

INTERNSHIP PROJECT C++: BANKING RECORD SYSTEM

DESCRIPTION OF THE PROJECT:

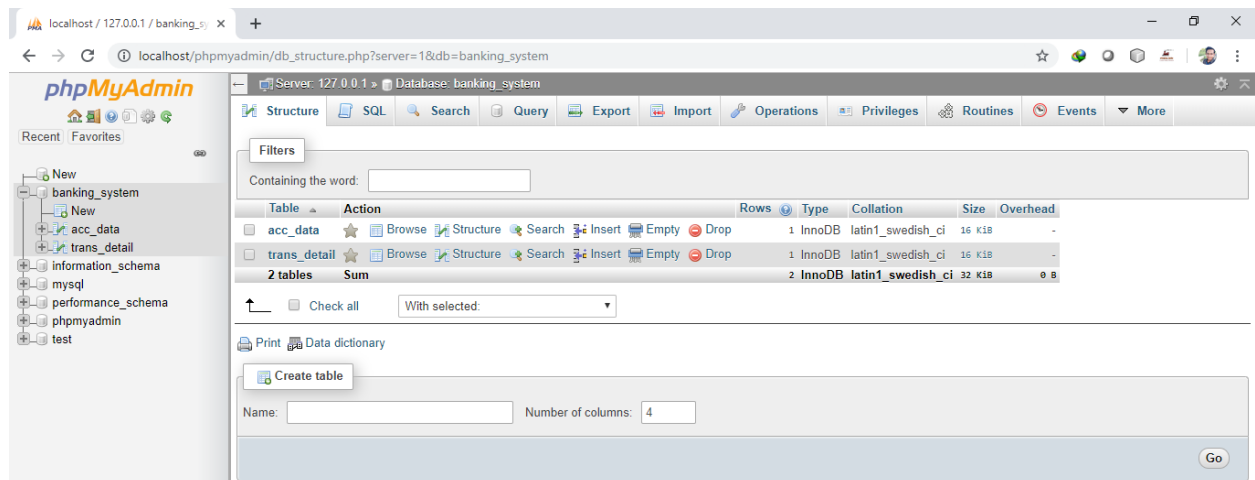
1. **Add Record:** User inputs information such as account number, first name, last name and money to be entered.
2. **Show Balance:** User inputs information such as account number, first name, last name, and output is his/her account balance
3. **Withdraw Record:** User inputs information such as account number, first name, last name and money to be withdrawn.
4. **Search a particular transaction number** and output is user's account information with the date & time of the transaction
5. Use **MySql For Database Connections.**

SOFTWARE USED

- Codeblocks 16.0.1 (minGW compiler)
- Xampp server
- Dev CPP

XAMPP LOCAL SERVER AND DATABASE DESCRIPTION

It has a database name “banking_system” that contain two tables viz. acc_data and trans_det. Various decription of the database is in following images.



Server: 127.0.0.1 » Database: banking_system » Table: acc_data

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	account	bigint(20)			No	None			Change Drop More
2	firstName	text	latin1_swedish_ci		No	None			Change Drop More
3	lastName	text	latin1_swedish_ci		No	None			Change Drop More
4	balance	float			No	None			Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

Server: 127.0.0.1 » Database: banking_system » Table: trans_detail

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	account	bigint(20)			No	None			Change Drop More
2	transNo	bigint(20)			No	None			Change Drop More
3	date	datetime		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP		Change Drop More

Check all With selected: Browse Change Drop Primary Unique Index Fulltext Add to central columns Remove from central columns

CODE EXPLANATION

The code has four functions to accomplish given four tasks. These functions are-

1. void addRecord();
2. void showBalance();
3. void withdrawRecord();
4. void searchTransaction();

MAIN FUNCTION:

Main function of the program uses switch case to ask user to enter any number from the given choices. According to the input of the user, any of the four functions listed above are called and they work accordingly, the working of these functions are explained below. This also uses labels that are responsible for giving choices again to the user after completing one task. Also the user has choice as “EXIT” from the program that directly takes the user out of the program and close it.

```

"F:\b tech\temp\bin\Debug\temp.exe"
*****
Banking Record System
1. Add Record
2. Show Balance
3. Money Withdrawal
4. Search transaction
5. Exit

Enter your choice:

```

WORKING OF DIFFERENT FUNCTIONS

1. ADD RECORD

This option calls `addRecord()`; which first creates a connection variable 'conn'. It makes a connection between the program and the database by the help of `mysql_real_connect()`, if the connection is established the user is asked to input the account details and based on this input an `INSERT` query is executed to insert data in table `acc_data` in the database `banking_stem`. If the connection is failed to established "Not Connected" is printed on the screen.

```
"F:\b tech\temp\bin\Debug\temp.exe"
*****
                        Banking Record System
1. Add Record
2. Show Balance
3. Money Withdrawal
4. Search transaction
5. Exit

Enter your choice: 1

Enter account number:372846

Enter first name:yash

Enter last name:s

Enter amount:2323.50

Record inserted...
*****
```

2. SHOW BALANCE

This option calls `showBalance()`; which first creates a connection variable 'conn'. It makes a connection between the program and the database by the help of `mysql_real_connect()`, if the connection is established the user is asked to input account details. Based on the details given by the user a `SELECT` query is executed which compares first name, last name and account details to the data present in table `acc_data` of the database, if these are matched with any record then the available balance of the account is displayed on the screen.

```
"F:\b tech\temp\bin\Debug\temp.exe"
*****
                        Banking Record System
1. Add Record
2. Show Balance
3. Money Withdrawal
4. Search transaction
5. Exit

Enter your choice: 2

Enter account number:372846

Enter first name:yash

Enter last name:s

Account Balance is 2323.5
*****
```

3. MONEY WITHDRAWAL

This option calls showBalance(); which first creates a connection variable 'conn'. It makes a connection between the program and the database by the help of mysql_real_connect(), if the connection is established the user is asked to input account details and the amount to withdraw. Based on the details an UPDATE query is executed which subtract the balance from the current amount and at the same time a random number is generated with the help of "rand" function. This random function now acts as a transaction number for the current transaction that the user has presently done. This transaction number is stored in another table trans_detail in the same database banking_system using INSERT query along with the account number and the current date and time of the server for further information. After all this queries and processing a success message along with that random number is printed on the screen for the user.

```
Select "F:\b tech\temp\bin\Debug\temp.exe"
*****
                        Banking Record System
1. Add Record
2. Show Balance
3. Money Withdrawal
4. Search transaction
5. Exit

Enter your choice: 3

Enter account number:372846

Enter first name:yash

Enter last name:s

Enter money to be withdrawn:1500

Successfully withdrawn!.
Transaction Number is 2000008
*****
```

4. SEARCH TRANSACTION

This option calls showBalance(); which first creates a connection variable 'conn'. It makes a connection between the program and the database by the help of mysql_real_connect(), if the connection is established the user is asked to input transaction number only. Now a SELECT statement is executed along with SQL JOIN. This query is executed to fetch account details of the customer who has done the transaction along with the date and time of the particular transaction. But these all data are stored in two different tables which are linked to each other by the account number, therefore JOIN is used. These details are printed on the screen if these are found in the database.

```
"F:\b tech\temp\bin\Debug\temp.exe"
*****
                        Banking Record System
1. Add Record
2. Show Balance
3. Money Withdrawal
4. Search transaction
5. Exit

Enter your choice: 4
Enter Transaction Number:2000008

Account number: 372846
First Name: yash
Last Name: s
Current Balance: 823.5
Date and Time of Transaction: 2019-05-17 23:21:00
*****
```

5. EXIT

This choice uses exit function to get out of the main function and shut the program.

REFERENCE

- Eckovation support team
- www.youtube.com

Made By: *Yash*

Email: yash754311@gmail.com

Mob: 9125041358