

Perl Module Guide: Finance::BankVal::UK

Contents

Introduction.....	1
Finding and Installing the Module.....	1
Decompressing/Unpacking the Module.....	1
<i>Windows Systems.....</i>	<i>1</i>
<i>Linux/UNIX Systems.....</i>	<i>1</i>
<i>Build and Install.....</i>	<i>2</i>
Using the Module.....	3
Importing the BankValUK Module.....	3
Calling the Services.....	3
<i>As a Class Method.....</i>	<i>3</i>
<i>As an Object Method.....</i>	<i>3</i>
<i>Parameters.....</i>	<i>4</i>
Returns from the Service.....	5
<i>BankValUK.....</i>	<i>5</i>
Using the LoginConfig file.....	6
Further Information.....	6
The Extended Industry Sort Code Directory.....	6

Introduction

This document describes the operations and uses of Unified Software's Perl module, Finance::BankVal::UK.

The Finance::BankVal::UK module handles REST web service calls to Unified Software's BankValUK web service. The module handles all aspects of calling the web services requiring one simple call to its exposed BankValUK sub. In the unlikely event of any problems with Unified Software's main server centre, this module will automatically call the backup servers.

Installing this module allows the easy integration of the BankValUK service with other code. This document explains how to install and use the module.

Finding and Installing the Module

The module can be found on CPAN (Comprehensive PERL Archive Network) at <http://search.cpan.org/> , by searching for Finance::BankVal::UK.

The easiest way to install is using cpan minus by typing

```
cpanm Finance::BankVal::UK
```

Or use cpan my entering the cpan command line and typing

```
install Finance/BankVal/UK.pm
```

Alternatively you can install it manually by downloading and then unpacking and unzipping the module.

Decompressing/Unpacking the Module

Depending on whether you are on a Linux /UNIX or Windows based system you should follow the respective guide below:

Windows Systems

The package can be unpacked and unzipped in one stage by using a freeware application such as WinZip.

Linux/UNIX Systems

On a Linux/Unix system the operation will take two stages

- First decompress by typing “gzip -d BankVal-UK-0.5.tar.gz” at the command console.
- Next unpack by typing “tar -xof BankVal-UK-0.5.tar” at the command console.

Build and Install

There is no requirement to compile the BankVal modules, simply place the BankVal folder into any folder on PERL's @INC path. Generally these will be a lib folder e.g. **C:\Perl\lib** or **C:\Perl\site\lib** on Windows.

Further information on installing modules from CPAN can be found from the following website <http://www.cpan.org/modules/INSTALL.html>.

Using the Module

Importing the BankValUK Module

To include the BankValUK module within your code you should:

- First, ensure that the module (with its folder structure) is in a location that is on your PERL @INC path, or add the following line to your code:

```
use lib <<ABSOLUTE PATH TO BANKVALUK>>;
```

- Second, import the bankValUK subroutine by adding the following line to your code:

```
use Finance::BankVal::UK qw(&bankValUK);
```

- Alternatively to use the BankVal::UK module as an object add the following line to your code:

```
use Finance::BankVal::UK;
```

Following these two steps the BankValUK service is accessible from within your code.

Calling the Services

As a Class Method

The services can be called by using the following line:

```
my $ans = bankValUK(@params);
```

As an Object Method

First create an instance of the BankVal::UK class, then call the bankValUK method as shown below:

```
my $obj = Finance::BankVal::UK->new();  
my $ans = $obj->bankValUK(@params);
```

In the previous examples @params is an array which would need to be populated with the relevant parameters, for example see below:

```
my $sortcode = "12-12-12";  
my $account = "12345678";  
my $UserID = "abcd123";  
my $PIN = "12345";  
  
my @params = ($sortcode,  
$account,$UserID,$PIN)
```

Parameters

There are a number of different parameter combinations which can be passed to the bankValUK function, these are:

- 1: Sort code - 6 digit number either 00-00-00, 00 00 00 or 000000
- 2: Account no. - 6 to 12 digit (unseparated i.e. 00000000)
- 3: UserID - available from www.unifiedsoftware.co.uk
- 4: PIN - available from www.unifiedsoftware.co.uk

The order of the parameters **must** always be as shown above.

The UserID and PIN can also be stored in the LoginConfig.txt file bundled with this module, the use of this file saves passing the PIN and user ID data with each call to bankValUK, for further information on this see the 'Using the LoginConfig file' section of this document.

The allowed parameter lists are as follows:

- To call **BankValUK** and supply UserID and PIN within the call:

```
bankValUK('$sortcode', '$account', '$UserID', '$PIN');
```

- To call **BankValUK** if the *LoginConfig.txt* file is being used:

```
bankValUK('$sortcode', '$account');
```

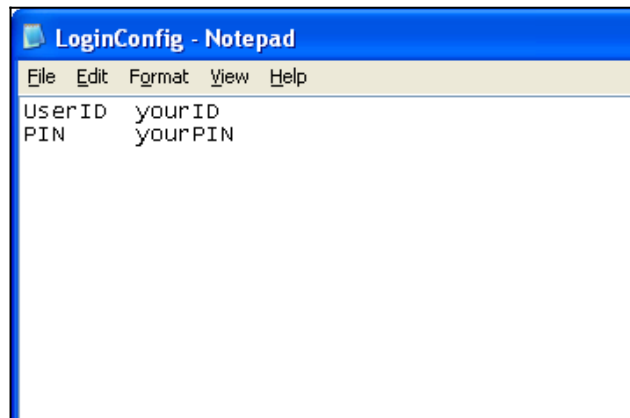
Returns from the Service

BankValUK

BankValUK will validate UK bank accounts and sort codes and return the transposed sort code and account number plus the latest EISCD (Extended Industry Sorting Code Directory) data in JSON format.

Using the LoginConfig file

If no UserID and PIN information is passed to the *bankValUK* sub within the BANKVAL::UK module, the module will attempt to load information from the **LoginConfig.txt** file. To use this file simply copy it to the same folder that your code is stored in, open it and fill out your UserID and PIN as shown below:



Following this it will be unnecessary to pass the UserID and PIN with the call to the BANKVAL::UK module. If your call is still made with UserID and PIN parameters, then the parameter data will be used rather than the LoginConfig file data.

Further Information

The Extended Industry Sort Code Directory

The Extended ISCD - Industry Sort Code Directory is the definitive UK bank branch and building society database. This sort code database is maintained centrally by VocaLink (formerly BACS) on behalf of the UK payments industry and it contains information about every bank branch in the UK.