Business Gateway

HTML Redirect Integration Guide

V5.2 September 2015

Use this guide to:

- Integrate your website with Worldpay
- Create and test HTML Redirect orders
- Look up ISO codes, payment method codes, and more

Contents

1 Introduction

Welcome to the HTML Redirect Integration Guide.

This guide describes how to integrate your website with our payment gateway using the HTML Redirect model. This is the simplest and easiest method of integration, and works on just about any platform.

This guide shows you how to:

- Send us orders in HTML
- Redirect a shopper from your website to the Worldpay payment pages
- Test your integration with Worldpay
- Customise your integration – for example, by giving the payment pages the look and feel of your own website, and creating your own customer messages
- Make sure your integration is safe and secure

For details of the revision history of this guide, see ***Changes to this Guide on page 57****.*

1.1 Why use HTML Redirect?

HTML Redirect ensures that:

- Your integration with Worldpay’s payment gateway is faster, simpler and cheaper
- Your shoppers make their payment on Worldpay’s secure payment pages, not your website

Collecting and storing shoppers’ payment details and payment method on your own website can be time-consuming and costly. Instead, if you use HTML Redirect, shoppers are redirected to our payment pages, and their orders and payments are processed by Worldpay. You have full daily reporting facilities, and we keep both you and your shopper updated at every stage.

1.2 Who is this guide for?

This is a technical integration guide, aimed at:

- Customers who are managing their own integration
- Other technical roles, including system integrators who are involved in designing your integration

1.2.1 Skills and knowledge

To integrate with us using HTML Redirect, you need:

- A basic knowledge of the Worldpay payment services
- A working knowledge of HTML, for a simple integration
- Web development expertise, including server-side scripting. This could be useful, for example, if you wanted to include an automated ordering system or bespoke shopping cart functionality in your integration

1.3 More help?

For more information about Worldpay’s products and services, including payment methods, see the Worldpay website at **http://www.worldpay.com**

For all our technical and operational guides, see **http://www.worldpay.com/support/bg/**

If you have a question our guides can't help you with, or you'd just like to speak to us directly, contact us at:

- Email: **support@worldpay.com**
- Phone: +44 (0)870 3661233

1.4 Legal

This document and its content are proprietary to Worldpay and may not be reproduced, published or resold. The information is provided on an “AS IS” basis for information purposes only and Worldpay makes no warranties of any kind including in relation to the content or suitability. Terms and Conditions apply to all our services.

Worldpay (UK) Limited (Company No: 07316500/ FCA No: 530923), Worldpay Limited (Company No: 03424752 / FCA No: 504504), Worldpay AP Limited (Company No: 5593466 / FCA No: 502597). Registered Office: The Walbrook Building, 25 Walbrook, London EC4N 8AF and authorised by the Financial Conduct Authority under the Payment Service Regulations 2009 for the provision of payment services. Worldpay (UK) Limited is authorised and regulated by the Financial Conduct Authority for consumer credit activities.

Worldpay, the logo and any associated brand names are all trade marks of the Worldpay group of companies.

2 Overview

In this chapter, we'll tell you what HTML Redirect is, and how you can use it to integrate your online business with Worldpay.

HTML Redirect is suitable for Internet shop environments, call centres, reservation centres, and multi-channel sales situations. It enables you to process payments in real-time, and ensures you have access to the maximum number of ways to make payment online.

The HTML Redirect service is secure, and provides Worldpay with the information needed to perform active fraud risk assessment. Also, it is the fastest way to get up and running with on-line payments. Payment takes place on our pages, so you don't have the costly burden of having to gather shopper data yourself. You just send the shopper to us to complete their purchase.

HTML Redirect is a simple and safe method of integration, and works with almost all payment systems.

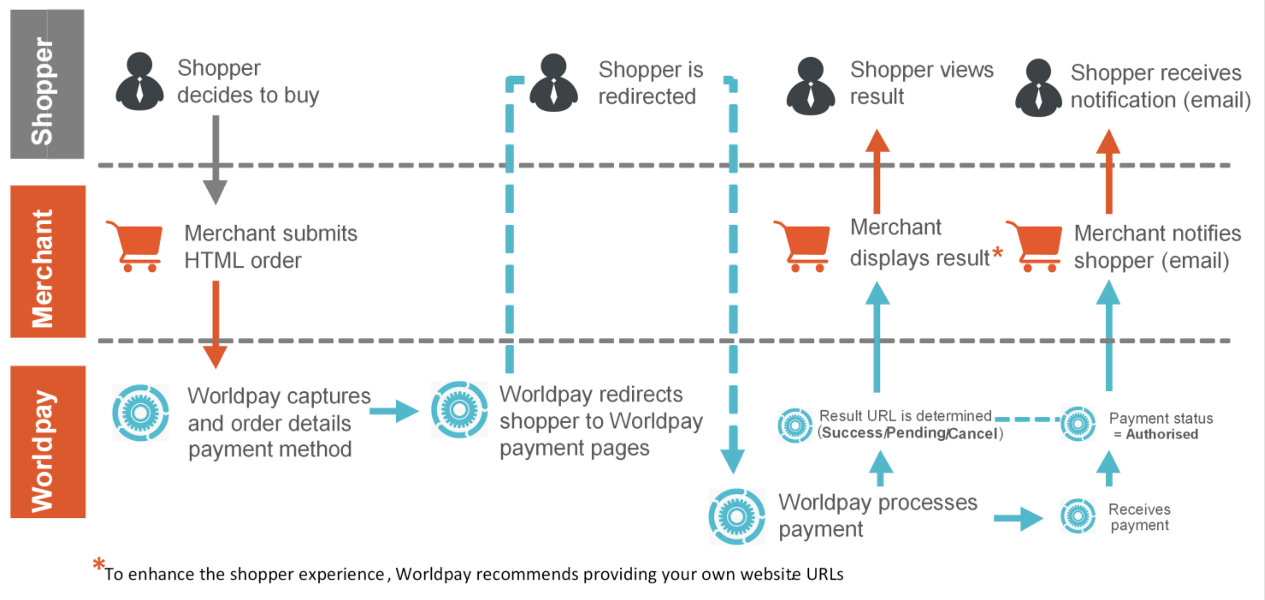

Shopper
Merchant
Worldpay
Shopper decides to buy
Shopper is redirected
Shopper views result
Shopper receives notification (email)
Merchant submits HTML order
Merchant displays result*
Merchant notifies shopper (email)
Worldpay captures and order details payment method
Worldpay redirects shopper to Worldpay payment pages
Result URL is determined (Success/Pending/Cancel)
Payment status = Authorised
Worldpay processes payment
Receives payment
*To enhance the shopper experience, Worldpay recommends providing your own website URLs

Figure 1: HTML system flow

2.1 How HTML Redirect works

The HTML Redirect service enables your website with its own online shopping system to work in conjunction with our payment service. This guide helps you create your own online shop, and get started.

HTML Redirect uses HTML forms to pass information to the Worldpay payment gateway for payment processing. You create the form in your web page and insert parameters such as currency that describe the purchase – this form is known as an order details form.

This is how HTML Redirect works:

1. When a shopper is ready to pay for their goods, your website should present them with a button or link which, when selected, submits the order details to our payment page.
2. The shopper is then taken to our payment pages to enter their payment details, such as credit/debit card details. Your website does not gather card details from the shopper – we do this in our payment pages.

3. If the shopper has chosen to pay by credit/debit card:
    a. We forward the shopper's details to the bank, where the shopper's credit worthiness is checked
    b. The bank returns an authorised or declined response to us

       If the payment is declined, the shopper is given two options – to try another means of payment or to cancel the purchase.
4. We then display a result page to the shopper showing the outcome of the payment transaction. We also send them a confirmation email. For more information, see chapter ***Telling your shopper about payment results* on page 30**.
5. We also inform you about the transaction. How this is done depends upon how you have configured your options with the Merchant Interface, but at the very least we send you an email. For more information, see chapter ***Telling you about payment results* on page 28**.

Have a look at our range of example internet shop websites, which we have built for you to use as the basis for your own. There are several examples of very simple sites, as well as more detailed custom-made sites. To view the example sites, click this link: **Example sites**.

2.2 Before starting the integration

2.2.1 Initial checks

Check that everything you need is in place before you start. For example:

- Are your systems capable of taking orders and submitting them to us?
- Have you tested your integration out?
- Have you been set up on the Worldpay system?
- Have you received your Worldpay Administration Code? (We send you this as soon as your application is provisionally accepted.) This is not the same as either your **Installation ID** or your **Merchant Code**, both of which are described shortly. The Administration Code is a unique code that identifies a Worldpay merchant. It is a reference under which all the merchant codes that a merchant has with the Worldpay payment service are stored.
- Have you set up appropriate usernames and passwords for yourself and your staff to log on to the system?
- Are you aware of the PCI-DSS standard on card-handling security? One feature of HTML Redirect is that Worldpay is responsible for the collection, storage and processing of cardholder data. This helps to reduce your costs, as otherwise you would be responsible for implementing the security measures needed for full PCI DSS compliance. For details, see **https://www.pcisecuritystandards.org/.**

If you have any questions about setting up your system, email the Worldpay Support team at ***support@worldpay.com****.*

2.2.2 Getting your installation ID from Worldpay

When you submit order details to us, you must include your Installation ID, which is a unique reference number we assign to you. It tells us which payment methods and currencies your installation supports.

To get your Installation ID:

- Go to **Merchant Interface > Installations**

You can use your Installation ID in both the live and test modes.

The Installation ID is not the same as the Worldpay Administration Code, which we send to you as soon as your application is provisionally accepted.

2.2.3 Deciding on your merchant codes

Something you need to consider before your account goes live is how many merchant codes you will need.

The Merchant Code is a unique identifier for a particular merchant account within the Worldpay payment service, although it is possible to have more than one merchant code, as explained below.

As a general rule, we open one merchant code (account) per currency set that you process, so if you trade in GBP and Euros, for example, you would have two merchant codes. However, this may not be enough for the kind of trading you do – you may need to consider using preferred merchant codes if you have a number of merchant codes with identical characteristics but where they are used for different purposes.

For instance, you may have a merchant code for software sales and another for hardware sales – if so, order details submitted to us for software need to specify the software merchant code, and order details submitted for hardware need to specify the hardware merchant code.

You can discuss this with us when your application is being processed.

For details of how you use the merchant code in HTML, see ***Specifying the merchant code* on page 20**.

2.2.4 Using different currencies

When you apply for a Worldpay merchant code (account), you get at least one free currency. You can add more currencies by paying a one-off fee for each currency. Or you may want to use currency combinations, where the acquisition and settlement currencies are different. For details, please email the Worldpay Support team at **support@worldpay.com**.

Exchange rates

If you are dealing with a number of currencies on your website you may want to ensure that you are using up-to-date exchange rates. The exchange rates we use are updated every 24 hours at midnight GMT. These rates are then guaranteed for the next 24 hours for all transactions through Worldpay.

2.2.5 When you register

When you register with Worldpay you will be given two separate URLs for each method of payment and currency set you will be using. One will be the URL for the Test Environment and the other will be the URL for the Production Environment.

2.2.6 The importance of well-formed HTML

When you submit an HTML order it must be well-formed, using the correct HTML syntax. If it isn't, it may not get sent. See **http://www.w3schools.com/html/html_intro.asp** for more information about HTML and how to write it.

You must open and close the elements with the correct tags, and all attribute values must be quoted. Also, make sure all nested code is correctly resolved, that is, the hierarchies are correct. It may help if you check your HTML code for accuracy by using a validator like this one:

http://jigsaw.w3.org/css-validator/#validate_by_input

2.2.7 Using iframes with our payment pages

Worldpay does not support the use of iframes on our payment pages. However, we are aware that some merchants prefer to use them anyway. If you decide to use an iframe to host the Worldpay payment pages then you need to:

- *Tell the Worldpay support team of your decision.*
- *Be aware that you are taking on security liability for using iframes. You must use appropriate technology to secure the iframes so that the host page does not interfere with either the content or the events on the Worldpay payment pages.*

2.2.8 How you can trade online safely

We take the risk of fraud on the internet very seriously. Our payment system uses state of the art encryption techniques, and supports the industry leading anti-fraud systems provided by both MasterCard and Visa.

In addition we have developed our own anti-fraud software which provides you with a sophisticated detection system designed to identify potentially fraudulent transactions.

To find out what you can do to reduce the risk of fraud in your online trading, see **Fighting Fraud**. For more information about security and fraud, see chapter ***Enhancing security with MD5* on page 46**.

2.3 Integrating your website shop with our payment service

You can now begin the process of integrating your website shop with our payment service.

Integration is very straightforward. You'll know when you have integrated successfully – when you can send us the order details of a purchase from your website and get an email response from us, you have integrated your website.

To integrate with Worldpay, what you need to do is:

- Gather order details from a shopper. What this means is:
  - The shopper selects products in their trolley
  - You collect the details and send them on to us
  - The shopper is redirected to our payment pages
  - We take their payment details
- In order to do that, you need to establish communications between our server and your website – basically this just means sending a test HTML transaction to us and getting an email response

As soon as you have integrated, you can then activate your installation and, subject to some checks, you can go live.

2.3.1 Choosing how to integrate

Depending on your level of HTML knowledge, you can opt for a very simple integration solution, or choose a more advanced option:

Integration option	Description	Further details
Storebuilder	This option provides you with a ready-made 'shopping cart' that is already integrated with HTML Redirect.	Section 3.1
Quick and simple	If you are familiar with HTML you can choose to set up a simple integration using examples provided in this guide, and by making use of our standard payment pages.	Section 3.2
More advanced	If you have advanced HTML skills you can develop your own fully automated ordering system, complete with customised payment pages.	Section 3.3
URL query string	You don't have to integrate using HTML forms; instead, you can submit order details in a single URL query string.	Section 3.4

Table 1: Choosing how to integrate

2.3.2 View example sites to base your integration on

If you want to see a range of integration examples before building your own, or if you just need some pointers about how to pass the information you gather to us, click **Example sites**.

This site has links to a number of model shops that we have built to help you, as well as the full HTML code which creates them, and which you can use as the basis for your own site. There are several examples of each of the three options below:

- Basic sites – plain, non-customised versions of the Worldpay payment pages
- Fully integrated sites – examples are a site with a full shopping cart with an editable basket, and another selling 20+ items on one page.
- Recurring payment sites – two examples of sites where payment is made over a period, perhaps for a magazine subscription. Payments are either on a rolling or a time-limited basis.

We cannot accept responsibility for the smooth running of any sites derived from the code on these example pages. The shops exist merely as examples and are not supported Worldpay products.

2.3.3 Checking the list of your installations

Go to Merchant Interface > Installations to see a list of your installations. To configure installation options, click the spanner button for the installation you are interested in—select the Test column if you are setting up for the first time.

There are Test and Production columns for each installation listed. If you are setting-up for the first time you should initially select the Test column, which enables you to develop and test every feature of your website pages before moving to a real production environment.

For general information about how the Merchant Interface works, see the **Merchant Interface Guide**. For information about customising your payment pages, see either the **Customising Guide (Standard)** or the **Customising Guide (Advanced)**.

2.4 Customising your installation to suit you

You can make our payment pages look like your own shop, even if you base your installation on one of the 'off-the-shelf' options shown in section ***View example sites to base your integration on* on the previous page**.

You can customise HTML Redirect to suit yourself by configuring various options. For instance, you can:

- Upload your own versions of the payment pages that are displayed to your shoppers, giving them the style, look and feel of your own website
- Decide what payment methods and currencies your shoppers can use
- Write your own messages to be displayed to shoppers during the payment process
- Customise the results pages and confirmation emails that you and your shoppers receive
- Specify that you want to use the ***Enabling and using Payment Response messages* on page 34** feature which, for example, enables you to send your own result page when your shopper has completed a transaction

For a more sophisticated level of customisation, you will need to modify the files used to build the Payment Service. To do this you should have expertise with HTML, and you will need to see the **Customising Guide (Advanced)**.

Although specific configuration tasks are described in other sections in this guide, you should note that you configure your installation within the Merchant Interface, which is our web-based financial administration tool for merchants using the Worldpay payment service.

2.4.1 Customising your payment pages

Once you have submitted order details to our payment service via your website and we have received them, we present a series of payment pages to your shopper. These pages gather the payment information needed to process the transaction. For an illustration of this, see ***Overview* on page 8**.

These pages can be customised in many ways to suit your website style and presentation. For comprehensive information on customising your payment pages, see the **Customising Guide (Advanced)**.

To customise payment pages

- Click the Edit Payment Pages button— an easy-to-use wizard helps you control how the payment pages are displayed. For more information see ***Display parameters* on page 38**.

If you have enabled Payment Response messages, you can control how these pages are displayed by using parameters in your submission. Payment Response enables you to automate and control your website. For more information, see ***Enabling and using Payment Response messages* on page 34**.

We gather credit/debit card details in these payment pages. This means there is no need for you to do this in your website, although you do collect and send us shopper details. We then collect the payment details, and process the rest of the transaction.

Currency selector box

This is where you choose the currency to be used for a particular payment from a shopper (if you accept payment in more than one currency).

If some of your chosen currencies are missing from the Currency Selector Box in the payment page, please check that you are using the correct **Installation ID** for the transaction. If this is correct, then your currencies may not have been added to our system; in this case, please email the Worldpay Support team at **support@worldpay.com**.

2.5 Testing your installation

You can test your site by carrying out test transactions in the Test Environment. This can be very useful when you are developing your website, as it enables you to try out your pages with actual live debit and credit cards without transferring funds.

This enables you to get a realistic sense of the whole payment processing procedure via Worldpay, without any money ever changing hands.

You can access the test environment from the Merchant Interface, our online management tool.

For more information about testing your installation, see the **Test and Go Live Guide**.

*You can also test your live merchant code (account) - this is different to running in the Test Environment. For more information see the **Test and Go Live Guide**.*

While you are testing, there is no risk of accidentally triggering a live transaction until you and Worldpay have agreed that you are ready to go live.

If you try to submit a live payment before you go live, the payment is refused and you receive an error message telling you that there are no appropriate accounts to process your payment.

2.6 Steps to going live

When you have finished developing your site and integrating it with Worldpay, you’re ready to go live, which means that you can accept and process real, live orders from customers.

These are the steps to going live:

1. Complete your testing, by sending us successful test transactions.
2. Request that we make your installation live. You do this by filling in the online Installation Activation Details page, and sending us the URL of the site you want to make live, and some other details. See the **Test and Go Live Guide** for further details.
3. We confirm your activation request by email.
4. We set your site live within two working days, providing we are satisfied with your technical integration.

For further details of going live, see ***Going live* on page 17**.

3 Integrating your website with Worldpay

Integration is very straightforward – when you can successfully send us details of an order and get an email response from us, you have integrated your website.

There are four ways to integrate your website with us, detailed below.

3.1 Use a Storebuilder package for simplicity and speed

The easiest way to get your website shop up and running is to use an off-the-shelf storebuilder package that is already integrated with HTML Redirect. These packages provide you with a ready-made 'shopping cart', which is a means for your customers to select goods from your store and then pay for them. For a list of these, see **Shopping carts**.

If you are taking this route, see the **Test and Go Live Guide** for information about testing your website and going live with it.

3.2 Use our ready-made HTML to integrate quickly and easily

If you are familiar with HTML, you can use our ***An example HTML form* on the next page** as the basis for your site.

What you need:

- A working Internet connection and your **Installation ID**, which is a unique reference number we assign to you
- Some knowledge of HTML. If you have this, there are two options:
  - Follow the steps outlined below in section **3.2.1. Setting up the HTML form quickly**, describing how to set up an integration using the simple ***An example HTML form* on the next page**.
  - Or if you want to try out a complete, working, ready-to-go example of an integrated HTML page, with an order details form already included, see ***A complete HTML page* on page 25**.

3.2.1 Setting up the HTML form quickly

This uses an HTML submission form with a small number of <input> tags containing parameters, which is used as a template in your own web pages.

This is the quick set-up:

1. Copy and paste the ***An example HTML form* on the next page** shown below into your website page advertising the product you wish to sell. The form holds the details of the order, and sends this information to us using its POST method, which is secure and encrypted.
2. Update the parameter values and elements to suit the product details in your website page. The parameters included in this example are mandatory, and must be included in the form. See section ***Mandatory details* on page 19** for more information.

Only update the parameter values, not the parameters themselves. If you decide to rewrite the parameters, take care with spelling – they must be exactly as shown below – and note that they are case-sensitive: INSTID is not the same as instId.

3. View the page in your browser, and select the 'Buy This' button. You are taken to our payment service where you can follow the progress of the transaction through the payment pages. The test version of the HTML example is set up for the Test Environment, so you can try it out without worrying about transferring real money.

4. You'll know your site is integrated if you can successfully get to the payment pages, and you can receive an email about the test purchase. We send you email notifications of each transaction automatically, including Test Environment transactions. You can use the example order details form as the basis for all your website pages or you can develop them further from this point in your own style.

Single and multiple items

A single order details form must describe a single total purchase rather than a series of items in a shopping cart. If you want to sell multiple items, you must summarise the total purchase amount and the description of the goods on your own site before passing the summarised details through to Worldpay.

When you are happy with your site in Test, you can move to the live / production environment. See ***Going live* on the next page** for details.

3.2.2 An example HTML form

The comments inside the <!-- and --> symbols can be included in your HTML page; the browser ignores them.

```
<!-- The first line of code creates a form which uses the POST method, and its action is to
send the form to us.
You do not need to set up a special connection to us before using it - your Internet
connection is all you need to communicate with us.-->

<form action="https://secure.worldpay.com/wcc/purchase" method=POST>

<!-- This next line contains a mandatory parameter. Put your Installation ID inside the
quotes after value= -->
<input type="hidden" name="instId" value="Your installation ID ">

<!-- Another mandatory parameter. Put your own reference identifier for the item purchased
inside the quotes after value= -->
<input type="hidden" name="cartId" value="Your ID for the product ">

<!-- Another mandatory parameter. Put the total cost of the item inside the quotes after
value= -->
<input type="hidden" name="amount" value="The cost of the product ">

<!-- Another mandatory parameter. Put the code for the purchase currency inside the quotes
after value= -->
<input type="hidden" name="currency" value="currency code e.g. GBP, USD ">

<!-- This creates the button. When it is selected in the browser, the form submits the
purchase details to us. -->
<input type=submit value=" Buy This ">

</form>
```

Code example 1: Basic order details form

3.2.3 A test version of the order details form

This version of the basic order details form also has a simple structure. It uses the URL for our test environment and includes the testMode parameter.

*Because this is a test transaction no money ever changes hands. For details of using test submissions see the **Test and Go Live Guide**.*

```
<!-- The first line of code specifies the URL for our test environment.

<form action="https://secure-test.worldpay.com/wcc/purchase" method=POST>

<!-- This next line contains the testMode parameter - it specifies that the submission is a
test submission -->
<input type="hidden" name="testMode" value="100">

<!-- This next line contains a mandatory parameter. Put your Installation ID inside the
quotes after value= -->
<input type="hidden" name="instId" value="Your installation ID ">

<!-- Another mandatory parameter. Put your own reference identifier for the item purchased
inside the quotes after value= -->
<input type="hidden" name="cartId" value="Your ID for the product ">

<!-- Another mandatory parameter. Put the total cost of the item inside the quotes after
value= -->
<input type="hidden" name="amount" value="The cost of the product ">

<!-- Another mandatory parameter. Put the code for the purchase currency inside the quotes
after value= -->
<input type="hidden" name="currency" value="currency code e.g. GBP, USD ">

<!-- This creates the button. When it is selected in the browser, the form submits the
purchase details to us. -->
<input type=submit value=" Buy This ">

</form>
```

Code example 2: Test version of a basic order details form

*You can specify the type of response you want from our system for a test submission by entering REFUSED, AUTHORISED, ERROR or CAPTURED as the value in the name parameter. You can also generate an AUTHORISED response by using a real name such as J. Bloggs. For more information, see the **Test and Go Live Guide**.*

3.2.4 Going live

When you have completed the development of your pages, and have tested them, you are ready to go live, which means that you can accept and process real, live orders from customers. You can make the ***An example HTML form* on the previous page** suitable for the live Production Environment by changing its URL and removing the testMode parameter.

For more information, see the **Test and Go Live Guide**.

3.3 Develop your own custom system

If you have advanced HTML skills, you can develop your own fully automated ordering system, complete with customised payment pages. But keep in mind that your order details submissions must contain the four mandatory parameters as a basic minimum. See section ***Mandatory details* on the next page** for details.

You can customise our payment pages so that they fit in with the look and feel of your own website—see ***Customising your installation to suit you* on page 13**.

You can choose to have transaction details returned to your server and, if required, display your own transaction confirmation message to your shoppers, using a feature called Payment Responses. For more information, see ***Enabling and using Payment Response messages* on page 34**.

3.4 Submitting a URL order details string

You do not have to use an HTML form to create and submit order details – you can use a URL query string. For example, you can carry out integration by submitting this single URL string:

```
https://secure-test.worldpay.com/wcc/purchase?instId=123456&cartId=WorldPay+Test
&amount=40.00&currency=GBP&desc=WorldPay+Test&testMode=100
```

Code example 3: URL order details string

3.4.1 Advantages and disadvantages

The advantages are that nearly all browsers support URL strings, and they are simple to code and to use. However, because the recipient can see all the attributes and values in the URL string, you may not feel it is secure.

This is an important difference when compared with using Worldpay's ***An example HTML form* on page 16**, which are encrypted and sent by the secure POST method.

3.4.2 URL string content

As is the case with form-based submissions, the URL order details string must contain the four mandatory parameters, as shown in the example. See section ***Mandatory details* on the next page** for more information.

The URL string shown above is for the Test Environment, and the testMode parameter is specified as 100 – this causes our system to treat the submission as a test submission.

If you specify any other value in the name field, such as a real shopper's name, the system responds with an AUTHORISED test result.

The URL string may also contain a range of optional parameters. For more information about parameters, see section ***HTML Redirect parameters* on page 35**.

3.5 Order details

When your shopper is ready to pay for their goods, your website submits the order details to our payment page. We then present the payment pages to the shopper for payment.

Your order details submission must contain at least the four mandatory parameters shown in ***Mandatory details* on the next page**. It's perfectly acceptable if that's all it contains. In that case, the payment pages open up to the shopper with empty fields – the shopper then has to enter details, such as name and address, directly into the payment pages.

3.5.1 Including optional details in your order

Alternatively, you could include other optional details in your order details submission. For example, you might gather name and address information in your own website pages, and include this information in your order details submission. We can then pass this information on to the payment pages, so that it is included when the pages are displayed to the shopper. For more information, see section ***Managing shoppers' billing address details*** **on page 21**.

Including Recurring Payments in your order

Your submission could also contain **Recurring Payments** (also known as FuturePay) details. Recurring payments are where the shopper pays for a purchase – for example, a magazine subscription – over, say, a year, with recurring monthly payments.

Including custom display settings in your order

You can also control the appearance of the payment pages by including custom display settings. See ***Display parameters*** **on page 38** for details.

You cannot collect card details on your website. One feature of HTML Redirect is that Worldpay is responsible for the collection, storage and processing of cardholder data. This helps to reduce your costs, as otherwise you would be responsible for implementing the security measures needed for ***full PCI DSS compliance****.*

3.5.2 Mandatory and optional details

Order details can be mandatory, in respect of either the order details submission or the payment pages.

For example, you can optionally include the shopper's address in the order details submission page, but if you do not include the address, the shopper must in any case provide this information on the payment page. See section ***Optional parameters*** **on page 36** for full details of optional details, and whether or not they are mandatory in the payment pages.

3.5.3 Mandatory details

The following four parameters must be included in the order details you submit to us:

Parameter	Definition	Example
instId	Your Worldpay Installation ID. This is a unique reference number we assign to you. It tells us which payment methods and currencies your installation supports.	1234567
cartId	A reference you assign to help you identify the purchase.	101KT0098
amount	The total cost of the purchase.	25.35
currency	The purchase currency, see ***ISO currency codes*** **on page 50**.	GBP

Table 2: Mandatory parameters

All parameter names are case-sensitive.

3.5.4 Optional details

You may include some optional details when you submit an order, such as a shopper's billing address details. But note that although they are optional in an order details submission, some of them are mandatory in the payment pages, as described in ***Mandatory and optional details* on the previous page**. This means that they can be left out of your order details message, but will have to be included in the payment pages in order for the transaction to be processed.

You can also include Recurring Payments (FuturePay) parameters in an order details submission. For more information, see ***Recurring payments (FuturePay) parameters* on page 42**.

3.5.5 Your own custom fields

You can highly customise the payment pages, the result pages displayed after a transaction has taken place, and the confirmation emails that you and your shoppers receive. This is done using the Payment Page Editor, which is the simplest and fastest way to customise your payment page. You access the Payment Page Editor via the Merchant Interface. For details of the fields you can customise, see ***Custom parameters* on page 43**.

If you want, you can carry out more detailed customisation – for details, see the **Customising Guide (Advanced)**.

3.6 Specifying the merchant code

In the HTML order details form you submit to us, you specify the merchant code to be used by using the parameter `accId1`. For example:

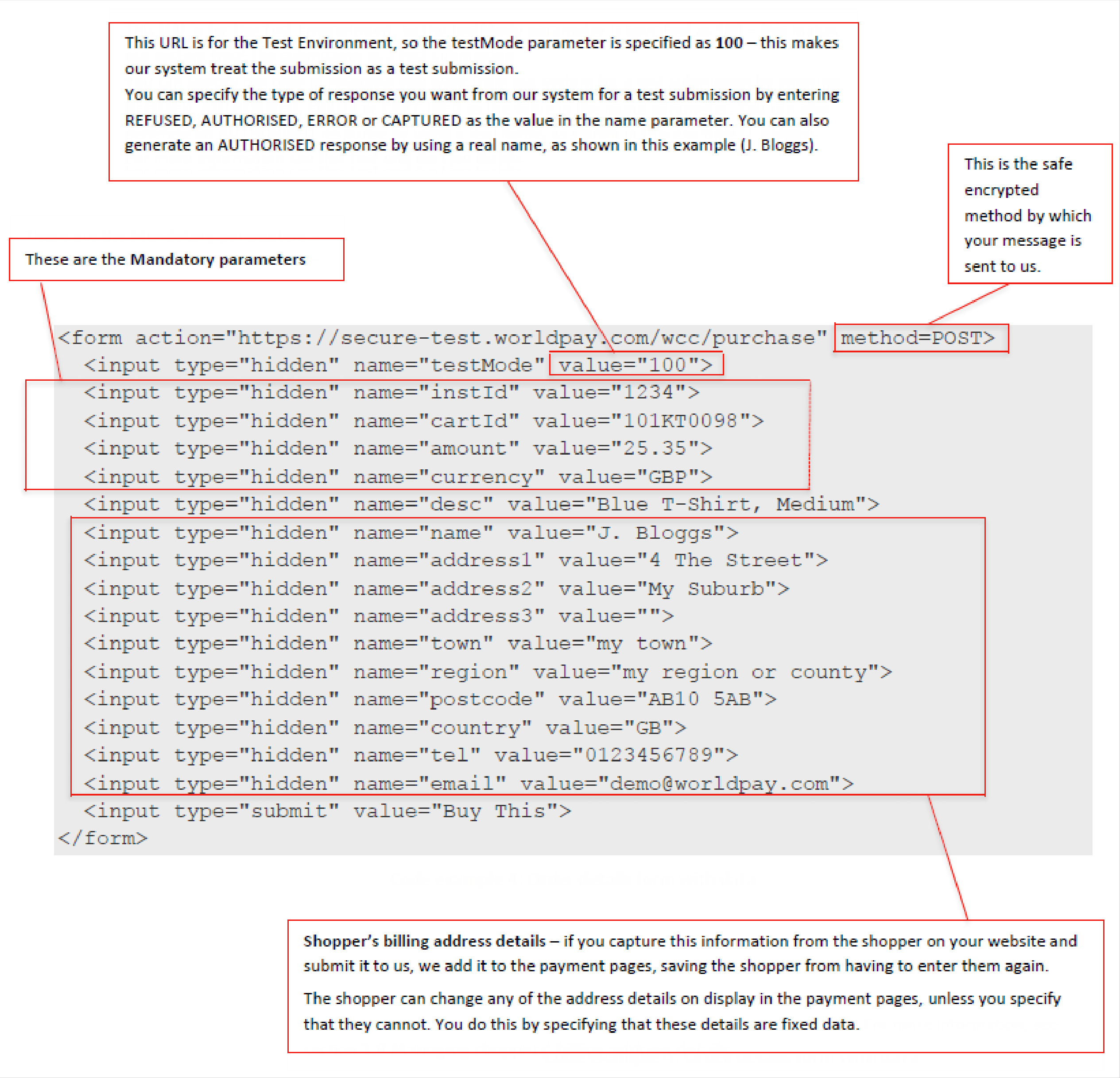
```
<input type="hidden" name="accId1" value="MYCOMPANY">
```

By default, our system searches your merchant codes in order until one suitable for the payment is found.

If none of the specified merchant codes matches, then our system goes on to try any other merchant codes you may have. Hence, these parameters specify a preference rather than an absolute list of merchant codes to use.

3.7 An example order details form with data

The form-based HTML code shown below is an example of the data that can be submitted to us for a single-item payment.



This URL is for the Test Environment, so the testMode parameter is specified as **100** – this makes our system treat the submission as a test submission.
You can specify the type of response you want from our system for a test submission by entering REFUSED, AUTHORISED, ERROR or CAPTURED as the value in the name parameter. You can also generate an AUTHORISED response by using a real name, as shown in this example (J. Bloggs).

This is the safe encrypted method by which your message is sent to us.

These are the **Mandatory parameters**

```
<form action="https://secure-test.worldpay.com/wcc/purchase" method=POST>
  <input type="hidden" name="testMode" value="100">
  <input type="hidden" name="instId" value="1234">
  <input type="hidden" name="cartId" value="101KT0098">
  <input type="hidden" name="amount" value="25.35">
  <input type="hidden" name="currency" value="GBP">
  <input type="hidden" name="desc" value="Blue T-Shirt, Medium">
  <input type="hidden" name="name" value="J. Bloggs">
  <input type="hidden" name="address1" value="4 The Street">
  <input type="hidden" name="address2" value="My Suburb">
  <input type="hidden" name="address3" value="">
  <input type="hidden" name="town" value="my town">
  <input type="hidden" name="region" value="my region or county">
  <input type="hidden" name="postcode" value="AB10 5AB">
  <input type="hidden" name="country" value="GB">
  <input type="hidden" name="tel" value="0123456789">
  <input type="hidden" name="email" value="demo@worldpay.com">
  <input type="submit" value="Buy This">
</form>
```

Shopper's billing address details – if you capture this information from the shopper on your website and submit it to us, we add it to the payment pages, saving the shopper from having to enter them again.

The shopper can change any of the address details on display in the payment pages, unless you specify that they cannot. You do this by specifying that these details are fixed data.

Figure 2: Example order details form

For information about the parameters included in this form, see ***Order details* on page 18**. For a full list of all the parameters that can be submitted to initiate a payment, see ***HTML Redirect parameters* on page 35**.

3.8 Managing shoppers' billing address details

If you pass the shopper's billing address details to us when you submit order details, we automatically place them into the billing address fields that the shopper would otherwise be required to enter in the payment pages. However, the shopper can change these address details in the payment pages unless you specify that they are fixed data.

3.8.1 Fixing and hiding parameters

To make sure your shopper can't change their billing address details on the payment pages:

1. Make the billing address fields mandatory on your site – how you do this is a matter of personal choice.
2. Use an additional parameter in your order details called fixContact to lock the contact information in the payment page.

Using the `fixContact` parameter locks the address information passed to us, so that your shoppers cannot change this information when they reach the payment pages, as shown in the example below.

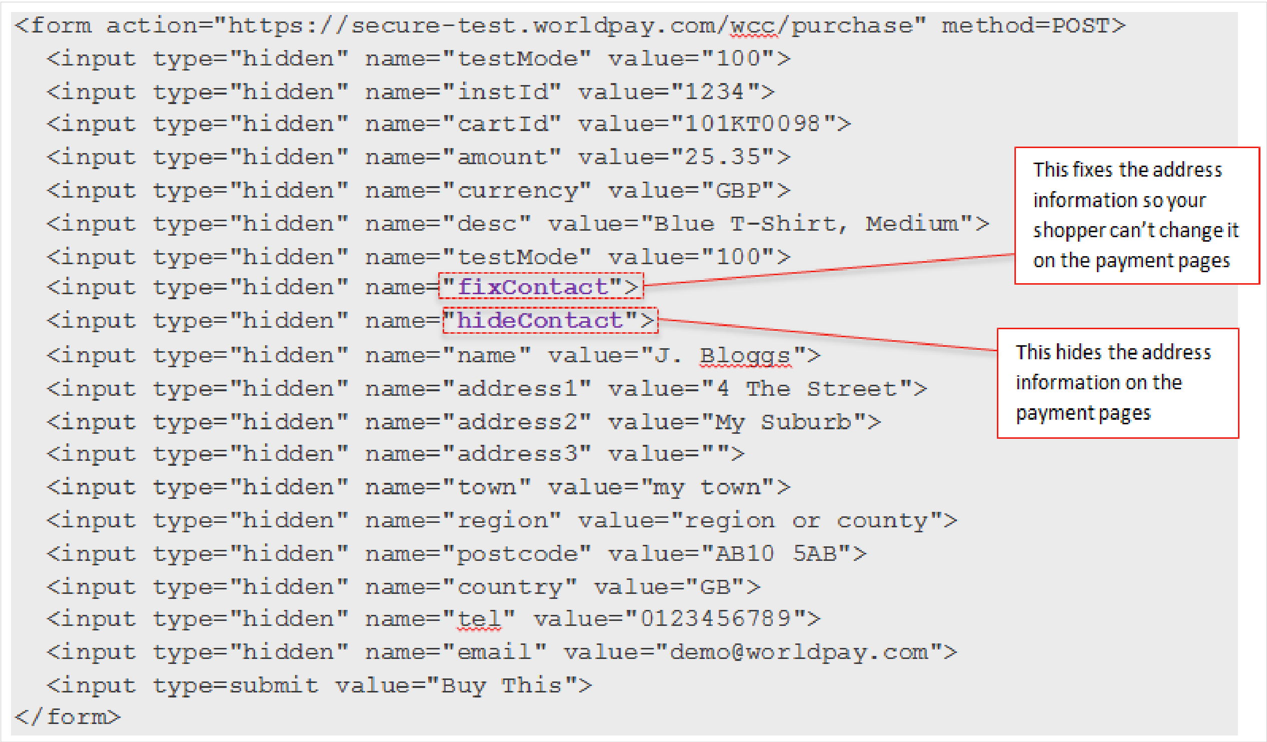
Alternatively, you can use the `hideContact` parameter to hide the address information of shoppers on the payment pages.

For V.me by Visa transactions, the billing address details supplied by V.me override any billing address submitted by you, even if the fixContact and hideContact parameters are used.

3.8.2 Examples of fixContact and hideContact use

The following example shows you how to use the fixContact and hideContact parameters.

If you use both the fixContact and hideContact parameters, the hideContact parameter takes precedence.



```
<form action="https://secure-test.worldpay.com/wcc/purchase" method=POST>
  <input type="hidden" name="testMode" value="100">
  <input type="hidden" name="instId" value="1234">
  <input type="hidden" name="cartId" value="101KT0098">
  <input type="hidden" name="amount" value="25.35">
  <input type="hidden" name="currency" value="GBP">
  <input type="hidden" name="desc" value="Blue T-Shirt, Medium">
  <input type="hidden" name="testMode" value="100">
  <input type="hidden" name="fixContact">
  <input type="hidden" name="hideContact">
  <input type="hidden" name="name" value="J. Bloggs">
  <input type="hidden" name="address1" value="4 The Street">
  <input type="hidden" name="address2" value="My Suburb">
  <input type="hidden" name="address3" value="">
  <input type="hidden" name="town" value="my town">
  <input type="hidden" name="region" value="region or county">
  <input type="hidden" name="postcode" value="AB10 5AB">
  <input type="hidden" name="country" value="GB">
  <input type="hidden" name="tel" value="0123456789">
  <input type="hidden" name="email" value="demo@worldpay.com">
  <input type=submit value="Buy This">
</form>
```

This fixes the address information so your shopper can't change it on the payment pages

This hides the address information on the payment pages

Figure 3: Using fixContact and hideContact

For more information on parameter names for contact details, see section ***HTML Redirect parameters* on page 35**.

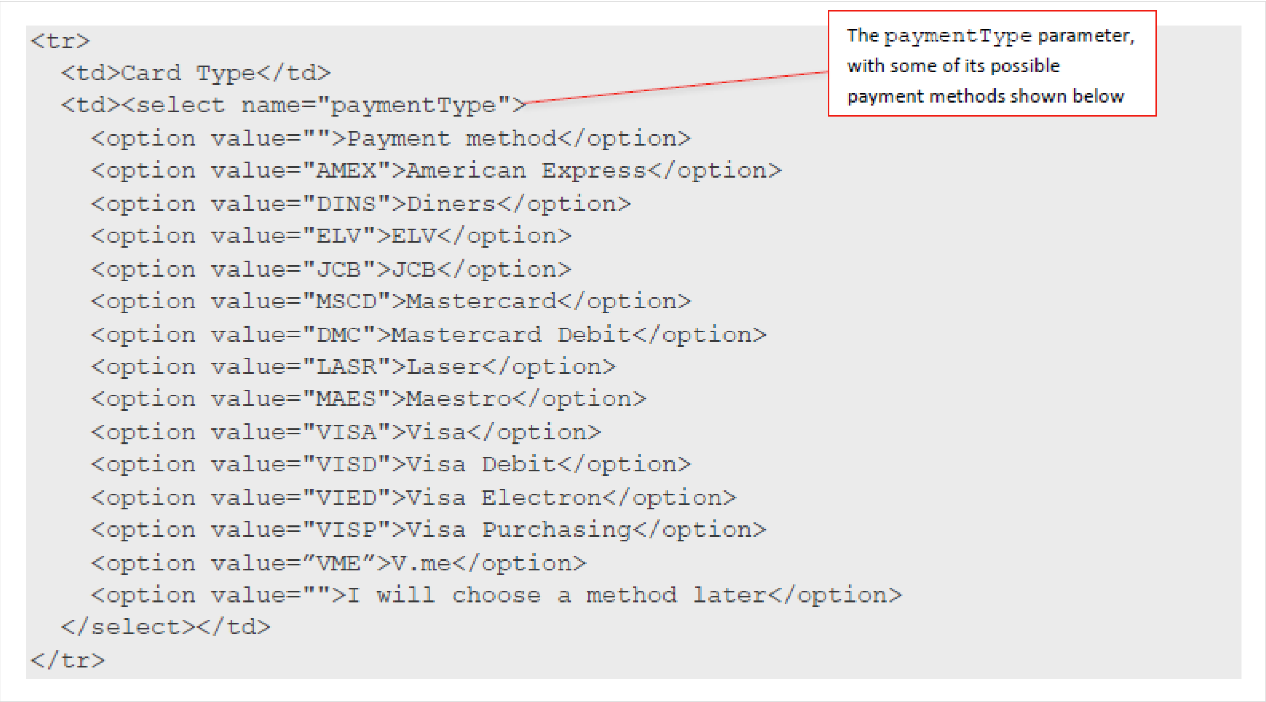
3.9 Bypassing the Payment Selection page

If you already know a shopper's desired payment method, you can enable the shopper to skip or bypass the Payment Selection page and go directly to the Payment page. Here they can enter details for their chosen payment method, such as the card number and its expiry date.

*If you have enabled multiple languages and you allow shoppers to bypass the Payment Selection page, the Select Language list box usually displays on the Payment Page. To avoid this, you need to specify a language and then "fix" it – for details, see Specifying the Language in the **Customising Guide (Advanced)**.*

To enable bypassing, you must supply the payment method in the order details submission by including the parameter `paymentType`, with its value set to the appropriate code for the payment method, as shown in the example below. See ***Payment method codes* on page 54**.

You may need to modify this HTML fragment, so that it displays only the payment methods that our payment service can accept for your installation.



```
<tr>
  <td>Card Type</td>
  <td><select name="paymentType">
    <option value="">Payment method</option>
    <option value="AMEX">American Express</option>
    <option value="DINS">Diners</option>
    <option value="ELV">ELV</option>
    <option value="JCB">JCB</option>
    <option value="MSCD">Mastercard</option>
    <option value="DMC">Mastercard Debit</option>
    <option value="LASR">Laser</option>
    <option value="MAES">Maestro</option>
    <option value="VISA">Visa</option>
    <option value="VISD">Visa Debit</option>
    <option value="VIED">Visa Electron</option>
    <option value="VISP">Visa Purchasing</option>
    <option value=”VME”>V.me</option>
    <option value="">I will choose a method later</option>
  </select></td>
</tr>
```

The `paymentType` parameter, with some of its possible payment methods shown below

Figure 4: Collecting the payment method

3.9.1 How to advise your shoppers about payment methods and cards

If you collect payment methods on your website you may wish to add some supporting information to assist your shoppers in their choice of payment method, such as advice on identifying cards.

You may also want to advise your shoppers about misreading their cards, which may cause them to select the wrong payment method for their card.

For example, they might select Visa as the payment method, when they actually have a Maestro card. If they do this and enter the Maestro card number into the Payment page, it will not match, and in this case their payment attempt will fail. However, in this case we redisplay the Payment page, and thus enable entry of payment details for the selected Visa card.

Add extra shopper advice to your Payment pages

When we redisplay the Payment page, this could be a good opportunity for you to add extra advice to your customised Payment Service pages to assist the shopper. For instance, you could enable shoppers to review their order, or you could provide a link to a troubleshooting page, before they try again. It could cut down on the number of issues in your system, and increase the number of successful transactions.

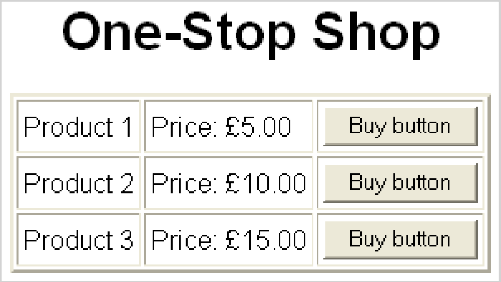
To do this, you simply modify our error messages and add you own. Make yourself familiar with the error messages that we display to shoppers for incorrect entry of payment details, and then modify them within the appropriate *messages_*.properties* files.

4 A complete HTML page

This chapter describes a complete and working example of a simple integrated HTML page. It includes an order details submission form, and helpful comments.

To use the page:

1. Copy the entire section of **HTML code shown below**, and save it as an HTML file in a suitable folder. To do this, use Notepad++, PS Pad or other source text editor.
2. Double-click the HTML file in the folder to run it. When the page opens in your browser, it displays the table shown below:



One-Stop Shop

Product 1	Price: £5.00	Buy button
Product 2	Price: £10.00	Buy button
Product 3	Price: £15.00	Buy button

Figure 5: Figure 2: A complete HTML page

3. Select one of the Buy buttons and then follow the sequence of secure pages to experience a transaction in progress.
4. Use the following test card details to proceed with the transaction (for more test card details see ***Test card numbers* on page 53**):
   - Visa Card Number: 4917610000000000 or MasterCard Number: 5454545454545454
   - Expiry Date: A date later than today but within seven years from submission
   - Security Code: 123

The transaction is sent to the Test Environment so no actual money changes hands. An email showing the transaction details is sent to the address you specify in the payment pages.

If you wish to make test transactions through your Worldpay merchant code then please use your Installation ID for the `instId` *parameter. To view the payments you will need to log in to the Merchant Interface.*

4.1 HTML for a complete page

```
<!-- HTML Redirect Example 0.1
                                                           -->
<!-- Please note the code on this page is provided by Worldpay as a working example only.
  -->
<!-- Any changes you make to the copies of these pages will not be supported by us.     --
>

<html>
<!-- The name, style, and properties of the page are defined in between the 'head' tags. -
->
<head>
```

```
<title>HTML Redirect 0.1</title>
<meta http-equiv="Content-Type" content="text/html">
<meta name="description" content="HTML Redirect Example 0.1">
<meta name="keywords" content="Redirect, html">
<style type="text/css">td {text-align:"left"; vertical-align:"middle"; font-
family:"arial";
color:"black"} h1,h2,h3,h4,h5,h6,h7 {text-align:"center"; vertical-align:"middle"; font-
family:"arial"; color:"black"}</style>
</head>

<!-- The content to be used on the page is placed between the 'body' tags. -->
<body>
<!-- This is an order details submission form, for more information on the elements within
a form see the sections 3.5 to 3.7 -->

<form action="https://secure-test.worldpay.com/wcc/purchase" name="BuyForm" method="POST">
<input type="hidden" name="testMode" value="100">
<!-- These first four elements are mandatory. -->
<input type="hidden" name="instId" value="1234">
<input type="hidden" name="cartId" value="abc123">
<input type="hidden" name="currency" value="GBP">
<input type="hidden" name="amount" value="0">

<!-- These elements below are optional. -->
<input type="hidden" name="desc" value="Blue T-Shirt, Medium">
<input type="hidden" name="name" value="CAPTURED">

<!-- End of order details submission form elements. -->

<!-- JavaScript is used to give functionality to some of the pages elements. -->
<!-- This function defines the price of each product. To add the product edit further down
the page. -->
<script language=JavaScript>
function calc(productNo)
{
if (productNo==1)
{
document.BuyForm.amount.value = 5.00;
document.BuyForm.desc.value = "Product 1";
}
else if (productNo==2)
{
document.BuyForm.amount.value = 10.00;
document.BuyForm.desc.value = "Product 2";
}
// To add a new product price, copy from here
else if (productNo==3)
{
document.BuyForm.amount.value = 15.00;
document.BuyForm.desc.value = "Product 3";
}
//...to here, and paste directly below.
// You will need to alter the 'productNo' and its price value.
}
</script>

<h1>One-Stop Shop</h1>

<!-- This table provides layout for the products listed on the webpage. -->
<table align="center" cellpadding="3" border="2">
<tr>
<td>Product 1</td>
```

```
<td> Price: &pound;5.00</td>
<td><input type="submit" value="Buy button" onClick="calc(1)"></td></tr>
<tr>
<td>Product 2</td>
<td>Price: &pound;10.00</td>
<td><input type="submit" value="Buy button" onClick="calc(2)"></td>
</tr>
<!-- To add a new product, copy from here... -->
<tr>
<td>Product 3</td>
<td>Price: &pound;15.00</td>
<td><input type="submit" value="Buy button" onClick="calc(3)"></td>
</tr>
<!-- ...to here, and paste directly below. You will need to alter three things: product
number, price, and calc (put product number here) -->
</table>
</form>
</body>
</html>
```

Code example 4: HTML for a complete page

5 Telling you about payment results

This chapter describes the types of payment result sent to you after we process your payment.

It describes:

- The different ways you can view or receive information about the payment result
- The different types of payment result

The shopper can cancel at any point in the payment process, including when a card is declined by its issuer. The shopper always has the choice of trying another method of payment, such as another card.

5.1 Viewing the payment result

When a payment has been processed, information about the processed transaction is made available to you in the following ways:

Information channels	**Description**
Merchant Interface	You can log in to the Merchant Interface, which is our payments management tool, to view details of a successful transaction.
Email confirmations	We send you an email about each successful transaction. The confirmation email is sent to the address you specified for the merchant code receiving the payment.
Payment Response message	If you have Payment Responses enabled, a message is sent to you whenever a payment transaction is either successfully completed or cancelled. All the details of the transaction are sent back to you in the message. For more information, see section ***Enabling and using Payment Response messages*** **on page 34**.

Table 3: Viewing the payment result

5.2 Types of payment result

The following table shows the types of payment result sent to you when a payment has been processed.

Status / condition	**Transaction ID generated (view with the Merchant Interface)**	**Email sent**	**Payment notification sent**
Successful	Yes	Yes	Yes
Declined	Yes	No	No
Cancelled	No	No	Yes
Browser closed, connection broken or similar	No	No	No

Table 4: Types of payment result

The Payment Notification sent column is only relevant if you have Payment Responses enabled. For more information, see ***Enabling and using Payment Response messages on page 34****.*

6 Telling your shopper about payment results

This chapter describes the types of payment results which may be sent to your shoppers, either by you or by Worldpay, after we process a payment.

Once a payment has been processed, both of these are sent automatically:

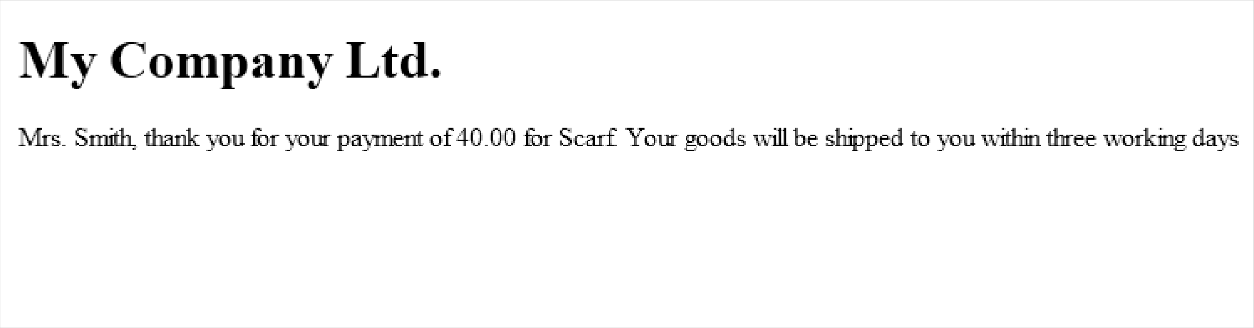
Information channels	Description
Result page	A result page is displayed to the shopper. You can decide to accept the default result pages or, alternatively, you can replace them by uploading your own versions, either generated by yourself solely (see ***Uploading your own result pages* on the next page**), or using the **Payment Responses** feature.
Confirmation email	A confirmation email showing the transaction details is sent to the email address they supplied during the purchase. You can also add your own text to the confirmation email using the **Payment Responses** feature. Note that this confirmation email is not a VAT receipt.

Table 5: Payment results sent to your Shopper

We cannot guarantee that the shopper will receive the email as we cannot verify the accuracy of the email address supplied, but they should get the result page (provided that their browser remains connected).

6.1 Telling your shopper about payment results using result pages

A result page is a web page that tells your shopper if their payment was successful or not.



My Company Ltd.

Mrs. Smith, thank you for your payment of 40.00 for Scarf. Your goods will be shipped to you within three working days

Figure 6: Example result page sent to the shopper

Typically, you would display this page when a transaction has been processed. One of two result pages is returned to the shopper: a successful transaction page or a cancelled transaction page.

6.1.1 Result pages

We send out one of these result files:

- resultY.html for a successful transaction
- resultC.html for a cancelled transaction

There are two ways in which you can supply your own customised response:

Response	Description
Uploading	You can upload your own versions of the resultY.html and resultC.html files to our server to replace the default result pages. For more information, see section ***Uploading your own result pages* below**.
Payment Response Result	If you are using the Payment Response feature, you can respond to our Payment Response message with a script of your own, by sending an appropriate page of HTML, which is displayed to the shopper instead of the resultY.html or resultC.html files. For more information, see section ***Enabling and using Payment Response messages* on page 34**.

Table 6: Your customised responses

6.2 Telling your shopper about payment results by email confirmation

We send a confirmation email to your shopper at the email address they supplied, to tell them if their payment was successful or not. Note that this is not a VAT receipt. In addition, you can send your shopper an email using the **Confirmation Email** option in the Payment Pages Editor.

We are legally required to provide some information in the email text, such as your company name.

6.3 Uploading your own result pages

You can replace the default `resultY.html` and `resultC.html` files by uploading your own versions to our payment service.

We recommend that you create your own versions from scratch rather than editing our files, which include complex tags. However, you may want to study our files for reference.

Whether you edit our files or create your own, please ensure that the file names match those we use: the file called resultY.html is for a successful transaction and the file called resultC.html is for a cancelled transaction.

Take care to save the Results files with the .html extension, as the .htm extension causes server issues.

6.3.1 Creating your custom results files

To create your own custom files:

1. In the merchant Interface, select the **Installations** option in the left-hand menu. Click the **Edit Payment pages** button.
2. Select the required Installation ID from the drop-down list, and click the **Edit Payment pages** icon.
3. Select the **File Management** option in the Payment Page Editor.
4. In the **HTML Files** section of the **File Management** window, click on the required Result file which then opens in your browser. Save it to your PC.

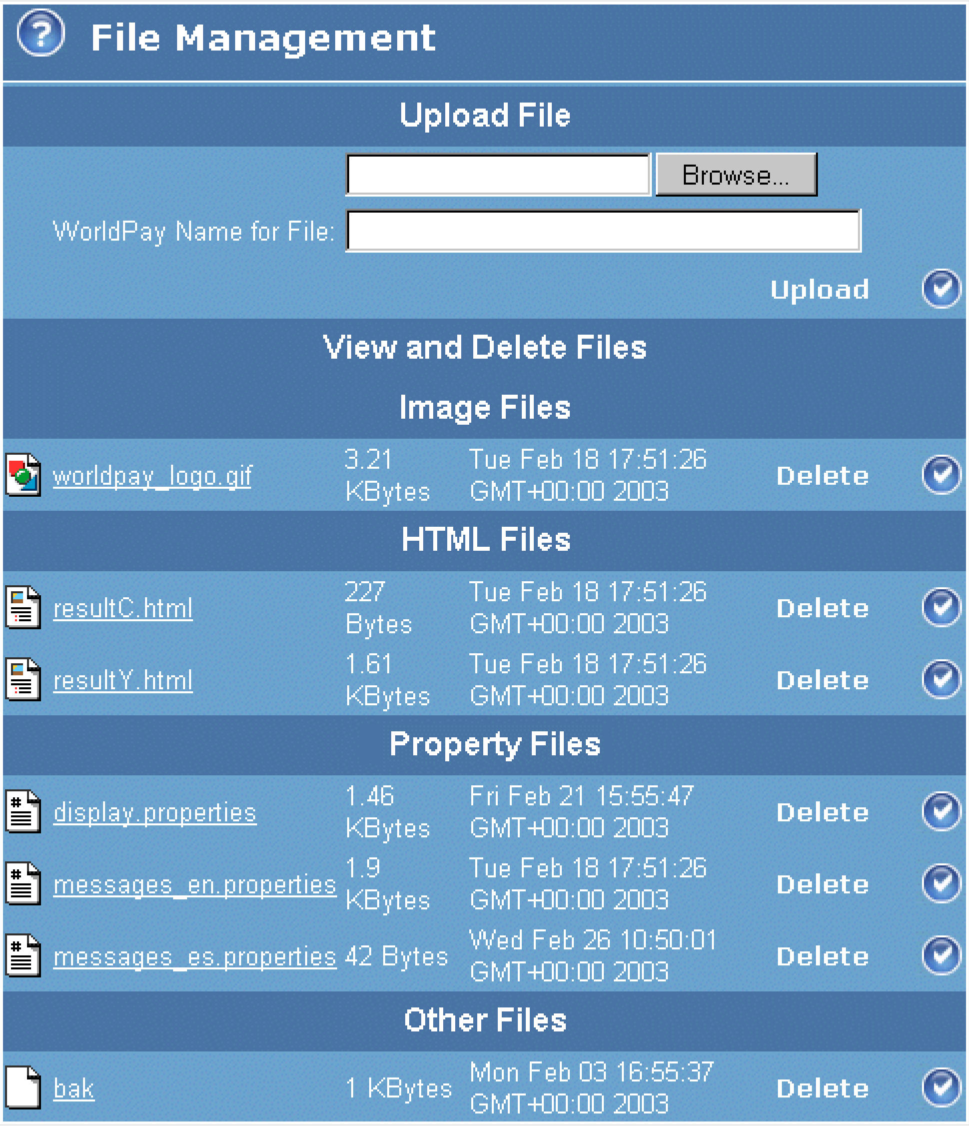


Image Files			
worldpay_logo.gif	3.21 KBytes	Tue Feb 18 17:51:26 GMT+00:00 2003	Delete
HTML Files			
resultC.html	227 Bytes	Tue Feb 18 17:51:26 GMT+00:00 2003	Delete
resultY.html	1.61 KBytes	Tue Feb 18 17:51:26 GMT+00:00 2003	Delete
Property Files			
display.properties	1.46 KBytes	Fri Feb 21 15:55:47 GMT+00:00 2003	Delete
messages_en.properties	1.9 KBytes	Tue Feb 18 17:51:26 GMT+00:00 2003	Delete
messages_es.properties	42 Bytes	Wed Feb 26 10:50:01 GMT+00:00 2003	Delete
Other Files			
bak	1 KBytes	Mon Feb 03 16:55:37 GMT+00:00 2003	Delete

Figure 7: File Management

5. Now you can make your desired changes.

The result page automatically displays with the header and footer stored for your installation on our payment service. These are either our defaults or your own, if you have created and uploaded them via the Payment Page Editor.

If you use the default result page, the standard banner is included in the final payment result page.

The banner contains legal information that must be conveyed to the shopper.

The banner is inserted at the <WPDISPLAY> tag, as follows:

```
<WPDISPLAY ITEM="banner">
```

If you customise or replace the result page, ensure that you include the header, footer, and the banner for the final payment result page.

6. Edit the rest of the page (which is under your control), as shown in the examples in the following sections.

   For further details about editing and adding the banner, see the **Customising Guide (Advanced)**

7. Use the Payment Page editor to upload the changed file back to your Test installation.

6.3.2 Adding your own details to the result page

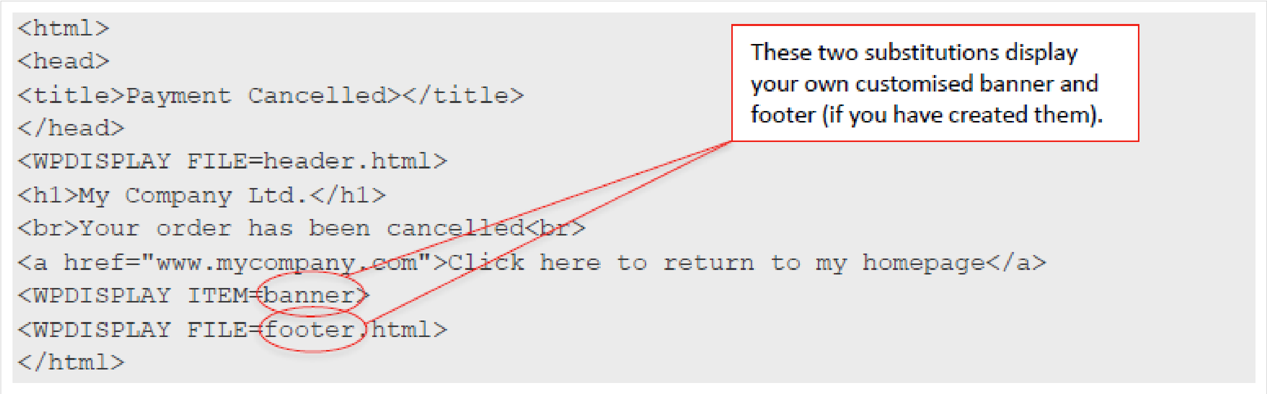
You can add your own details to the result page, in order to individualise each transaction. For example, you can add the shopper's name, the product description and the cost. This procedure is known as parameter substitution.

When we pass the result page to the shopper, we automatically include your substitutions in it, relevant to that particular transaction. The example below in ***Example – resultY.html: successful transaction* on the next page** shows parameter substitution at the <WPDISPLAY> tags.

For more information about substitution and how you do it, see ***Customising your HTML messages* on page 44**.

*You can also use custom parameters in your resultY.html or resultC.html files; for more information, see **Custom parameters**.*

6.3.3 Example – resultC.html: cancelled transaction

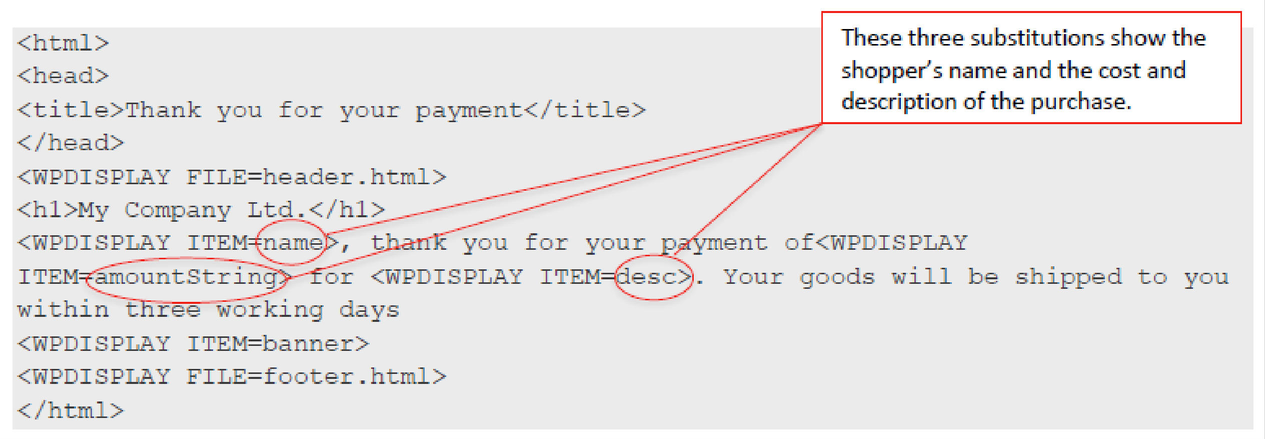


```
<html>
<head>
<title>Payment Cancelled></title>
</head>
<WPDISPLAY FILE=header.html>
<h1>My Company Ltd.</h1>
<br>Your order has been cancelled<br>
<a href="www.mycompany.com">Click here to return to my homepage</a>
<WPDISPLAY ITEM=banner>
<WPDISPLAY FILE=footer.html>
</html>
```

These two substitutions display your own customised banner and footer (if you have created them).

Figure 8: resultC.html - cancelled transaction

6.3.4 Example – resultY.html: successful transaction



```
<html>
<head>
<title>Thank you for your payment</title>
</head>
<WPDISPLAY FILE=header.html>
<h1>My Company Ltd.</h1>
<WPDISPLAY ITEM=name>, thank you for your payment of<WPDISPLAY
ITEM=amountString> for <WPDISPLAY ITEM=desc>. Your goods will be shipped to you
within three working days
<WPDISPLAY ITEM=banner>
<WPDISPLAY FILE=footer.html>
</html>
```

These three substitutions show the shopper's name and the cost and description of the purchase.

Figure 9: resultY.html - successful transaction

6.3.5 Enabling and using Payment Response messages

You can set up and use the Payment Response messages feature, which enables you to automate and control your website. The Payment Response feature posts the payment information from our server to a URL on your server securely. This enables you to:

- Validate the information we return to you
- Update the order, and use the information to trigger actions on your own system
- Display your own shopper result pages, if required

You can also benefit from the content of messages we send to you, as follows:

- More data is returned to you in Payment Response messages than you would receive in the standard email notification
- You can pass your own parameters through our payment service and back to yourself
- Examine any available fraud check results, such as those arising from verification of Address, Security Code and Authentication information
- Maintain any recurring payment agreements

For comprehensive coverage of this feature, including setup and use, see the **Payment Response Guide**.

7 HTML Redirect parameters

This chapter is intended as a quick reference to the HTML Redirect parameters, which you use when creating and customising orders, responses and other HTML messages.

They are grouped according to their intended use:

- ***Initiating purchase transaction parameters* below**
- ***Payment page parameters* on page 40**
- ***Payment result parameters* on page 40**
- **Recurring payments (FuturePay) parameters**
- **Custom parameters**

Each parameter must be named precisely as shown. Also note that the parameters are case-sensitive.

All parameters are available for substitution, as discussed in section ***Adding your own details to the result page* on page 33**. For more information about substitution and the tasks involved, see ***Customising your HTML messages* on page 44**

For information about Recurring Payments (FuturePay) parameters, see the ***Recurring Payments Service (FuturePay) Guide***.

7.1 Initiating purchase transaction parameters

The parameters described in these tables are intended to be used when submitting order details; doing so initiates a purchase transaction. Note that some parameters are mandatory.

Where a length in characters is given, this is the maximum length that the parameter can be – anything longer is truncated to this length.

7.1.1 Mandatory parameters

Order details parameters can be mandatory either with regard to the order details submission or the payment pages.

For example, you can optionally include the shopper's address in the order details submission page, but if you do not include the address, the shopper must then provide this information on the payment page.

The following four parameters must be included in the order details you submit.

Parameter name	Type	Description
instId	Integer	Your Installation Id.
cartId	255 char	Your own reference number for this purchase. It is returned to you along with the authorisation results by whatever method you have chosen for being informed (email and / or Payment Responses).

Parameter name	Type	Description
amount	Decimal	A decimal number giving the cost of the purchase in terms of the major currency unit e.g. 12.56 would mean 12 pounds and 56 pence if the currency were GBP (Pounds Sterling). Note that the decimal separator must be a dot (.), regardless of the typical language convention for the chosen currency. The decimal separator does not need to be included if the amount is an integral multiple of the major currency unit. Do not include other separators, for example between thousands.
currency	3 char	3 letter ISO code for the currency of this payment – see ***ISO currency codes* on page 50**.

Table 7: Mandatory parameters

7.1.2 Optional parameters

The following parameters are optional with regard to the order details submission.

Some parameters are mandatory with regard to the payment pages – if they are not supplied in the order details then they must be entered in the payment pages by the shopper.

Parameter name	Type	Mandatory in payment page?	Description
address1	84 char	Yes	The first line of the shopper's address. Encode newlines as "&#10;" (the HTML entity for ASCII 10, the new line character). If this is not supplied in the order details then it must be entered in the payment pages by the shopper
address2	84 char	No	The second line of the shopper's address. Encode newlines as "&#10;" (the HTML entity for ASCII 10, the new line character).
address3	84 char	No	The third line of the shopper's address. Encode newlines as "&#10;" (the HTML entity for ASCII 10, the new line character).
town	30 char	Yes	The town or city. Encode newlines as "&#10;" (the HTML entity for ASCII 10, the new line character). If this is not supplied in the order details then it must be entered in the payment pages by the shopper.
region	30 char	No	The shopper's region/county/state. Encode newlines as "&#10;" (the HTML entity for ASCII 10, the new line character).
postcode	12 char	Can be set to mandatory	The shopper's postcode. Note that at your request we can assign mandatory status to this parameter. That is, if it is not supplied in the order details then the shopper must enter it in the payment pages.

Parameter name	Type	Mandatory in payment page?	Description
country	2 char	Yes	The shopper's country, as 2-character ISO code, uppercase. See ***ISO country codes* on page 52**. If this is not supplied in the order details then it must be entered in the payment pages by the shopper.
desc	255 char	-	A textual description of this purchase, up to 255 characters. This is used in web-pages, statements and emails for yourself and the shopper.
resultFile	string	-	The name of one of your uploaded files, which is used to format the result. If this is not specified, resultY.html or resultC.html is used as described in ***Telling your shopper about payment results* on page 30**.
accId<n>	string	-	This specifies which merchant code should receive funds for this payment. By default our server tries accId1.
authMode	char	-	This specifies the authorisation mode to use. If there is no merchant code with a matching authMode then the transaction is rejected. The values are "A" for a full auth, or "E" for a pre-auth. In the payment result this parameter can also take the value "O" when performing a post-auth.
testMode	integer	-	A value of 100 specifies that this is a test payment. Specify the test result you want by entering REFUSED, AUTHORISED, ERROR, or CAPTURED in the name parameter. When you submit order details using the testMode parameter and the URL for the live Production Environment, you are presented with a page asking you if you want to redirect the order details to the Test Environment – select the Redirect button if you do. If you submit the order details to the live production environment our systems attempt to debit merchant codes (accounts). Reversing transactions such as these, and adjusting accounts, causes unnecessary work for us as well as you. Set this parameter to 0 (zero) or omit it for a live transaction. See the **Test and Go Live Guide**.

Parameter name	Type	Mandatory in payment page?	Description
authValidFrom	integer	-	This specifies a time window within which the purchase must (or must not) be completed, for example, if the purchase is a time-limited special offer. Each of these parameters is a time in milliseconds since 1 January 1970 GMT - a Java long date value (as from System.currentTimeMillis() or Date.getTime()), or 1000* a C time_t. If from<to, then the authorisation must complete between those two times. If to<from, then the authorisation must complete either before the to time or after the from time. Either may be zero or omitted to give the effect of a simple "not before" or "not after" constraint. If both are zero or omitted, there are no restrictions on how long a shopper can spend making their purchase (although our server times out their session if it is idle for too long).
authValidTo	integer		
name	40 char	-	The shopper's full name, including any title, personal name and family name. Note that if you do not pass through a name, and use Payment Responses, the name that the cardholder enters on the payment page is returned to you as the value of name in the Payment Responses message. Also note that if you are sending a test submission you can specify the type of response you want from our system by entering REFUSED, AUTHORISED, ERROR or CAPTURED as the value in the name parameter. You can also generate an AUTHORISED response by using a real name, such as, J. Bloggs. For more information see the **Test and Go Live Guide**.
tel	30 char	-	The shopper's telephone number.
fax	30 char	-	The shopper's fax number.
email	80 char	Yes	The shopper's email address. If this is not supplied in the order details then the shopper must enter it in the payment pages.

Table 8: Optional parameters

7.1.3 Display parameters

The following parameters control the appearance of the payment pages.

Parameter name	Type	Description
fixContact	needs no value	If present, this causes contact details to be displayed in non-editable format. You must ensure that all mandatory contact details are submitted in your initial request.
hideContact	needs no value	If present, this causes contact details to be hidden. You must ensure that all mandatory contact details are submitted in your initial request. Existing merchants should set the following message files to empty strings for the feature to work: cont.instr.existing, cont.instr.new, cont.heading
hideCurrency	needs no value	If present, this causes the currency drop down to be hidden, so fixing the currency that the shopper must purchase in.
lang	6 char	The shopper's language choice, as a 2-character ISO 639 code, with optional regionalisation using 2-character country code separated by hyphen. For example "en-GB" specifies UK English. The shopper can always choose a language on our pages or via browser preferences but if your site has already made this choice then you can make things more convenient by submitting it to us.
noLanguageMenu	needs no value	This suppresses the display of the language menu if you have a choice of languages enabled for your installation but want the choice to be defined by the value of the lang parameter that you submit. Please contact your local Technical Support department if you would like this facility enabled on your account.
withDelivery	needs no value	Displays input fields for delivery address and mandate that they be filled in.

Table 9: Display parameters

7.1.4 The subst parameter

This parameter is intended for use during testing. It is only relevant if you are creating your own messages files.

Parameter name	Type	Description
subst	string: "yes" or "no"	If the value is "no" then message substitution is turned off. This means that you see the names of the message properties from the messages_xx.properties file used to create the page. This situation persists until you submit a payment with subst=yes, or your session is ended.

Table 10: The subst parameter

7.2 Payment page parameters

All of the parameters shown in the Initiating Purchase Transaction Parameters tables are available for use in the payment pages. The following parameters are also available for use in the payment pages.

Parameter name	Type	Description
amountString	Variable length char string	An HTML string produced from the amount and currency that were submitted to initiate this purchase.
countryString	Variable length char string	The full name of the country, derived from the country code submitted to initiate this purchase or supplied by the shopper.
compName	Variable length char string	Name of the company associated with this installation.

Table 11: Payment page parameters

7.3 Payment result parameters

All of the parameters shown in the Initiating Purchase Transaction Parameters tables, and the Payment Page Parameters table are available for use in the payment results. The following parameters are also available for use in the payment results.

At the payment result stage of processing, the contact details for the shopper are those that were used for authorising the payment.

The parameters authAmount, authCurrency, etc., are set to those of the actual transaction carried out. Note that these could be different to the values submitted to us when the transaction was initiated, as the shopper may have chosen a different currency when in the payment pages.

Where a length in characters is given, this is the maximum length that the parameter can be – anything longer is truncated to this length.

Parameter name	Type	Description
transId	integer - 16 digits	The Worldpay ID for this transaction.
futurePayId	integer - 16 digits	The Worldpay ID for a FuturePay agreement (where relevant).
transStatus	1 char	Result of this transaction - "Y" for a successful payment authorisation, "C" for a cancelled payment (note that as a merchant you never see a declined payment, as the shopper is always given the option of retrying with another means of payment, or else cancelling the payment).
transTime	long integer	The time of this transaction in milliseconds since the start of 1970 GMT. This is the standard system date in Java, and is also 1000x the standard C

Parameter name	Type	Description
		time_t time.
authAmount	decimal	The amount that this transaction was authorised for, in the currency given as authCurrency.
authCurrency	3 char	The currency used for authorisation.
authAmountString	Variable length char string	HTML string produced from authorisation amount and currency
rawAuthMessage	string	The text received from the bank (typically including an authorisation code, or a reason for failure).
rawAuthCode	1 char	A single-character bank (or internal Worldpay) authorisation code. This is retained for backward compatibility. 'A' means 'authorised' and is directly equivalent to transStatus='Y'. Failed transactions may have a variety of auth codes which are usually explained more fully in the rawAuthMessage parameter.
callbackPW	string	The Payment Responses password, if you have set it in our database via the Merchant Interface. This is only available in the parameters sent in the Payment Responses message. It is not available for substitution into the page sent to the shopper.
cardType	string	The type of card used by the shopper.
countryMatch	1 char	A single character describing the result of the comparison of the shopper's contact country (where supplied) and the issue country of the card used by the shopper (where available). Note that this parameter is retained for backward compatibility - equivalent information is now provided as part of the AVS results (see AVS below). Key: Y - Match N – No match (that is, mismatch) B – Comparison not available I - Contact country not supplied S - Card issue country not available
AVS	4 char	The Address Verification System (AVS) provides the ability to check the shopper's billing address and postcode against the card issuer's details. The AVS results combine the results of the AVS check with the results of a CVV check. CVV is a three-digit number printed at the end of the signature strip on the

Parameter name	Type	Description
		vast majority of credit/debit cards. The CVV is a unique number that cannot be replicated by fraudsters, and for this reason, it is commonly known as the 'security code'. The results of Security Code and Address Verification are provided as a four character string in which each character reports the result of a particular verification/check. The characters give the results of the following checks: 1st character - Card Verification Value check 2nd character - Postcode AVS check 3rd character - Address AVS check 4th character - Country comparison check (please also refer to countryMatch) The possible values for each result character are: 0 - Not supported 1 - Not checked 2 - Matched 4 - Not matched

Table 12: Payment result parameters

7.4 Recurring payments (FuturePay) parameters

You can make online arrangements with your shoppers using the Recurring Payments service, also known as FuturePay. It’s similar to standing orders and direct debits, but with Recurring Payments you arrange to get funds debited from the shopper's card, rather than from their bank account.

You can include Recurring Payments parameters in an order details submission (for details of creating orders, see ***Order details* on page 18**). For instance, this might be to set up a Recurring Payments agreement, where a number of payments are to be made at intervals, such as in a subscription.

To do this, specify the Recurring Payments parameters within the order details, as shown in the example below.

```
<!-- Futurepay parameters start here -->
<INPUT TYPE="HIDDEN" NAME=futurePayType VALUE="regular">
<INPUT TYPE="HIDDEN" NAME=option VALUE=1>
<INPUT TYPE="HIDDEN" NAME=startDelayMult VALUE=3>
<INPUT TYPE="HIDDEN" NAME=startDelayUnit VALUE=2>
<INPUT TYPE="HIDDEN" NAME=noOfPayments VALUE=11>
<INPUT TYPE="HIDDEN" NAME=intervalMult VALUE=1>
<INPUT TYPE="HIDDEN" NAME=intervalUnit VALUE=3>
<INPUT TYPE="HIDDEN" NAME=normalAmount VALUE=17.99>
<INPUT TYPE="HIDDEN" NAME=initialAmount VALUE=10.00>
```

Code example 5: Recurring payments parameters

For more information, see Recurring Payments sections in the **Payment Response Guide**. For information about FuturePay parameters, see the different agreement types in the **Recurring Payments Service (FuturePay) Guide**.

7.5 Custom parameters

You can create your own custom parameters and then use them in your own versions of the resultY.html or resultC.html pages that we return to the shopper in response to their payment.

Your own custom parameters can also be passed through our payment service and returned to your server in a Payment Response message. For more information, see the **Payment Response Guide**.

7.6 Creating custom parameters

When you create custom parameters you must give them different prefixes, depending on their use, as described in the table below.

See code example ***Custom parameters* below** for examples of how they are used.

Custom parameter prefix	Where used	Conditions
"C_"	Custom parameters to be used just in the shopper's result page should have names starting with "C_".	Custom "C_" parameters are available for use in the result pages, but are NOT available to your Payment Responses script.
"M_"	Custom parameters that you want to return to yourself in a Payment Response message should have names starting with "M_".	Custom "M_" parameters are available to your Payment Responses script, but are NOT available for use in the result pages.
"MC_" or "CM_"	For use when you want to use a parameter for both the shopper's result page and in a Payment Response message.	Custom "MC_" and "CM_" parameters are available to both your Payment Responses script and the result pages.

Table 13: Custom parameters

Custom "M_" or "MC_" parameters that you submit in order details are returned in the Payment Responses message. Also note that "C_" parameters are not returned in the Payment Responses message, but are displayed to the shopper in the result page.

```
<input type="hidden" name="M_mycustomvar" value="sent in the HTML Payment Responses
message">
<input type="hidden" name="C_mycustomvar" value="displayed on the Shopper's page">
<input type="hidden" name="MC_mycustomvar" value="M_ and MC_ combined">
```

Code example 6: Custom parameters

8 Customising your HTML messages

This chapter describes how you customise and expand your orders, responses and other HTML messages. You do this by making substitutions in the files used in the construction of pages or emails.

All parameters are available for substitution, provided that the transaction has reached the appropriate stage. For instance, there would not be a `transStatus` value available before a transaction had been processed to completion. If you use an unavailable parameter in substitution then the default as specified in the substitution tag is used.

For details of the parameters available, see ***HTML Redirect parameters* on page 35**.

8.1 Parameter and file substitution

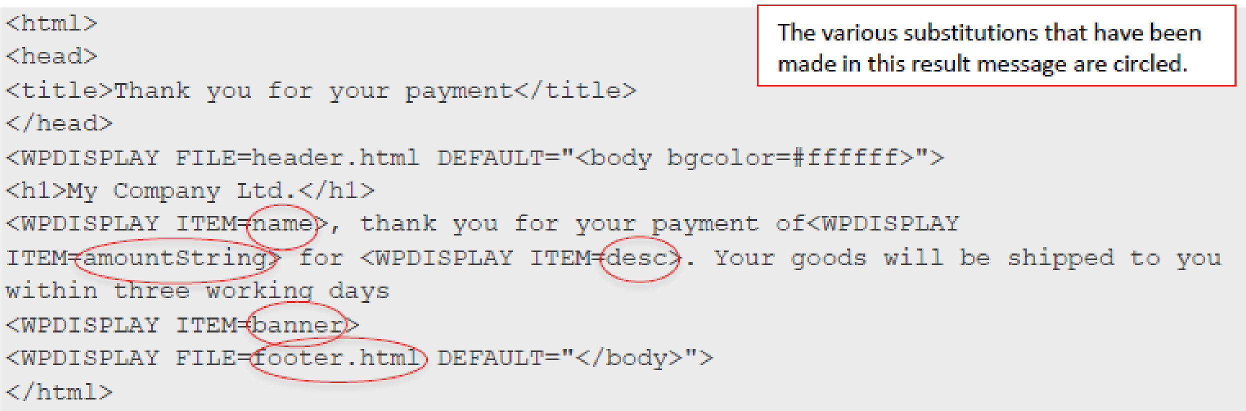
You can carry out parameter substitution in any file that is used in the construction of pages or emails. Substitution is carried out using special WPDISPLAY tags. These tags can be placed anywhere, including within HTML tags. For details of this, see ***Embedding tags within tags* on the next page**.

You can also carry out file substitution in the same way. An example is shown in the code example below, where a footer file has been substituted.

Parameters provide information about the transaction. For instance, the parameter `amountString` is the cost of the item or product in the payment pages, and `desc` is the description of the item or product. Other parameters hold other information, such as the shopper's billing address details and the transaction status. Any of these parameters can be substituted using WPDISPLAY tags.

8.1.1 Substituting the value of parameters

In the following example of a resultY.html response page, each ITEM tag is replaced with the values of the named parameters:



```
<html>
<head>
<title>Thank you for your payment</title>
</head>
<WPDISPLAY FILE=header.html DEFAULT="<body bgcolor=#ffffff>">
<h1>My Company Ltd.</h1>
<WPDISPLAY ITEM=name>, thank you for your payment of<WPDISPLAY
ITEM=amountString> for <WPDISPLAY ITEM=desc>. Your goods will be shipped to you
within three working days
<WPDISPLAY ITEM=banner>
<WPDISPLAY FILE=footer.html DEFAULT="</body>">
</html>
```

The various substitutions that have been made in this result message are circled.

Code example 7: resultY response page

You can also embed your standard header and footer, as well as any graphics you wish. Note that these graphics must be uploaded to our payment service, even if the page is returned as a result page by your server. For more information about uploading graphic files, see the **Customising Guide (Standard)** or **Customising Guide (Advanced)**.

If you have used custom parameters in the order details, any "M_" or "MC_" parameters that you submitted are also returned. Also note that "C_" parameters are not returned in the Payment Responses message, but are displayed to the shopper in the result page. For more information, see ***Custom parameters on page 43****.*

8.1.2 Substituting the value of a display property

The following HTML constructs a font tag containing the value of the parameter named `title.bg` from display.properties. This means that you can use the same parameters that are used in the payment pages in your own templates. You can also add parameters of your own to `display.properties` and use these in your templates.

```
<WPDISPLAY DISP="title.bg" DEFAULT="<font color=#000000>" PRE="<font color=" POST=">">
```

8.1.3 Embedding the contents of a file

In the example below, the FILE tag is replaced with the contents of the named file `footer.html`, which should be in your upload area. The embedded file can contain any Worldpay tags except for another file embedding command.

```
<WPDISPLAY FILE=footer.html DEFAULT="</body>">
```

8.1.4 Embedding tags within tags

The strings given as the values of WPDISPLAY attributes can themselves contain WPDISPLAY tags – except for file embedding tags. For example, given a `cartId` of 123, the line:

```
<WPDISPLAY FILE="blurb<WPDISPLAY ITEM=cartId>.html">
```

expands to:

```
<WPDISPLAY FILE="blurb123.html">
```

which then results in the embedding of the file `blurb123.txt`.

Do not use the same type of quotes in the inner tag: either omit them altogether as in the example above (only viable for single words) or use the other kind (single versus double quote), as shown below.

```
<WPDISPLAY FILE="blurb<WPDISPLAY ITEM='cartId'>.html">
```

8.1.5 Custom parameter substitution

Any "C_" or "MC_" custom parameters that you supplied in the initial submission to us are also available for substitution.

For example, assume that you have assigned a custom parameter `MC_mycustomvar` with the value "Don't forget there is 10% off all gift packs on Friday 10th Sept!". Then, in the following HTML line:

```
<WPDISPLAY ITEM="MC_mycustomvar" DEFAULT="" PRE="<b>" POST="</b>">
```

the PRE bold tag and POST bold-off tag bracket the ITEM tag, so that the substitution is done as follows:

```
<b>Don't forget there is 10% off all gift packs on Friday 10th Sept!</b>
```

which displays as:

Don't forget there is 10% off all gift packs on Friday 10th Sept!

9 Enhancing security with MD5

This chapter describes MD5 encryption and how to use it to improve your security, whether you use a ready-built shopping cart or a bespoke online store.

9.1 What is MD5 encryption?

We use MD5 encryption to improve the security of our HTML Redirect service.

You can use it to add an encrypted signature to each order you send to us, helping to protect your order details from unauthorised tampering.

9.1.1 How does MD5 encryption work?

Using MD5, you can choose which parts of an order you want our gateway to check for unauthorised tampering.

You can also create a password to send as part of your order information, the 'MD5 secret'. This will only be known to us. We can check each order you send to make sure it corresponds with your password.

If the signature does not match this password, we reject the transaction.

You must still include all mandatory parameters in the order details submission.

9.2 What you need to use MD5 encryption

To use MD5 encryption, your store must be set up using either of these:

- A ready-built shopping cart that can use MD5 encryption (please consult your cart vendor for advice). For details about installing MD5, see ***If you use a ready-built shopping cart*** **below**
- A bespoke setup generated using a programming language with an available MD5 library, such as C, C++, Visual Basic, PHP, ASP and Perl (but not HTML or Javascript). For details about installing MD5, see ***If you use a bespoke setup*** **on the next page**

HTML and Javascript are unsuitable for this feature because it is essential the encryption process happens server-side to ensure the security of the data.

9.2.1 If you use a ready-built shopping cart

If you use an off-the-shelf shopping cart, you need to take the following steps to add MD5 encryption:

1. **Enable MD5 encryption in your store, following the advice of your shopping cart provider.** Implementation of MD5 will vary, depending on which shopping cart you use. You will probably need to choose the fields included in the encrypted signature, and supply your shopping cart with the 'MD5 secret' used to encrypt the signature.

   The MD5 secret should be a string (spaces are permitted) of 20 to 30 characters, known only to yourself and to us. Capitals, numerals and punctuation are all permitted.

2. **Set your MD5 secret in the Merchant Interface.** To do this, enter this value into the **MD5 secret for transactions** field in the Integration Setup for your installation using the Merchant Interface >

Installations option. If you can't locate this field, email Support on **support@worldpay.com**. Click the **Save changes** button.

If you have specified a secret, you must sign all transactions correctly otherwise they are rejected. If you wish to disable the MD5 functionality at any point, simply remove the secret key value from your installation.

3. **Specify your protected parameters to us.** These parameters must be entered into the **SignatureFields** field in the Integration Setup for your installation, and must match the value in your shopping cart.

9.2.2 If you use a bespoke setup

To add MD5 encryption to your site if you have built a bespoke store, you need to do the following:

*You can also set up MD5 encryption on the Installation Administration page of the Merchant interface. For more information, see the **Customising Guide (Advanced)**.*

1. **Choose an 'MD5 secret' to encrypt the parameters you wish to protect.** This should be a string (spaces are permitted) of 20 to 30 characters, known only to yourself and to us. Capitals, numerals and punctuation are all permitted.

2. Set your **MD5 secret** in the Merchant Interface. To do this, enter this value into the **MD5 secret for transactions** field in the Integration Setup for your installation using the Merchant Interface > Installations option. If you can't locate this field, email Support on **support@worldpay.com**. Click the **Save changes** button.

*If you have specified a secret, you **must** sign all transactions correctly otherwise they are rejected. If you wish to disable the MD5 functionality at any point, simply remove the secret key value from your installation.*

3. Choose which of the order details parameters you wish to protect with the signature. You can encrypt whichever parameters you like, including custom parameters (C_, M_ and MC_).You must specify these as a colon-separated list in the Integration Setup for your installation, using the Merchant Interface.

We recommend that you include: amount, currency, instId and cartId parameters.

```
    Example: instId:amount:currency:cartId:MC_userId
```

4. Construct a string consisting of the secret and the values of the parameters you want to encrypt. The items in this string should be separated by colons. For example, if you choose the phrase Lions&2Tigers&3Panthers as your secret, and the fields you have chosen to encrypt have the following values: amount=123.00, currency=GBP and cart Id=ABC123, your string should look like this:

```
    Lions&2Tigers&3Panthers:123.00:GBP:ABC123
```

It is vital that you list the parameter values in the same order in which you listed them in the SignatureFields parameter.

5. Calculate an MD5 signature from this string, and include the signature in the order details. The MD5 algorithm returns a hexadecimal value 32 characters long. You should include this digital signature in your order details submission as the value of a parameter named 'signature':

```
       <input type="hidden"name="signature" value="58e41db32a6f2ff9c3c96eea6583ffbd">
```

*The way you use MD5 encryption for a bespoke setup depends on the code used to build your store. Regrettably, we are not able to help you with incorporating the code into your store, but can provide you with instructions on how to send that information to us. For more information about MD5, see section **MD5 reference sites below**.*

9.3 Additional security

As an additional safety feature in HTML Redirect, you can set a time period for which a transaction is valid. This limits the amount of time available for an unauthorised user to try to decrypt the transaction.

To specify this time limit, use the parameter authValidTo. The value of authValidTo should be a date and time given in the Unix time standard: seconds since 1st January 1970. For example,

```
<input type="hidden" name=authValidTo value="938736000000">
```

15 minutes is a commonly-used period of validity, but you might choose to reduce this time period to 10 or even five minutes.

To implement this feature you should:

1. Ensure the clock on your server is synchronised to an accurate time source (for example, by using NTP at **www.ntp.org/**).
2. Take a time-reading from your server at the time of the transaction, and convert it to the Unix time standard, if necessary.
3. Multiply the number of seconds you want the order details to be valid for by 1000, and add this to the time. This will be the value of `authValidTo`.

9.3.1 MD5 reference sites

Click the following links to find out more about MD5 encryption, and how to use it:

- **http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/md2-md4-and-md5.htm**: RSA Laboratories definition of MD5
- **http://userpages.umbc.edu/~mabzug1/cs/md5/md5.html**: details on how to implement MD5 in many program languages
- **www.faqs.org/rfcs/rfc1321.html**: specification of the MD5 function provided by the inventor at MIT

Appendix A: Troubleshooting

This section describes some common error messages and problems.

I copied the HTML example code and it doesn’t work

If you have problems using the example HTML code, check that:

- You have copied the code correctly. In particular, take care with spelling the parameters and note that they are case-sensitive: INSTID is not the same as instId
- You have updated the HTML code to use your own installation rather than the example installation
- The punctuation and spacing of your code matches the example HTML code

For a full list of parameters and their descriptions, see ***HTML Redirect parameters* on page 35**.

I received a ‘no suitable accounts for this purpose’ message

This error message can be generated in response to a premature attempt to submit order details to the Production Environment before your merchant code (account) has been enabled.

Initially, your live merchant code is not enabled for transactions, so there is no risk of accidentally triggering a live transaction until you and we have agreed that you are ready to go live. You can develop and test your site using the test environment, which behaves in the same way as the live Production Environment, except that no money is actually transferred when you submit order details.

If you try to submit order details to the Production Environment before you go live, you receive this error message. You may also get the error message, "No currency value/variable is being passed across".

For more information about the test environment, see the **Test and Go Live Guide**.

I get a processing error when I attempt a transaction

This error message usually occurs while a merchant is in the process of integrating.

The most common causes are that an unknown character is being included in the values for the parameters, or no value is being supplied at all. For example, consider the following:

```
<input type="hidden" name="instId" value="#12345">
```

The # symbol would cause an error to occur when the content of the form is posted to our payment service, because the server is not expecting the value of `instId` to contain a non-numeric character.

Some of my chosen currencies are missing

If some of your chosen currencies are missing from the Currency Selector Box in the payment page, please check that you are using the correct Installation ID for the transaction. If this is correct, then your currencies may not have been added to our system; in this case, please contact us.

The test environment displays all currencies. To check the available currencies, use your production environment.

For more information on currencies, see section ***Using different currencies* on page 10**.

Appendix B: ISO currency codes

The currencies accepted by the Worldpay payment service are listed in the table below.

For the full ISO 4217 list of ISO currency codes, see ***http://www.iso.org****.*
Worldpay does not take responsibility for an external link's operation or content.

ISO currency codes

Currency	ISO currency code	Exponent
Nuevo Argentine Peso	ARS	2
Australian Dollar	AUD	2
Brazilian Real	BRL	2
Canadian Dollar	CAD	2
Swiss Franc	CHF	2
Chilean Peso	CLP	0
Yuan Renmimbi	CNY	2
Colombian Peso	COP	2
Czech Koruna	CZK	2
Danish Krone	DKK	2
Euro	EUR	2
Pound Sterling	GBP	2
Hong Kong Dollar	HKD	2
Hungarian Forint	HUF	2
Indonesian Rupiah	IDR	2
Iceland Krona	ISK	0
Japanese Yen	JPY	0
Kenyan Shilling	KES	2
South Korean Won	KRW	0
Mexican Peso	MXN	2
Malaysian Ringgit	MYR	2
Norwegian Krone	NOK	2
New Zealand Dollar	NZD	2

Currency	ISO currency code	Exponent
Philippine Peso	PHP	2
New Polish Zloty	PLN	2
Swedish Krone	SEK	2
Singapore Dollar	SGD	2
Thai Baht	THB	2
New Taiwan Dollar	TWD	2
US Dollar	USD	2
Vietnamese New Dong	VND	0
South African Rand	ZAR	2

Table 14: ISO currency codes

Appendix C: ISO country codes

The country parameter value is an upper-case two-letter ISO 3166 standard country code, as shown in the following example:

```
<input type="hidden" name="country" value="GB">
```

Code example 8: Country parameter

For the full ISO 4217 list of ISO country codes, see ***http://www.iso.org****.*
Worldpay does not take responsibility for an external link's operation or content.

Appendix D: Test card numbers

You can use the following credit / debit card numbers to test transactions in the test environment only. When using test cards, you can specify an expiry date up to seven years in the future. The test cards do not have a card verification code and issue number.

For more information about testing your HTML Redirect integration, see the ***Test and Go Live Guide****.*

Test card numbers

Card type	Test card number
Airplus	122000000000003
American Express	34343434343434
Cartebleue	5555555555554444
Dankort	5019717010103742
Diners	36700102000000 and 36148900647913
Discover card	6011000400000000
JCB	3528000700000000
Laser	630495060000000000 and 630490017740292441
Maestro	6759649826438453 and 6799990100000000019
MasterCard	5555555555554444 and 5454545454545454
Visa	4444333322221111, 4911830000000 and 4917610000000000
Visa Debit	4462030000000000 and 4917610000000000003
Visa Electron (UK only)	4917300800000000
Visa Purchasing	4484070000000000

Table 15: Test card numbers

Appendix E: Payment method codes

The table below shows the valid payment method codes for the paymentType parameter, which must be included in an order details submission to cause the Payment Selection page to be skipped or bypassed.

Payment method	paymentType
Amex	AMEX
Diners	DINS
ELV	ELV
JCB	JCB
Laser	LASR
Mastercard	MSCD
Mastercard Debit	DMC
Maestro	MAES
Visa Card	VISA
Visa Delta	VISD
Visa Electron	VIED
Visa Purchasing	VISP
V.me by Visa	VME

Table 16: Payment method codes

Appendix F: Important Changes from VISA Europe

Important Changes from VISA Europe for MCC 6012 (Financial Services) payments

From the 1st of June 2014 we have updated the Worldpay Payment Gateway (WPG).

This is to meet a requirement from VISA for merchants with the Merchant Code Classification 6012 for financial services. The update means your shoppers who pay with a VISA card in British pounds must complete four extra fields. You pass the information from the fields to the acquirer as part of your integration to Worldpay. The extra four parameters are:

Parameter	Description
shopperAdditionalAccountNumber	This field can be empty. It contains a maximum of 10 characters. It must contain only letters or digits – no special characters such as ^ % * and similar. This is the partial account number or the full account number of the shopper who has an agreement with you. This shopper is called the primary recipient and the account number can be a bank account, a mortgage ID or a personal reference number. You can also send up to the first 10 characters from the account number. This can be a partial PAN (Primary Account Number) number: so you must send the first 6 digits + last 4 digits, for example FFFFFFLLLL. If the value of this field does not follow these rules, you receive this error message: The shopperAccountNumber cannot be longer than 10 characters and must contain only digits and letters.
shopperAdditionalLastName	This field can be empty. It must not contain digits. If the value of the shopperAdditionalLastName does not follow the rules above, you receive this error message: The shopper additional last name must not contain digits.
shopperAdditionalBirthDate	This must be a valid date in the past. Format is YYYY-MM-DD. Date must be from the past. Can be empty. If the value of shopperAdditionalBirthDate is not correct, you receive the following error message:

Parameter	Description
	The shopper additional birth date must contain a valid date in the past and in YYYY-MM-DD format.
shopperAdditionalPostalCode	Can be empty. Must be a valid UK postcode (one of the format: AA9A 9AA, A9A 9AA, A9 9AA, A99 9AA, AA9 9AA, AA99 9AA). Postcodes must have a space between the first and last group of characters/numbers. If the value of shopperAdditionalPostalCode is not correct, you receive this error message: The shopper additional postal code must contain a valid UK postal code.

Table 17: MCC 6012 parameters

9.3.2 Example of valid data

```
<INPUT TYPE="hidden" NAME="shopperAdditionalAccountNumber" VALUE="444433111">
<INPUT TYPE="hidden" NAME="shopperAdditionalLastName" VALUE="Doe">
<INPUT TYPE="hidden" NAME="shopperAdditionalBirthDate" VALUE="1976-03-02">
<INPUT TYPE="hidden" NAME="shopperAdditionalPostalCode" VALUE="CB4 0WE">
```

Code example 9: Valid MCC 6012 parameters

Appendix G: Changes to this Guide

Revision	Release date	Changes
5.2	September 2015	Converted: • Original Word file to Madcap Flare
5.1	September 2014	Updated: • Reformatting. Some rewriting • Removed temporarily MasterPass digital wallet service information.
5.0	July 2014	Updated: • Rebranded. • Minor bug fixes • Added information about the MasterPass digital wallet service.
4.6	April 2014	Added: • MCC 6012 information for British VISA payments.
4.5	January 2014	Updated: • Information about testing ELV payments.
4.4	September 2013	Updated: • Minor bug fixes.
4.3	August 2013	Added: • Information about the V.me by Visa digital wallet service.
4.2	March 2013	Updated: • Test card numbers.
4.1	January 2013	Updated: • Minor bug fixes.
4.0	December 2011	Updated: • Gateway and guide name added to navigation path. • Fix for "hidden" parameter.
3.1	October 2011	Updated: • Payment pages.
3.0	July 2011	Updated: • Style and formatting for Worldpay rebrand.

Code example 10: Changes to the guide

Contact us

To find out more, get in touch with your Relationship Manager or:

- Email **corporatesupport@worldpay.com**