9   Prices and Discounts

It would be easy to overlook the complexity of prices when you think about an eCommerce system. You may be tempted to say: "Each of my products has a price and that's it. The higher the price, the more money I make".

In real-world eCommerce systems, things are not so simple. There are many factors that influence a price. For example, you could introduce bulk prices for your products, where the price per product drops if you buy more of them at the same time. In addition, you could use a different pricing model during specific times of the year. This would enable you to increase the price during a high-demand seasonal period. You may also have different prices for different variants of your products, as you saw how to do in Chapter 5. Finally, you may want to introduce discounts that make products cheaper for specific groups of users or during a period of time (or both), or to grant a discount on an order above 200 euros.

All of these ways to influence the price are supported by Dynamicweb out of the box. For cases where you need even more control over the price of a product or an order, you can make use of Dynamicweb Extensibility and use hooks such as a Price Provider or a custom Sales Discount. The Extensibility options are outside the scope of this chapter, but you'll find more information on them in Chapters 15 and 17. In this chapter, the focus is on the built-in features to work with prices. In particular, I'll discuss the following topics:

●   A brief overview of configuring default prices for a product

●   How to configure bulk prices on products

●   How to set up and use Campaigns to change prices during specific periods of time

●   The different Sales Discounts that Dynamicweb provides and how to use them to set up complex, dynamic discount structures with little work

To get started, I'll begin with the easiest part: default prices.

9.1   Prices

In Dynamicweb, you set prices on products and variants in the administrative interface when creating or changing products. This means you define most of the prices at "design time" as opposed to run-time, which is the case for Sales Discounts as you'll see later. Exceptions to that rule are prices determined by a Price Provider (which is a part of the Dynamicweb Extensibility API, as you'll see in Chapter 17). You can define the following five types of prices:

●   Default Prices

●   Bulk prices

●   Campaign Prices

●   Variant Prices

- Product Unit Prices

Each of these is discussed in the following sections.

9.1.1 The Default Prices

You've already seen Default Prices at work in a number of earlier chapters of this book. You assign a default price to a product or a variant as shown in Figure 9-1:

Figure 9-1

These prices are called *default prices*, or *fallback prices*, as they are used as the price for a product or variant when no alternatives are available. As you'll see in the remaining sections of this chapter, the alternatives can come from a variety of sources such as bulk prices, variants, campaigns, units, or custom price providers.

Note that there's a clear distinction between a price and a sales discount. You should see a price as a property of a product, either fixed (such as the default price) or dynamic (a bulk price for example). Discounts, on the other hand, are values that are subtracted from the price when you order that product. This offers a few distinct differences. First, you can display the price and the discount separately. This means you can display something like "Normal price: 250 euro, but your special price is only 200 euro!". Second, discounts can be calculated dynamically based on factors such as the user, the time of year, the total amount of products in the cart, the total order amount and a number of other factors – as you'll see towards the end of this chapter.

For many cases, the default prices are all that you need. For other cases, read on, as I'll show you how to work with bulk prices, and prices for campaigns, variants and product units.

9.1.2 Bulk Prices

Bulk prices are a great way to persuade customers to buy more of your products. Why pay a dollar for a single pen, while you can have 10 for only 9 dollars? That's a 10% savings!!

To configure bulk prices in Dynamicweb, carry out the steps of the following walkthrough:

> You can combine prices for variants, campaigns, units and bulk prices. In addition, you configure different prices for each of the available currencies in your system. But the fact that you can, doesn't mean you should since all of these prices will quickly lead to a messy (and complex) price matrix. Later in this chapter you'll see a quick demonstration of combining all these price options together, so you can see why it should be avoided.

Walkthrough – Configuring Bulk Prices

In this walkthrough you'll create a new product and set up bulk prices. You'll then create a new page with the Product Catalog showing just this product so you can see exactly how the prices behave.

1. Log in to Dynamicweb and click the eCommerce button in the Main Menu.

2. Create a new product – called **Price Demo** for example – and give it a stock level and a default price of 100 euro. If you wish you can assign values to other properties such as images, but that's not required for this walkthrough.

3. Switch to the **Prices** category and from the list on the left choose **Bulk prices** and then click the right-facing arrow. Confirm the dialog that asks if you want to delete all prices. Since you haven't set up any prices yet, you're not actually deleting anything.

4. In the list on the right, click **Bulk prices**. A new textbox labeled **Bulk prices** appears. Enter **1;5;10** (the numbers are separated by semi-colons) and then click the **Save** (not

Save and close) button. The screen is reloaded and now shows the following price structure:

Figure 9-2

At the bottom of the screen, leave 100 euros set for the row with the value **1**, and enter **90** and **80** euro respectively for the two other values. The prices are used when the ordered quantity for this product is between the specified value (inclusive) and the value of the next level (exclusive). That means that if you buy four or fewer products, you pay 100 euros a piece. If you buy between five and nine products, they only cost you 90 euros apiece. If you buy ten or more, the price goes down to 80 euros.

5. Click **Save and close** to apply the changes.

6. Next, set up a new page called **Prices**. Add a new paragraph to it called **Prices** and then add a Product Catalog module. In the **Settings** category, choose **Product** and then from the Product picker select the **Price Demo** product you created earlier. You can leave settings at their default. Save all your changes and request the page in the browser.

7. Add the product to your cart and then request the Cart page. You should see something like Figure 9-3

Figure 9-3

The price of the order line is 115 euro; 100 euro for the product and an additional sales tax of 15. Your sales tax may be different, resulting in a different total price.

8. Click the green Plus (+) icon three times to increase the quantity to four. The order line price should go up to 460 euro (100 * 4 * 1.15).

9. Next, click the Plus icon once more. The price should now be 517,50, which consists of a product price of 450 euro (5 * 90) and a sales tax of 67.5 in my example

10. Keep clicking the Plus icon until the quantity has reached 10. The total price should now be 920 euros (10 * 80 euros + tax), which is actually cheaper than buying nine!

Price Demo	10	€920,00	+ – ✕
Paymentfee ()		€0,00	
Deliveryfee ()		€0,00	
Total		€920,00	

Empty Cart Next

Figure 9-4

Note: there is no built-in way to display the bulk prices in the Products template. Only when people start adding products to the cart and increase the quantity will they discover the new price. You could render the available prices using a `ProductTemplateExtender`. Creating your own `ProductTemplateExtender` is discussed in Chapter 16.

Besides bulk prices, campaigns are another great way to influence a product's price.

9.1.3 Campaign Prices

In Chapter 4 you learned that campaigns are basically simple periods of time that can either be active or not. In order for a campaign to be useful you somehow need to attach it to a product. In Chapter 4 you saw how to assign a campaign to a product using the Details tab in order to display the product during the timeframe set by the campaign only. In this chapter you'll see how to change the price of a product for the duration of a campaign.

Walkthrough – Changing Prices during a Campaign

In this walkthrough you'll reuse the **Summer Season** campaign you created in Chapter 4 to lower the product's price while the campaign is valid.

1. First, using the Management Center, make sure your campaign is active and valid (e.g. make sure today's date falls between the campaign's start and end date.). Refer to Chapter 4, section 4.10 – Campaigns if you're not sure how to do this.

2. Next, locate the **Price Demo** product you created earlier in the backend end and switch to the **Prices** tab.

3. Remove the **Bulk prices** from the list at the right by clicking it once and then clicking the Left-facing arrow between the two lists. Confirm the deletion by clicking **OK**.

4. Next, click **Campaigns** in the list on the left and click the Right arrow. If you don't see **Campaigns** in the list, you don't have any active campaigns. Click **Save** on the toolbar.

5. At the bottom of the screen, enter **90** as the price used during the **Summer Season** campaign and then click **Save and close**.

6. Locate your product in the frontend again and notice how the price has changed. If you want, you can set the campaign's end date to a date in the past to see the price change back to its initial, default price.

    Another way to change a product's price is by assigning different prices to variants, as discussed next.

9.1.4 Variant Prices

You can assign prices to different variants in two ways. First, you can open the Edit screen of a variant and change its default price. You saw how to do this in Chapter 5. The other option is similar to how you assign various prices to campaigns. Because it's so similar, I won't show you how to do this with a walkthrough, but just describe the steps.

1. On the **Prices** tab, move the **Variants** option from the left to the right list. Remove the other price differentiators from the list at the right (if any).

2. Save your changes so the screen reloads and shows the variants available for the current product.

3. Enter the prices for the available variants, as shown in Figure 9-5:

Figure 9-5

4. Click **Save and close** to apply the changes.

When you now request different variants of this product at the frontend, as you saw how to do in Chapter 5, you should see the price change to reflect the changes you made in the **Prices** tab.

Note that setting up the prices for variants like this doesn't change the actual price on the variant itself. If you're working with extended variants (where you've overridden properties of the main product), you could also change the price on the variant directly instead of here in the price matrix. I prefer to change the prices on the variant directly as it makes things more consistent: you only see a single price for each variant (in the variant's edit screen), and not two different ones (in the edit screen and in the price matrix).

The final way to change a price through the user interface is by introducing Product Units for your products.

9.1.5 Product Unit Prices

Product units – often simply referred to as units – define the units or quantity in which you sell and ship your products. Imagine that you're selling wine. If your site targets the consumer market, you may sell your wine by the bottle, or in a box with twelve bottles. If you also target businesses such as bars and restaurants, you may also sell them by the pallet – with 40 boxes totaling 480 bottles. You could create these three options as separate products in the backend of your shop, but that would mean duplicating a lot of data, such as the product description, images and so on. Dynamicweb makes offering these kinds of products very easy with Product Units.

Setting up a unit is a three-step process:

1. First, using the Management Center, you create the product units. These are just text labels, such as Box or Pallet.

2. On the Stock tab of a product, you select the applicable product units.

3. You then set up product units as a differentiator on the Prices tab of a product.

Once you've set up the units and prices, you can display them at the front end. In many respects, product units are similar to product variants. In fact, under the hood, product units *are* variants and they are stored in the same tables as variants are, with a special VariantGroupUnit marker column set to true. This means that in order to display the various product units in a product template, you can use the same tags you also use for variants. You'll see how to do this in the next walkthrough.

Walkthrough – Changing Prices for Product Units

In this walkthrough you'll set up a few product units and then alter your Price Demo product – pretending it is wine that is sold by the bottle, box and pallet. You'll then see how to present the various product units on the details page and provide links to add them to the cart.

1. Go to the Management Center and expand the path **eCommerce** | **Product catalog** | **Product units**.

2.   Click **New product unit** on the toolbar, enter **Single bottle** as the **Name** and click **Save and close**.

3.   Repeat the preceding step twice and add **Box** and **Pallet** as the name for the units.

4.   Click eCommerce in the Main Menu and locate your **Price Demo** product.

5.   Click the **Stock** button and tick all three checkboxes for the three units. In the drop-down at the top, choose **Single bottle** as the default unit. Click **Save** on the Ribbon bar. The page reloads and now enables you to enter Stock information for each unit (as well as a Weight and Volume, which you can leave to zero). Enter a value greater than zero for each three units:
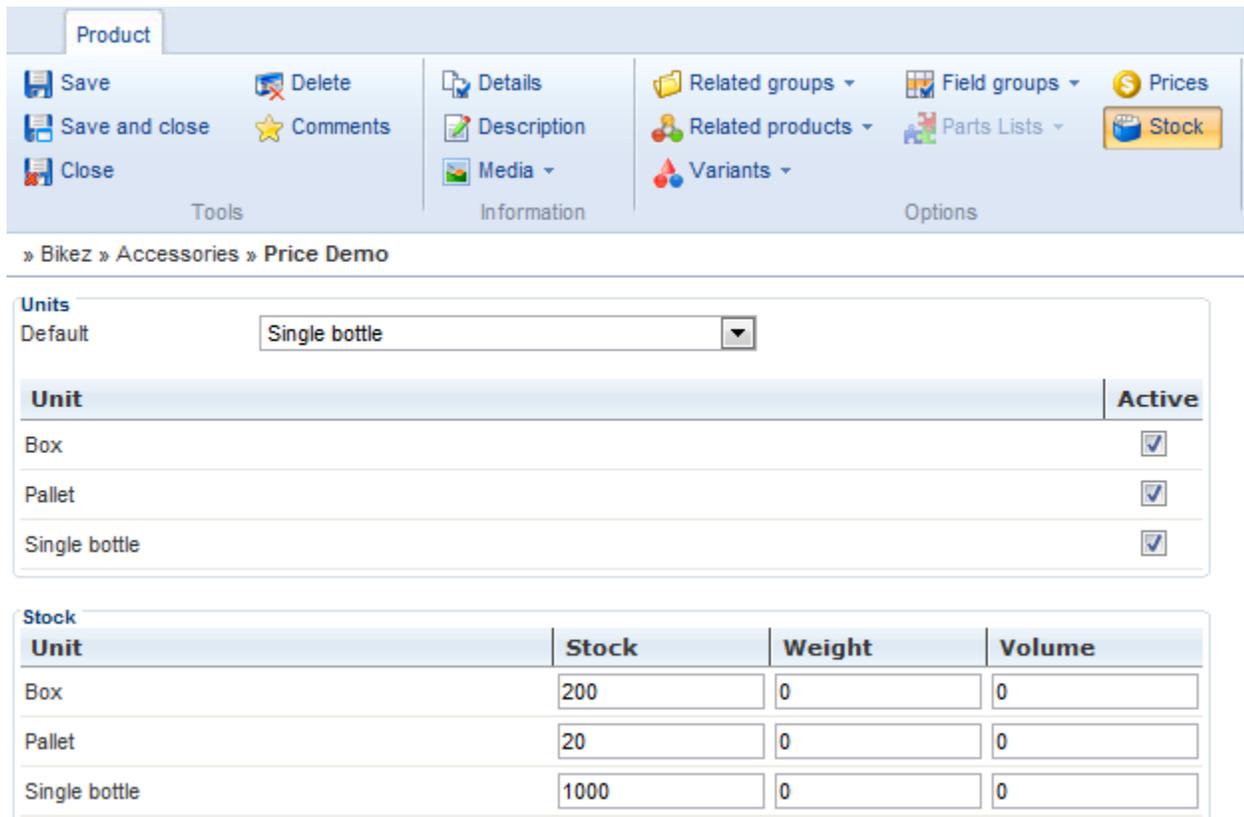
Figure 9-6

Click **Save** on the Ribbon bar to apply your changes.

6.   Next, click **Prices** on the Ribbon and remove any items from the list on the right. Then move **Units** from the left list to the right list and click **Save** on the Ribbon bar. The screen reloads and now enables you to enter prices for each unit. Complete the dialog as follows:

Figure 9-7

With these prices, the price for a single bottle is 4 euro. When bought by the box, the price drops to 3,50 a bottle and drops to 3 euro when you buy an entire pallet. Click **Save and close**.

7. Next, create a copy of the `Product.html` template and call it `ProductWithUnits.html`. You'll find the template in the folder `/Templates/eCom/Product`.

8. Open the template (using Dynamicweb's File Manager or Visual Studio) and add the following code to the end of the file, just before the Include tag for the comments template:

```
<!--@LoopStart(Units)-->
  <!--@HeaderStart-->
    <h2>Product units</h2>
    <ul>
  <!--@HeaderEnd-->
    <li><input type="radio" onclick="document.getElementById('UnitID').value = '<!--
    @Ecom:VariantOption.ID-->';" name="unitselector" value="<!--
    @Ecom:VariantOption.ID-->"
    <!--@If Defined(Ecom:VariantOption.Selected)-->checked="checked"<!--
    @EndIf(Ecom:VariantOption.Selected)--> /><!--@Ecom:VariantOption.Name--> - <!--
    @Ecom:VariantOption.Price--></li>
  <!--@FooterStart-->
      <ul>
  <!--@FooterEnd-->
<!--@LoopEnd(Units)-->
```

This code loops over the available units and outputs a link to add each item to the cart. As the link text, the unit name (such as Box) is used, followed by the unit's price.

9. Remove the Add to cart link, which you'll find in the template right above the short description (search for "Add to cart" to find the link). Since you're now going to add units, there's no point in adding the main product to the cart.

10. Open the Product Catalog module's settings on your Prices page, which you created earlier. Change the Product template from product.html to ProductWithUnits.html and save your changes.

11. Open your Prices page in the browser. You should now see something like this:

Product units

- ◎ Box - €42.00
- ◎ Pallet - €1,440.00
- ◎ Single bottle - €4.00
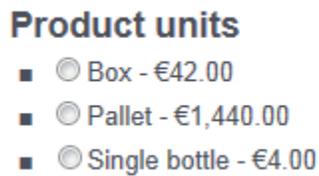
Figure 9-8

Just as with standard products, VAT has been added to the prices of the units.

12. Select the Box option to add the product to the cart and then request the cart page. You should see the Box item in the cart, with a price of 52.50 euro.

To give you an example of how complex the price matrix can be, consider Figure 9-9:
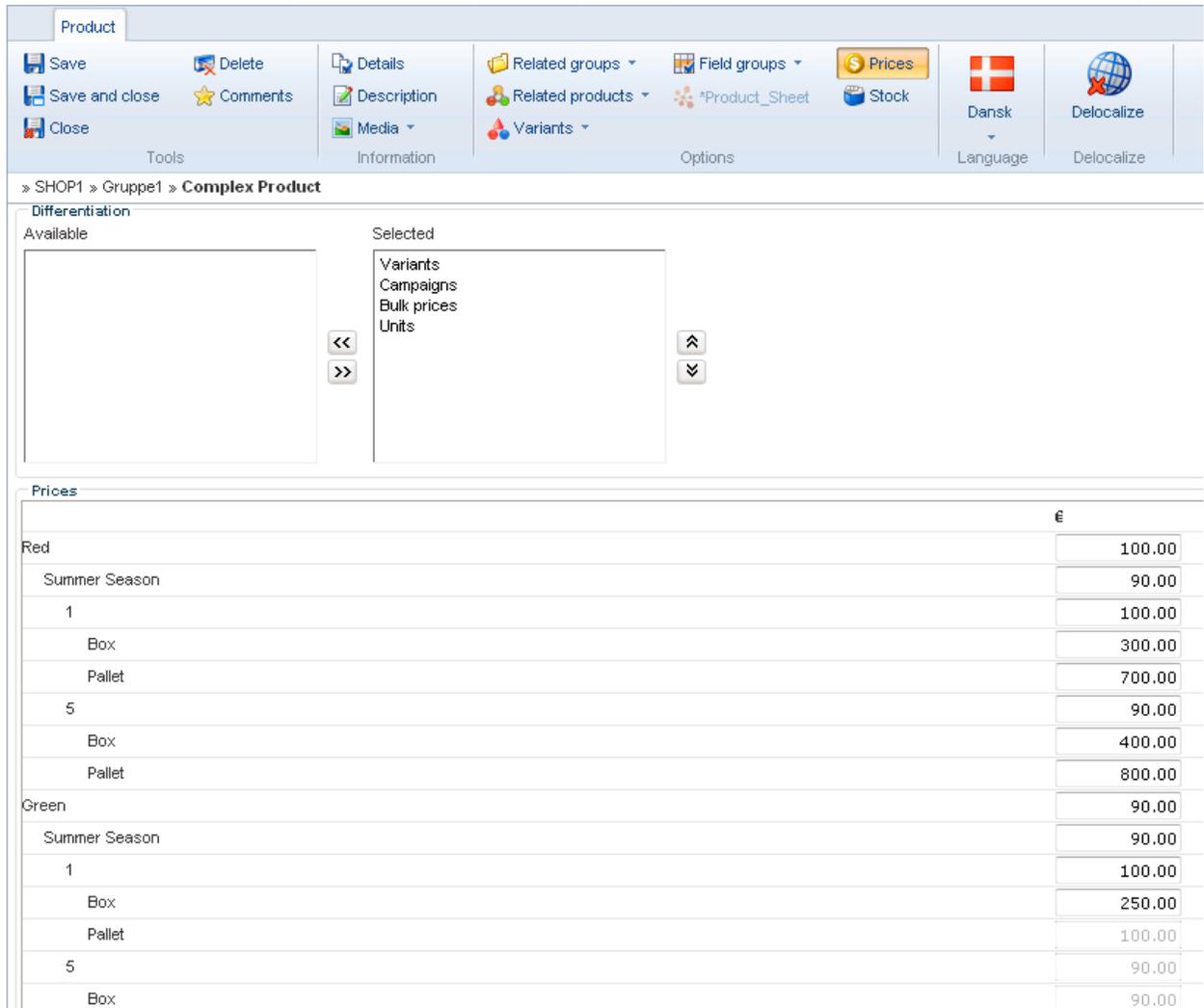
Figure 9-9

For this product, I enabled all available price options. You should read this from top to bottom, where each indented level makes the applied price more specific. For example, the Red variant costs 100 euro. During the Summer Season, the price drops to 90. If you buy them as bulk, say, five or more, the price changes again, and so it does when you buy a box or pallet instead of a single product. Add multiple currencies to this mix, and you'll be sure to end up with a confused and frustrated show manager. Try to avoid this complexity and limit the number of price levels to a maximum of two (e.g. variants during a campaign, or bulk prices for specific units).

Bulk Prices, Variants, Campaigns, and Product Units are the only built-in ways to influence prices in Dynamicweb using the backend administrative interface. In Chapter 17 you'll see how you can access even more ways to change prices when I discuss the Price Provider.

In the next section you'll see how to work with Sales Discounts.

9.2 Sales Discounts

The Sales Discounts module in Dynamicweb is an excellent way to set up a flexible pricing structure and reward your customers for their loyalty or sales volume. Dynamicweb ships with a number of different sales discounts, giving you fine control over when to grant a discount, and how much. The following table lists the available discount options:

Sales discount	Description
Product discount	Grants a discount of a fixed amount or a percentage when buying a specific product, or any product.
Product quantity discount	Grants a discount when you buy a minimum amount of a specific product or any product. The discount can be a fixed amount, a percentage or another product.
Order discount	Grants a discount when the total amount of the order matches some criteria such as greater than 1000 euro. The discount can be a fixed amount, a percentage or another product.
Order field discount	Grants a discount when an order field matches some criteria. The discount can be a fixed amount, a percentage or another product.
Order field plus shipping method discount	Similar to the Order field discount, but also enables you to limit the criteria to a specific shipping provider.
Product price-field discount	Enables you to take the price of a custom field instead of the main price. This way you can set a specific discount price on one or more products that are used when the discount is applicable.
Shipping method discount	Grants a discount when the customer selects a specific shipping provider. The discount can be a fixed amount, a percentage or another product.
Quantity discount	Grants a discount when the total number of products in the cart exceeds the specified amount. This discount applies to all products in the cart, and can't be used to limit the applicable products. If that's what you need, use the Product quantity discount instead. The discount can be a fixed amount, a percentage or another product.

Most of the Sales discounts are self-explanatory so I won't show you how to use each one of them in detail. Instead, in the next sections I'll give you an overview of how discounts work and then show one of them in detail in a walkthrough.

9.2.1 Understanding Discounts

Dynamicweb supports two types of discounts: Order line discounts that apply to one order line only, and Order discounts that apply to the complete order. In the list above, the Product discount and the Product quantity discounts are order line discounts and the rest are order discounts.

A discount consists of four main parts that you need to set up before you can use it:

● A name and a description to identify the discount.

- The rules that define the type of discount and when it is applied

- The actual discount you want to give (e.g. a fixed amount, a percentage or one or more free products)

- Additional factors that define when the discount is applicable, such as a period of time or an Extranet group membership.

Figure 9-10 shows the screen that you use to define a discount. The screen is divided into four areas – **General**, **Sales discount type**, **Sales discount value** and **Limitation** – that each map to the four areas I just described.

Figure 9-10

In the **General** area you enter a **Name** and a **Description**. The name is also shown in the cart for the end user and as such should be as descriptive as possible.

In the **Sales discount type** area you define the type of discount, which is one of the items from the table of types you saw earlier in this section. To see all the available options, click the **Additional sales discount types** link:

Figure 9-11

When you click a sales discount type on the left, the **Parameters** section to the right changes. Here you select the rules that affect the discount. In Figure 9-11 you can see that you can apply the **Product discount** to all products, or just to a selected number of products or groups. Figure 9-12 shows the settings for an **Order discount** that is applied when the total amount of all products in the cart is greater than 400 euro.

Figure 9-12

Each discount has different parameters that let you tailor the discount to match your requirements.

The next step is configuring the value of the discount. For most discounts, this can be one of the following three values:

Value	Description
Amount	This option gives a discount with a fixed amount, e.g., when the discount applies, you can get a discount of 10 euro.
Percentage	This option gives a discount as a percentage of the total order amount, e.g., when your cart contains products with a combined value of 500 euro and you get a 5% discount, the discount value is 25 euro.
Products	With this option you can give away one or more free products. For example, when you buy an item from the Mountain Bikes category, you could give away a free helmet.

The final part of a discount you need to configure is when the discount is valid, and optionally for whom. If you have the Extranet module available, you can define one or more user groups for whom the discount is applied. This enables you to build up customer loyalty. For example, you could grant the discount to all subscribers to your newsletter, helping you to persuade your customers to sign up. Figure 9-13 shows the available options:

Figure 9-13

With these settings, the discount applies to all users that place an order between the dates set in the **Date from** and **Date to** fields. To grant the discount to any user that is currently logged on, choose **Authenticated users**. To enable the discount for selected users only, choose **Selected users and groups**, and click the **Add user** or **Add group** icons to pick a user or a group from the popup that appears. You end up with your screen looking similar to Figure 9-14:

Figure 9-14

Note that it's typically easier to select groups instead of individual users. If you need to grant a discount to a few users, you may be better off assigning those users to a new

Extranet group and then applying the discount to that group. This makes managing the discount a lot easier.

Finally, make sure the **Active** checkbox (visible at the bottom of Figure 9-14) is checked. It's off by default when you create a new discount, but you need to turn it on in order for the discount to be active at all.

In the next walkthrough you'll create two discounts: one that applies to products, and one that applies to the entire order. In a later section you'll see how these discounts appear at the frontend and how you can change Dynamicweb's behavior of calculating discounts when two or more discounts are applicable at the same time.

Walkthrough – Creating Discounts

1. Open the Management Center and expand the path **eCommerce** | **Product catalog** | **Sales discounts**.

2. Click **New sales discount** on the toolbar and enter **Quantity discount** as the **Name**. You can leave the **Description** empty.

3. Click **Additional sales discount types** to expand the list and then click **Quantity discount**.

4. In the **Parameters** section enter **2** for the **Product count**.

5. Under the **Sales discount value** header, enter **10** in the **Amount** field. This grants a 10 euro discount if you buy two or more products at the same time.

6. Check off the **Active** checkbox at the end of the page and then click **Save and close** to save your discount.

7. Click **New sales discount** again and now enter **Order discount** as the **Name**.

8. Under the **Sales discount type** header, click **Order discount** and then set the **Order buy limit** to **500** euro.

9. Under the **Sales discount value** header, select the radio button for the **Percentage** option and enter **5** in the **Percentage** field. This grants a five percent discount to the entire order. Check off the **Active** checkbox at the bottom of the screen. Click **Save and close**.

You now have two discounts that theoretically could be applicable at the same time. In the next section you'll see how these discounts behave at the frontend and what you can do to tell Dynamicweb to apply only a single discount when more than one active discount exists.

9.2.2 Displaying Discounts at the Frontend

Since discounts are added as order lines to your cart, you don't have to do anything for them to show up in the user interface if you're using the default templates. However, you may need to alter their appearance; such as adding an icon, or changing the colors used in the discount rows. In addition, you may want to prevent Dynamicweb from granting multiple discounts at the same time, as that may cost you too much money. In the following walkthrough you'll see how to make this work.

Walkthrough – Displaying Discounts

1. Open up one of your pages with the Product Catalog module on it in your browser and add a random product to your cart.

2. Go to the cart page and notice how your **Quantity discount** has not been added to the cart because you only added a single product. Neither do you see the **Order discount** as the total amount: the order hasn't reached the threshold set for that discount, as shown in Figure 9-15:

Price Demo	1	€115,00	+	−	×
Paymentfee ()		€0,00			
Deliveryfee ()		€0,00			
Total		€115,00			

Empty Cart Next

Figure 9-15

Click the green Plus (+) icon to increase the quantity of the product (or add other products to your cart) An order line for the **Quantity discount** is added to the cart with a value of 11,50; the 10 euro you set plus VAT.

3. Click the Plus icon again until the total order amount is greater than the threshold you set for the Order discount. Once you reach that value, another discount is added to the cart:

Price Demo	7	€805,00	+	−	×
Quantity discount	1	€-11,50			
Order discount	1	€-40,25			
Paymentfee ()		€0,00			
Deliveryfee ()		€0,00			
Total		€753,25			

Empty Cart Next

Figure 9-16

Notice how the **Order discount** has been calculated over the total order amount.

4. If you only want to grant one discount at a time, go into Management Center and expand the path **eCommerce** | **Advanced configuration** | **Sales discounts**. The default value for the **Choose discount** drop-down list is **Aggregated,** which means multiple discounts can be added to the cart at once. If you choose **Highest** or **Lowest** instead, Dynamicweb calculates the value of each discount and only adds the one that matches your selection. To see how this behaves, choose **Highest** and click **Save** to apply the changes. Then refresh the Cart page in your browser (or visit it again from the site menu). The **Quantity discount** is now gone, and only the **Order discount** remains. Note that it could be the other way around in your case, depending on the price of the product and the values you supplied for the two discounts.

5. Next, open the file that is used to display the cart. By default, this is `ShowCart.html` from the folder `/Templates/eCom7/CartV2/Step`. The `OrderLines` loop (defined

as `<!--@LoopStart(OrderLines)-->` and `<!--@LoopEnd(OrderLines)-->` contains both the products and the discounts. From within this loop you can use the `<!--@If Defined(Ecom:Order:OrderLine.IsProduct)-->` to test if the order line item is a product or discount. For example, look at the way the increment, decrement and delete links are hidden for lines that are not products (e.g. discounts):

```
<!--@If Defined(Ecom:Order:OrderLine.IsProduct)-->
  <a class="orderfunctions" href="<!--@Ecom:Order:OrderLine.IncrementLink-->">...</a>
  <a class="orderfunctions" href="<!--@Ecom:Order:OrderLine.DecrementLink-->">...</a>
  <a class="orderfunctions" href="<!--@Ecom:Order:OrderLine.DeleteLink-->">...</a>
<!--@EndIf(Ecom:Order:OrderLine.IsProduct)-->
```

In addition to the `IsProduct` template tag, there's another way to distinguish between a product and a discount and that is by looking at the `Ecom:Order:OrderLine.Type` tag. The value of this tag can be any of the following four values:

Value	Name	Description
0	Product	The value 0 is used for products.
1	Discount	This value is used for discounts such as the Quantity and Order discounts you set up earlier
2	Fixed	Fixed should be treated as internal to Dynamicweb and is used in a custom solution to override the price set by an integration module or other custom code.
3	ProductDiscount	The value 3 is used for discounts that have another product as its value.

You can use an `@If` tag on these values to determine the type of discount that is being rendered. The following example changes the background color of the discount lines using an `If` tag. Note that using inline CSS styles for this is not really recommended. In your own site you'd be better off using a CSS class. However, for this demo it's enough to show you how to use this. First, change the opening `<tr>` tag inside the `OrderLines` loop to the following:

```
<tr class="orderline<!--@Ecom:Order:OrderLine.Type--> orderline"
<!--@If(Ecom:Order:OrderLine.Type==1)--> style="background-color: LightGrey;"
<!--@EndIf-->>
```

Then refresh the cart page in your browser. You should see something like this:

Price Demo	7	€805,00	+ − ✕
Quantity discount	1	€-11,50	
Order discount	1	€-40,25	
Paymentfee ()		€0,00	
Deliveryfee ()		€0,00	
Total		€753,25	

Empty Cart [Next]

Figure 9-17

I admit that this is not the most aesthetically pleasing shopping cart presentation, but hopefully you can see how to make use of the available tags so that you can use them in your own website (albeit with a better looking design).

For more information about the available tags for the Shopping Cart v2 module, check out the Template tags section of the Dynamicweb site, which you can access as follows: http://templates.dynamicweb-cms.com/eCommerce/Dynamicweb-eCommerce-template-tags/Shopping-Cart-V2/Order/Cart.aspx. There you'll you find a complete list of the available tags and the loops you can use, such as the `OrderLines` loop.

9.2.3 Viewing Discounts at the Backend

When a customer has submitted an order that contains a discount, you can view its details in the Dynamicweb backend, just as you would view standard order lines for products. The discount shown in Figure 9-18 ends up as follows in the backend:

Number	Name	Quantity	Unit price	Total price
	Order discount	1	-37.50	€-37.50
10001	Images Demo	3	250.00	€750.00
	Total purchase price excluding sales tax			€570.00
	Payment : (No Pay (not for Holland))			€17.13
	Shipping			€0.00
	Sub-total excluding sales tax			€587.13
	Sales tax			€142.50
	Total price including sales tax			**€729.63**

Figure 9-18

This way you can always see how much discount has been granted for an order.

9.2.4 Using Sales Discounts as Fees

One of the questions I am often asked is why Dynamicweb doesn't have a separate Fee module. For example, imagine you're selling car tires. When you sell a set of four tires you may want to charge an additional fee to mount the tires under the customer's car. Customers often tell me that a Fee module would be an excellent addition to the Dynamicweb eCommerce environment to implement this solution. The good news is: Dynamicweb already has a Fee module, and it's called the Sales discounts module. Just think of fees instead of discounts, and enter a negative amount or percentage when defining the discount. Then Dynamicweb treats your discount as a fee, and adds the value to the cart's total, instead of subtracting it. Figure 9-19 shows a quick example of a cart that contains a set of four tires and a mounting fee:

Product name	Amount	Price			
Set of four Michelin Pil...1		€612.50	+	÷	✕
Mounting fee	1	€31.25			
Paymentfee ()		€0.00			
Deliveryfee ()		€0.00			
Total		€643.75			

Empty Cart Next

Figure 9-19

9.3 Summary

This chapter gave you an in depth look at Dynamicweb's pricing and discount options. The chapter started off with a discussion of the various ways to change a price, including default prices, prices for variants, product units, and campaigns and bulk prices.

The ability to define prices at such a granular level is very powerful, but may also lead to eCommerce systems that are hard to maintain. Try to limit yourself to using only one or two price influences for a single product; definitely don't use all five of them at once.

The second part of this chapter discussed the available Sales Discounts that ship out-of-the-box with Dynamicweb. Using the Sales Discounts module you can grant discounts to your customers based on a number of criteria such as the products ordered, the number of products ordered, the total value of the cart, the customer, the time of year and more. In case you want to implement a discount that's not supported by Dynamicweb, it's easy to create your own, as you'll learn in Chapter 17.

In the next chapter you'll see more about managing orders in the backend.