletter

Once you have your template designed—and customers who have subscribed, of course—you can queue your newsletter to send. From the list of newsletter templates, select **Queue newsletter...** in the **Action** drop-down menu.

From this **Edit Newsletter** screen, you can set the start date for your newsletter, choose subscribers from any or all of your Stores, and make any last minute changes to the newsletter template. Once you've finished on this screen, click **Save Newsletter** to set your e-mail in motion.

After saving, you will be taken to the **Newsletter Queue** screen where you can view the status of your newsletter, including the start and finish time, the current status, how many e-mails have been processed, and how recipients are scheduled to receive your newsletter.

> Magento generally sends newsletter e-mails in batches—usually 15 at a time—so don't be surprised to see the number of processed e-mails increment in stages, rather than all at once.

Sitemaps

In the world of web development, there are two ways sitemaps are used:

- As a Table of Contents, of sorts, as a page on the site listing all publicly available pages
- An XML file stored on the server and used by search engines

The content of both is quite similar: the pages of your site. Yet, each are constructed a bit differently, since the first type is part of your public website and the latter is only seen by search engines.

The Magento sitemap

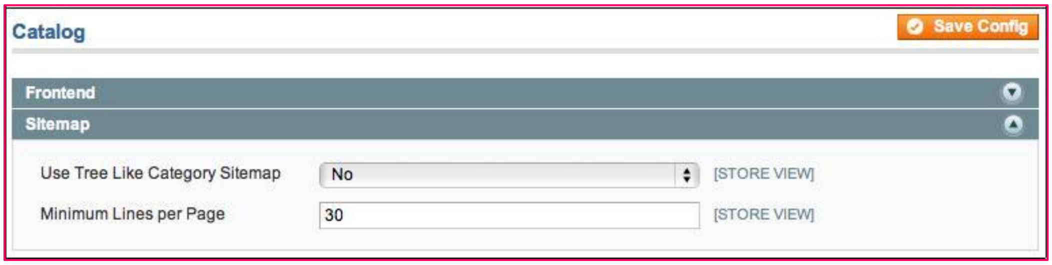
Magento automatically creates a public sitemap page for your site. However, unlike sitemaps created in platforms such as WordPress, the Magento sitemap merely lists the categories and products within your product catalog. In fact, there is little to configure (or possible to configure) with the Magento sitemap page.

To enable the sitemap page, follow these steps:

1. Go to **System** | **Configuration** | **Catalog**.
2. Expand the **Search Engine Optimizations** panel.
3. Select **Enable** for **Autogenerated Search Terms**.
4. Click on **Save Config**.

(We'll be addressing many of the other items in this panel in the later section on Search Engine Optimization.)

The configuration settings for the sitemap page are under **System** | **Configuration** | **Catalog**, in the **Sitemap** panel, as shown here:



Use Tree Like Category Sitemap: No [STORE VIEW]

Minimum Lines per Page: 30 [STORE VIEW]

As you can see, you can only configure whether you want category lists in a tree-like hierarchy, and what the minimum number of items you want listed on a single web page is.

There are several third-party extensions available for Magento that add additional layout functionality to the built-in sitemap page. See **Magento Connect** for the latest modules (http://www.magentocommerce.com/magento-connect/filter/all?query=sitemap).

The Google Sitemap

While Magento refers to the XML sitemap as the Google Sitemap, it really is a standard-format XML file that can be used by Yahoo!, Bing, and other search engines. These sites use the sitemap to understand more about the content of your site. While search engines regularly crawl websites by following the links on pages, the XML sitemap gives them direct links to all the content on your site.

Configuring the XML sitemap for your site takes several steps:

1. Configure the Google Sitemap settings.
2. Generate the sitemap files.
3. Submit the sitemap URL to the search engines.

Configuring the Google Sitemap

Under **System** | **Configuration** | **Google Sitemap** in your Magento backend are four panels visible if your **Current Configuration Scope** is **Default Config**. Each of the top three allow you to set how often the information on your site is updated, and its relative priority on a scale of 0 to 1.

For example, the default settings provide for a priority of 0.5, 1, and 0.25 for **Categories**, **Products**, and **CMS Pages**, respectively. If you are an e-commerce store, changes to your products should receive the highest priority; categories are secondary; and, lastly, the other content of your site—the CMS pages—are of the lowest priority.

> Unless you have specific needs for altering these default values, I would suggest you leave them as they are.

The fourth panel—only visible at the **Default Config** scope level— is called **Generation Settings**. Magento has the ability to automatically generate new sitemap files periodically according to the settings within this panel. However, configuring this panel alone does not actually mean your XML sitemap files will be automatically generated.

In *Chapter 10*, *Advanced Techniques*, you'll learn how to trigger cron jobs in Magento to periodically perform certain functions, including generating sitemaps.

So what if you can't configure a **crontab**? Is all lost? No. At least not as far as sitemaps are concerned. As we'll see in the next section, you can manually generate your XML sitemaps as often as you please. In fact, if you rarely update your site, say about every week or so, you really don't need to worry about automatically creating your sitemaps. Simply add the manual generation step to your updating activities.

Ask for help

As with any complex application, we sometimes find, from time to time, certain inadequacies with Magento that are at times frustrating. Sometimes they're even showstoppers. The management of scheduled tasks in Magento is certainly not a showstopper, but it is a bit frustrating.

When I started out with Magento, I assumed, as many do, that by configuring the **Generation Settings** in the backend, my sitemaps would be automatically generated. As we know now, that is not the case.

You'll periodically come across issues with Magento like this one that seem to be real head-scratchers. Books like this one are created to help resolve these questions. However, when it comes to crontabs, unless you're a real Unix hack—and especially if you're not hosting your Magento store on a Linux/Unix server—defer to the real techies that understand this issue: your hosting provider.

Like the man who refuses to stop and ask directions when driving, I hardly ever turn to my hosting provider unless the issue is behind what I'm focusing on: building, configuring, and delivering a quality Magento store for my client. Getting under the hood with Unix is one of those times.

If I can at least flag those areas which are outside the expertise of most of my readers, then I can help turn what might be a showstopper into a successful implementation.

Generating the sitemap files

Now that you have your configuration settings, the next step is to create the actual XML sitemap file:

1. Go to **Catalog** | **Google Sitemap** in your backend.
2. Click on **Add Sitemap**.
3. Enter a **Name** for your sitemap file. While it is assumed, from this screen, that any name can be used, in fact, only sitemap.xml is acceptable to Magento.

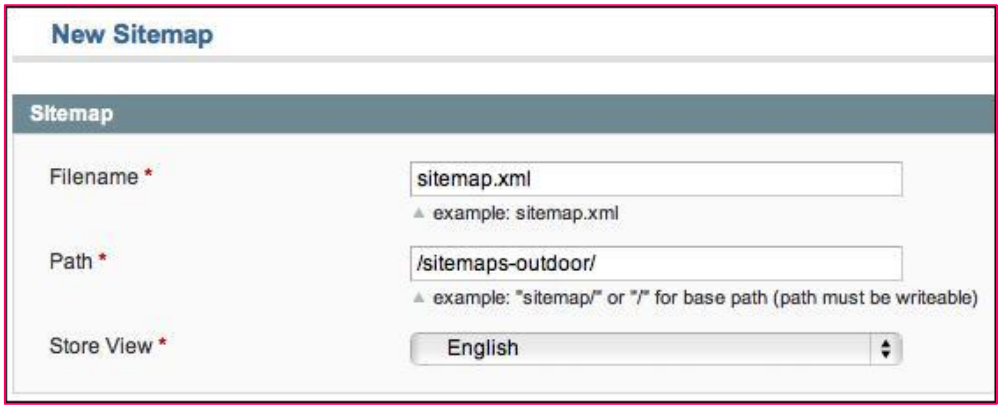
4. If you wish to store the sitemap files within a directory on your server, you can specify the **Path**. The directory must have write permissions.
5. Finally, choose the **Store View** for which the sitemap file applies.
6. Click on **Save and Generate**.

Generating sitemaps for multiple stores

Since Magento will only allow sitemap.xml as a filename, how can you generate multiple sitemap files for each **Store View**?

The answer is to create subdirectories for each store's sitemap.xml file. For example, if we want to create a sitemap for our Outdoor English-language store, we might create a subdirectory within our Magento installation called sitemaps-outdoor. By specifying this **Path** in our sitemap configuration, our sitemap file will be stored in that directory.

The following screenshot shows how I configured a sitemap for our Outdoor English language store:



New Sitemap

Sitemap

Filename * sitemap.xml
example: sitemap.xml

Path * /sitemaps-outdoor/
example: "sitemap/" or "/" for base path (path must be writeable)

Store View * English

Once you have the sitemap generated, you can manually generate the sitemap as needed by clicking on **Generate** on the row of the intended sitemap.

Submitting your sitemap

When you register your website with **Google Webmaster Tools** or **Bing/Yahoo! Webmaster Tools**, you will be asked to provide a sitemap for your site. Once your sitemap is generated, the Google Sitemap screen will show you the full path to your sitemap in the link for Google column.

Simply copy this full URL path and use it whenever a search engine asks for a sitemap file.

You can see what the search engines see by pasting this URL into your browser location bar. Depending on your browser, you may see the actual XML code or an interpretation of the code as an RSS feed.

Search engine optimization

The subject of **search engine optimization** can be, as you probably know, a very lengthy discussion worthy of several books. For our intentions, we want to go over some of the tools that are inherent in Magento to assist you with the process of adding search engine friendly attributes to your online store.

Magento wisely includes convenient fields so you can manage the meta keywords, meta description, page title, and URLs for the categories, products and CMS pages in your site. If you invest the time to manage these attributes, you'll find your site will be much better catalogued by the search engines.

Global SEO settings

The first place to start is to define the default settings for your page attributes at the website and store levels. If you don't individually enter values for these fields at the category, product, or CMS level, you still want some value to be included.

> While global settings are convenient, you should take the time to configure the individual page attributes whenever possible, as duplicate content across your site—multiple pages with the same SEO attributes—can work against you.

URLs

By default, a Magento URL looks like this:

http://www.sitedomain.com/index.php/furniture/living-room/ottoman. html

While this certainly works, it's better to shorten this URL when possible. By using the **URL Rewrites** configuration in Magento, our URL will now look like this:

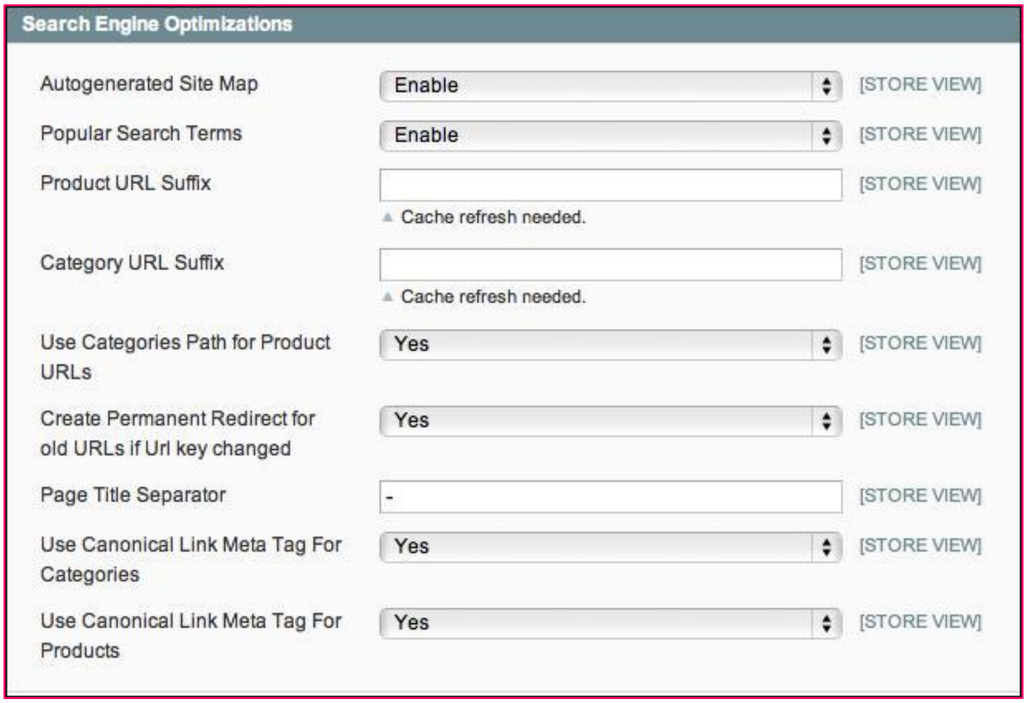
http://www.sitedomain.com/furniture/living-room/ottoman.html

1. Go to **System** | **Configuration** | **Web**.
2. Expand the **Search Engines Optimization** panel.
3. Select **Yes** for **Use Web Server Rewrites**.
4. Click on **Save Config**.

Now, let's go one step further. Let's remove the .html from the end of our URLs. This will even further shorten our URL and leave it nice and clean.

1. Go to **System** | **Configuration** | **Catalog**.
2. Expand the **Search Engine Optimizations** panel.
3. Remove .html from both **Product URL Suffix** and **Category URL Suffix**, leaving both fields blank.
4. In most cases, we do want to set **Use Categories Path for Product URLs** to **Yes**, as this automatically adds the category name to the URL, which could be good for search engine searches. However, if you have several layers of categories, or if your URLs will be more than 255 characters, consider setting this to **No**.
5. You should also select **Yes** for both **Use Canonical Link Meta Tag for Categories** and **Use Canonical Link Meta Tag for Products**. This adds a meta tag in your page header that helps prevent search engines from penalizing you for any pages that may have duplicate content on them.
6. Click on **Save Config**.
7. Go to **System** | **Cache Management** and flush your Magento cache.

The following is an example of what the **Search Engine Optimizations** panel should look like:



Search Engine Optimizations

Autogenerated Site Map	Enable	[STORE VIEW]
Popular Search Terms	Enable	[STORE VIEW]
Product URL Suffix	 ▲ Cache refresh needed.	[STORE VIEW]
Category URL Suffix	 ▲ Cache refresh needed.	[STORE VIEW]
Use Categories Path for Product URLs	Yes	[STORE VIEW]
Create Permanent Redirect for old URLs if Url key changed	Yes	[STORE VIEW]
Page Title Separator	-	[STORE VIEW]
Use Canonical Link Meta Tag For Categories	Yes	[STORE VIEW]
Use Canonical Link Meta Tag For Products	Yes	[STORE VIEW]

Default meta tags

To configure your default meta tags:

1. Go to **System** | **Configuration** | **Web**.
2. Expand the **HTML Head** panel.
3. If you have multiple websites or stores, and you want different default information for each, select the proper **Current Configuration Scope** in the upper-left part of your screen.
4. Enter a **Default Title** that will appear at the top of your visitors' browser window. This will be overridden by any specific choices made at the category or CMS page level. However, where these specific entries are not made, this title will appear. This is also the title that will appear on your home page.

5. The **Title Prefix** and **Title Suffix** fields allow you to automatically include words to appear before and after any specified page title. For example, if the page title for a product is *Sony VAIO Computer*, we could have Magento prepend *Buy* before the page title, and *at Bret's Electronics* after the title, thereby creating a page title of *Buy Sony VAIO Computer at Bret's Electronics*. Of course, the word *Buy* would appear even before a CMS page, such as *Buy About Us*. Therefore, experiment to find what words work best.

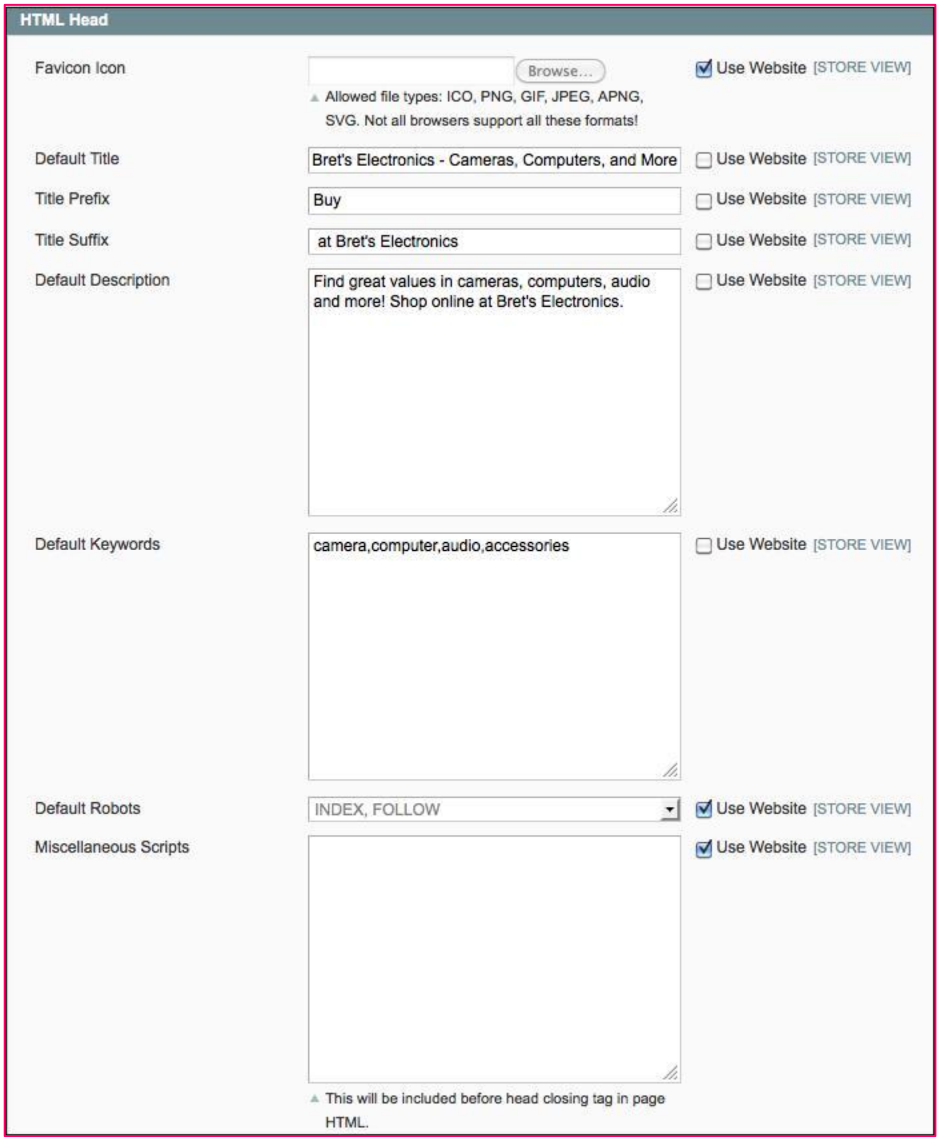
> Personally, I only use the **Title Suffix**, leaving the specified page title without any prefix.

6. Enter a **Default Description**. Again, this will also serve as the meta description for your home page.
7. Likewise for **Default Keywords**.

> **Meta keywords**
>
> While including keywords won't penalize you, major search engines generally ignore the meta keywords tag. Years ago, webmasters would load up this tag with lots of keywords, including words that had nothing to do with the actual page content, but were intended to draw online visitors. Long ago, search engines caught on to this keyword spamming technique. I personally don't worry about the meta keyword field.

8. If you want the search engines to index all the pages in your site (and really, what e-commerce store doesn't?), leave **Default Robots** set for **INDEX, FOLLOW**.
9. Click on **Save Config**.

Based on these instructions, here's a sample **HTML Head** panel:



HTML Head

Favicon Icon	Browse... Allowed file types: ICO, PNG, GIF, JPEG, APNG, SVG. Not all browsers support all these formats!	☑ Use Website [STORE VIEW]
Default Title	Bret's Electronics - Cameras, Computers, and More	□ Use Website [STORE VIEW]
Title Prefix	Buy	□ Use Website [STORE VIEW]
Title Suffix	at Bret's Electronics	□ Use Website [STORE VIEW]
Default Description	Find great values in cameras, computers, audio and more! Shop online at Bret's Electronics.	□ Use Website [STORE VIEW]
Default Keywords	camera,computer,audio,accessories	□ Use Website [STORE VIEW]
Default Robots	INDEX, FOLLOW	☑ Use Website [STORE VIEW]
Miscellaneous Scripts	This will be included before head closing tag in page HTML.	☑ Use Website [STORE VIEW]

Setting page-level attributes

Whether you're editing a category, a product, or a CMS page, you have specific control over the page title, meta description, meta keywords, and URL for that item.

As an example, let's review the process for managing the SEO attributes for a product:

1. Under **Catalog** | **Manage Products**, click on a listed product.
2. In the **General** panel, look at the **URL key** field. By default, Magento takes the name of the product, converts all characters to lowercase, and replaces spaces with hyphens. This value is what becomes the last part of the URL for this product. For instance, http://www.sitedomain.com/htc-touch-diamond is the URL for one of the Sample Data products where the URL key is set to *htc-touch-diamond*.

> You may feel the need to add or edit the URL key field. If you do so, you'll see a checkbox labelled **Create Permanent Redirect for old URL**. By checking this box, Magento will create an automatic redirect from the original URL for this product to the new one you have entered once you save. This is convenient if your product has already been indexed by the search engines, as you don't want the old link appearing in a search results page if clicking on the link results in a **404 Not Found** error for the visitor. With a redirect, clicking on the old link will take the visitor to the new URL automatically.

3. Next, click **Meta Information** on the left tab menu.
4. By default, the meta fields on this page are blank. You can enter whatever values you wish to here, and Magento will replace any default values with your entries.
5. Click on **Save** when finished and clear the Magento cache to see your changes.

Automatic product meta values

Earlier, we discussed how to set default values for the page title, meta description, and meta keywords. Guess what? Magento takes automation even further!

If you do not enter any values for these meta tags for an individual product, Magento automatically generates these values as follows:

- For **page title**, Magento uses a combination of the product name and the category hierarchy, separated by hyphens. For example, for the HTC Touch Diamond product, the page title is *HTC Touch Diamond - Cell Phones - Electronics*. If we have configured a **Title Prefix** and **Title Suffix**, as we did earlier, these are appended to this default title, rendering something like *Buy HTC Touch Diamond - Cell Phones - Electronics at Bret's Electronics*.

- The **meta description** is generated using whatever you have entered as the **Short Description** for the product.
- **Meta keywords** are simply the product name. Unfortunately, no commas are inserted between the words, but, as noted before, meta keywords are no longer an important SEO consideration.

For categories, Magento will automatically create a page title, but it will use the default meta description and keywords you specify in the system configuration unless you specifically add values under **Catalog** | **Manage Categories**.

Summary

When I began building websites in 1995, all we had to do was launch the site to get noticed, as there wasn't much online competition back then. Today, it's a very competitive landscape.

While your store marketing plans will certainly include many disciplines, such as pay-per-click advertising, social networking, and more, Magento helps you in terms of promoting your site and making it more search-engine friendly.

Together, we covered:

- Managing customer groups
- Promotion and coupons
- Designing and sending e-mail newsletters
- Generating sitemaps for visitors and search engines
- Configuring meta information useful for search engine optimization

At this point, your site is pretty much ready to go! If you've gone through the process of installing and configuring your Magento site to this point, you're ready to start taking orders. How exciting!

I would like to add at this point that while we have conquered configurations for Magento, much of the work in creating a successful online store lies in your marketing efforts. Simply publishing products online will not create sales; you need to consistently utilize the built-in tools of Magento—RSS feeds, newsletters, promotions, and so on—to publicize and promote. If you're the one in charge of creating sales, learn how to use the many other tools of the Internet, such as blogs, product search sites, and more, to bring attention to your store and its products. The technical work may be done, but the process of bringing customers to your shop is just beginning.

As you begin working with your store—and particularly if you're in charge of the technical aspects of the installation—you'll want to take Magento even further, to add on to its capabilities. In the next chapter, we will cover how to extend Magento.