

A database called “*bankofmunster*” (script available on Moodle) contains one table called “*accounts*”. The accounts table contains bank account information for a number of account holders.

The structure of the accounts table is as follows:

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	<u>AccountNumber</u>	int(11)			No	None		Change Drop Primary
2	FirstName	varchar(50)	latin1_swedish_ci		No	None		Change Drop Primary
3	LastName	varchar(50)	latin1_swedish_ci		No	None		Change Drop Primary
4	SortCode	varchar(10)	latin1_swedish_ci		No	None		Change Drop Primary
5	Balance	double			No	None		Change Drop Primary

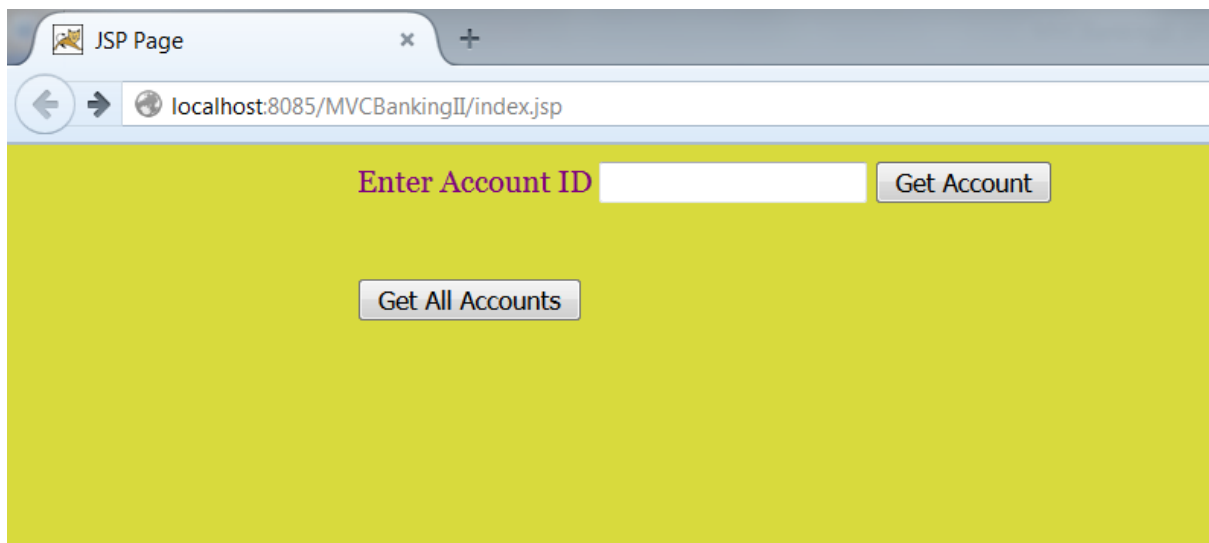
Its contents are as follows:

AccountNumber	FirstName	LastName	SortCode	Balance
12345	Alan	Ryan	34-34-98	9000000.01
32456	Seamus	Doyle	76-55-08	-567
34456	Gerry	Guinane	34-45-34	1799
54321	Brendan	Watson	90-34-35	-97318.23
78655	Ita	Kavanagh	89-32-30	9876
98765	Tom	Costello	56-34-34	2799.34
363663	Elizabeth	Bourke	39-32-65	100000
583822	Carol	Rainsford	90-34-35	98763
7854744	Michael	Winterburn	34-45-34	9788
9082834	Desmond	OCarroll	87-45-54	-98

To Do: Write a MVC compliant web application whose homepage will afford the user two options.

1. The ability to search for a particular bank account (using the account number as a search criteria).
2. Provide a link that when followed will display all account information.

The home page for the application might look like the following:



Enter Account ID

When the user clicks on the “Get All Accounts” button – all account details within the accounts table are to be displayed in a table. Any accounts which have a negative balance must be highlighted in red. The account balances must be formatted correctly too.



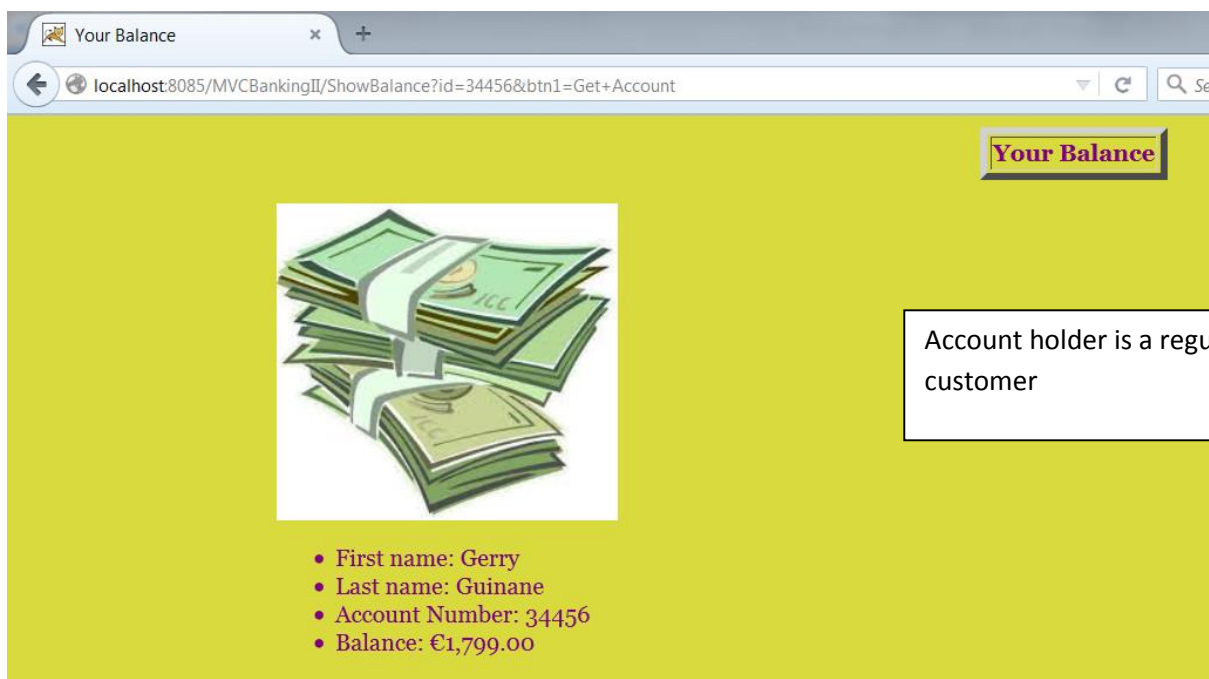
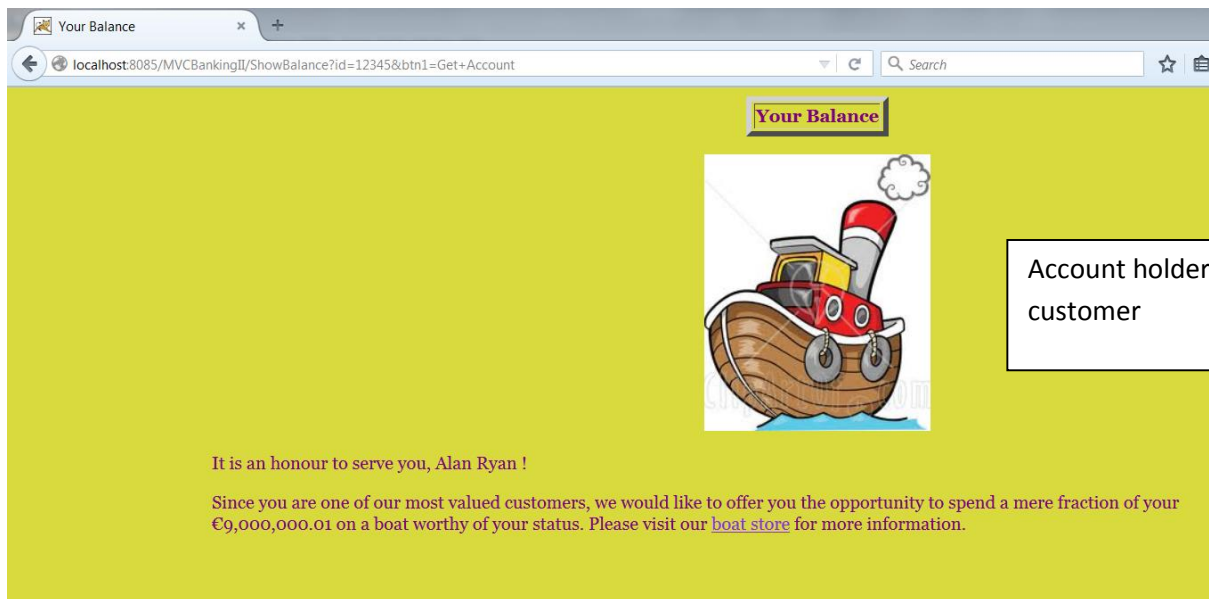
Account ID	First Name	Last Name	Sort Code	Balance
12345	Alan	Ryan	34-34-98	€9,000,000.01
32456	Seamus	Doyle	76-55-08	€569.00
34456	Gerry	Guinane	34-45-34	€1,799.00
54321	Brendan	Watson	90-34-35	€97,318.23
78655	Ita	Kavanagh	89-32-30	€9,876.00
98765	Tom	Costello	56-34-34	€2,799.34
363663	Elizabeth	Bourke	39-32-65	€100,000.00
583822	Carol	Rainsford	90-34-35	€98,763.00
7854744	Michael	Winterburn	34-45-34	€9,788.00
9082834	Desmond	O'Carroll	87-45-54	€98.00

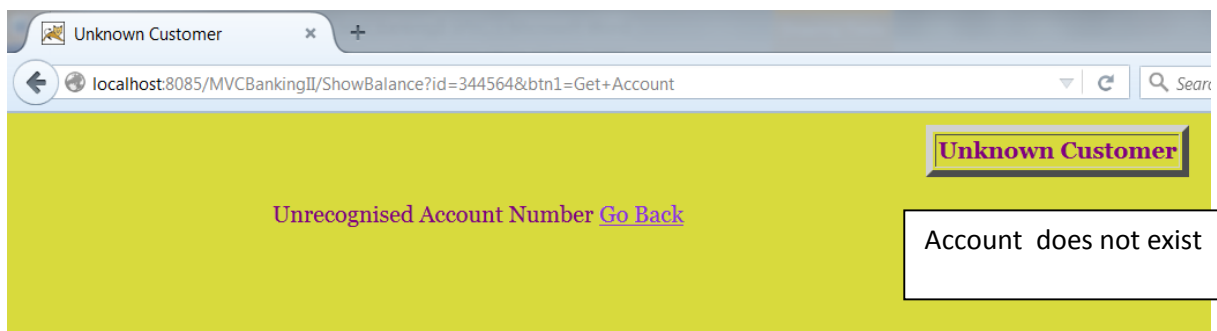
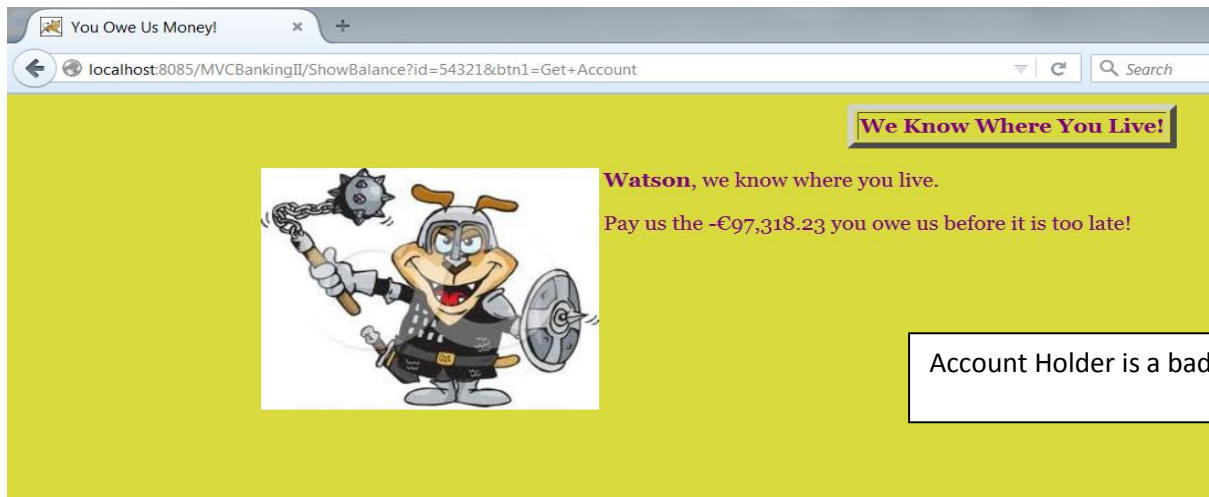
Assume that within the banking system customers/account holders are classified into one of three categories:

1. A Bad Customer (a bank account with a negative balance).
2. A Regular Customer (a bank account with a balance between 0 and 10000).
3. An Elite Customer (a bank account with a balance greater than 10000).

When the user searches for a particular account, ensure that when the account details are displayed, the category that the customer belongs to is reflected in the page that is subsequently displayed. If

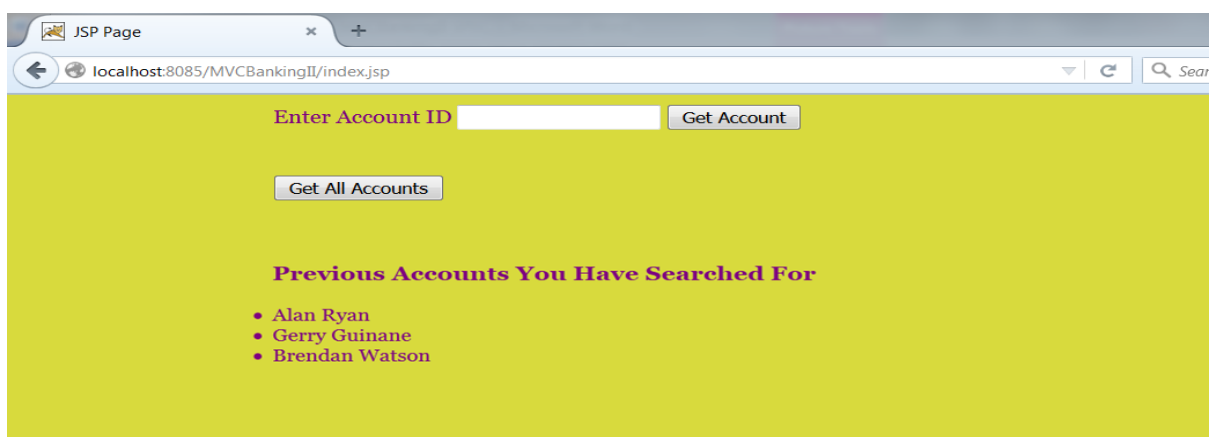
the user searches for a non-existent customer ensure that a relevant error message is displayed.
Consider the following screen grabs.





Enhancements:

1. Include functionality whereby any previously searched for customers/accounts are displayed in a bulleted list on the homepage. How would you approach that?



You would need to ensure that this information is displayed every time a user views the homepage.

2. When the list of accounts currently within the bank are displayed (in the table), enable each account number so that it is a link, that when clicked on will (take the user to a new page and) allow the user to increase/decrease the balance for that account.