

Banking System
Arrenn Baral, Prakriti Basyal
Team 10
CS 157A Summer 2020
Instructor Mike Wu

Project Description

This project will be a database application that is based in the general banking system.

The goal of this application is to provide a convenient way to access user's banking information. Users will be able to login with designated username and password, access bank account, deposit, withdraw, get a report of transaction if a user wishes to. This application will also provide a user-friendly interface that is clear and intuitive. We use bank system in our daily lives. We login into bank systems like Chase, Wells Fargo, Bank of America, and list goes on. Deposits, withdraws, transfers, statements are some of the example of our daily used operations on the bank systems. But we never thought what actually runs behind the screen. Since we are taking database class, we are getting great opportunity to explore system behind the screen. We got the motivation from our daily used operations in the bank system. Let's start with simple structure our structure of the banking system

Word on structure

We will be Apache Tomcat webserver to run our banking system.

We will be implementing our code through Java IDE.

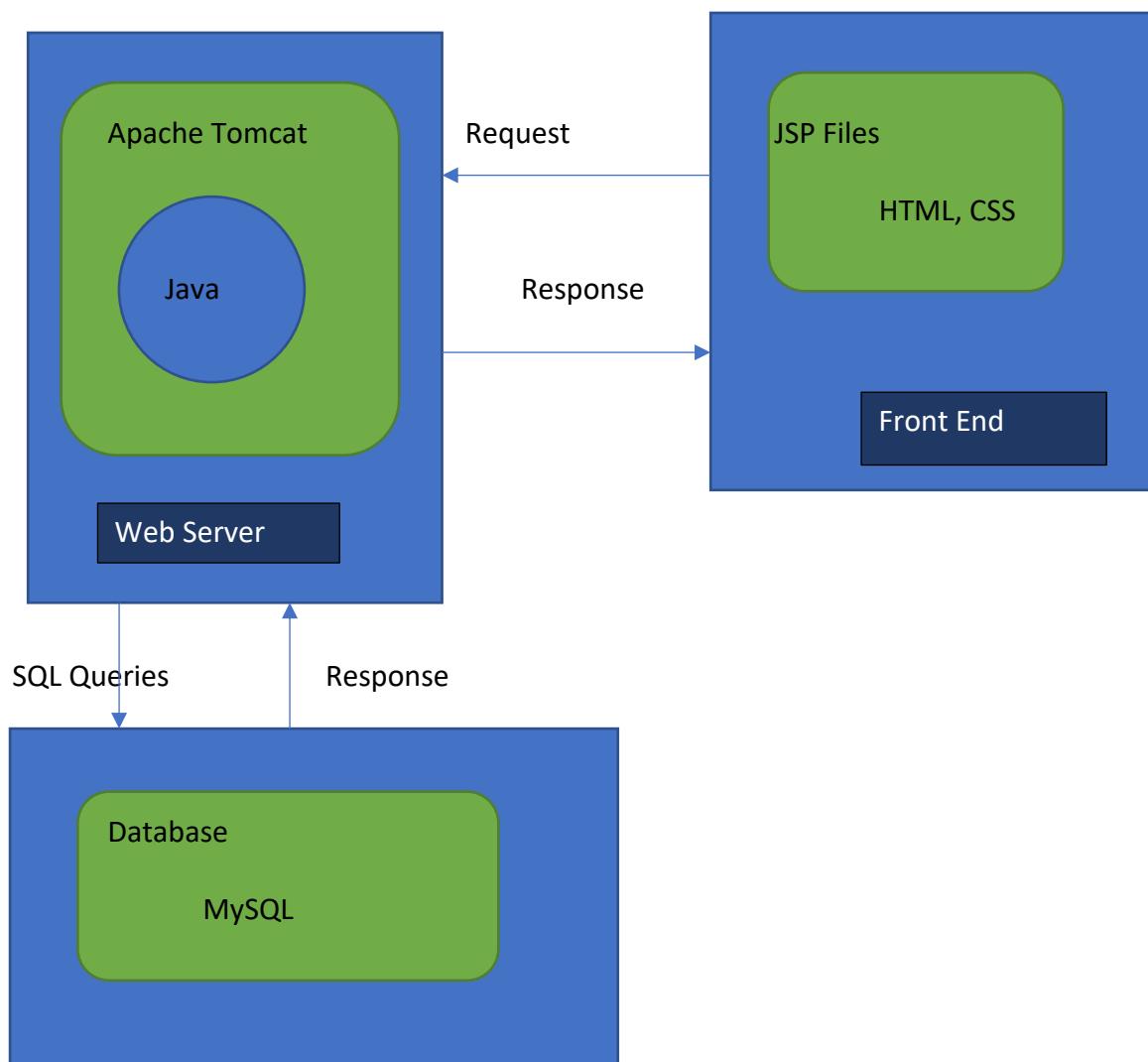
For database system, we are using MySQL but data entries.

We will be using DMLs, and DDLs for operations like deposits, withdrawals, and get statement.

We will be using HTML and CSS for beautify our banking system. It is important to give user-friendly look to users.

System Environment

Structure of the System



Hardware and Software

Apache Tomcat

MacOS

RDMS

MySQL Community Server 8.0.20

Application Languages

Java, HTML, CSS, PHP (maybe), SQL

Functional Requirements

How user will access the system

The system provides functionality to create an account. User will be asked to set up a new account if he/she is new user else simply log in using their existing user credential in the database. If user wishes to setup a new account, the system opens new page for the user where the system asks several information for setting up the account such as First name, Last name, Date of Birth, Email Address, Gender, username, and password. Once the setup process completed, user will be redirected to banking system where they can select different operations. User cannot login if user's credential is not saved in the database. User cannot access without setting up the account. If user tries to access with fake username and password, they will be redirected to login page to use valid username or password. However, once user logs in, system will not ask to sign in again. System will assume user logged in through trusted device. Once logged in, users will be able to use several functions.

Functions

Log in

User shall set up an account to get an access to the banking system.

User's credentials must be saved in the database.

Withdraw

Once logged in, user shall select the withdraw option to get their money.

Once user chooses this function, the withdrawal sum will be deducted from the total amount.

Deposit

Once logged in, user shall use this function to deposit the money.

Once deposited, the deposited sum will be added to the total amount.

Statement

User shall choose this function to get a statement of the transaction after successful withdrawal and deposits.

Apply for Loan

User shall use this function to apply for a loan.

User gets approved only if thousand dollars is available in his bank account.

Order a cheque book

User can use this function to order a cheque book.

There will be a charge of \$1.00 for ordering a check.

Manage/update account information

User shall use this function to edit account information such as address, and phone number.

Transfer fund

User shall use this function to transfer fund to other.

Receiving person must be setup before transferring.

Search statement by date

If user wants to search statement from specific date, they can choose this function to select the date and search.

Sort the transactions

If user wishes to sort the transactions from higher to lower or lower higher, they can use this function to sort as wished.

Delete account.

If user wishes to terminate the account, they can simply use this function to delete the account. Once deleted, account will not be recovered. It will be permanent.

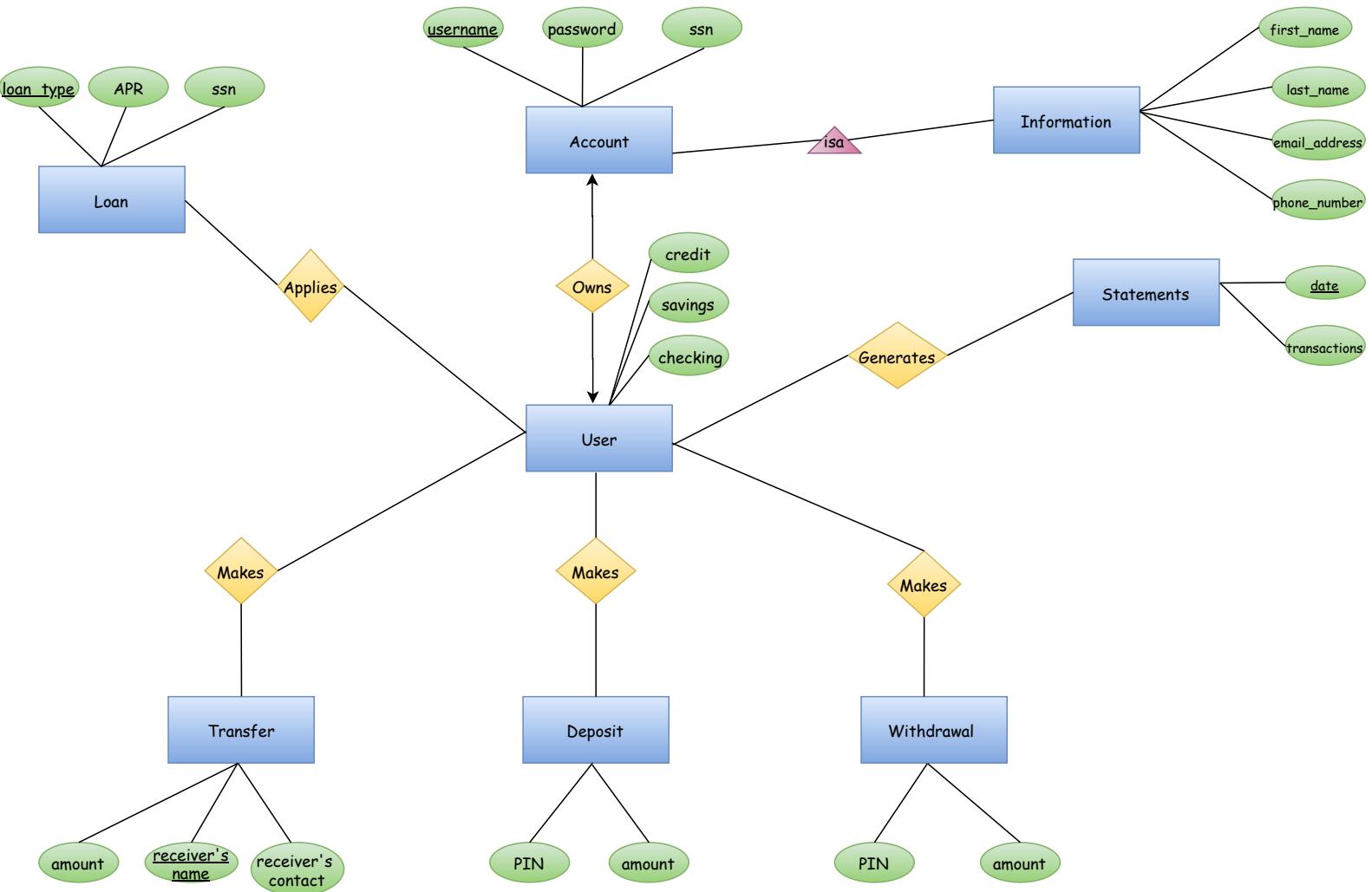
Display Account Information

The user will be able to view the account information through this function.

Graphics User Interface (GUI)

For the GUI, we will be using HTML to provide user-friendly environment. If possible, we will allow user to upload a picture of them for the account.

The security of each user account will be protected by a username and password created by the user. The username for each account will not be their email. User name will be the one which they choose while setting up the account. The user account information will be securely stored in the server. In order to login to the system the user must provide an existing username account along with the associated password. The website will be implemented using https providing an encrypted connection between user and server.



Account:

Account entity is the service offered by the bank where one may deposit, withdraw or transfer money. Every customer has a unique account. The entity set contains the customers username and password to login and the ssn. Username is primary key.

Loan:

Loan entity represents lending money to one or more individuals with the required interest rate from the bank. There are different types of loan. Loan_type, APR, and ssn are the attributes, where loan_type is the primary key.

Information:

Information entity represents the unique customer arche type information. Information has the basic information or data of the customer. It has customer's first_name, last_name, email_address and phone_number. This entity set is sub-class of the account entity set.

Statements:

Statement entity represents a record of the financial activities done by the customer which are also given as the written reports. It basically keeps tracks the record list of transaction done by the customer over a certain period of time. So, date and transactions are in the attributes.

User:

User is one of the entities sets. It keeps the records of user's account summary. User can review their balance. Credit, savings, and checking are the attributes of this entity set.

Transfer:

This entity set represents the electronic fund transfer from one person to another to the bank accounts. It is one of the services provided by the bank so that people can transfer making people's life easier. This entity set has the amount, receiver's name and receiver's contact so they can transfer directly to their account.

Deposit:

Deposit is entity set which is available to customer to deposit money into their bank account. These deposits can be made through cash, check by the customer. So, for the security of the customer's information. Deposit entity set has the PIN and the amount as an attribute.

Withdrawal:

Withdrawal entity set represents giving the customer an option to withdraw money whenever they want. The bank grants access to the customer for the withdrawals except for the unusual activities. This entity set contains withdraw amount and the PIN number to withdraw from their account as an attribute.

UserAppliesLoan

UserAppliesLoan is a relationship that connects Users and Loan entity sets. This is many to many relationship because there can be many users and apply for different loan amount.

UserGeneratesStatement:

UserGeneratesStatement is a relationship that connects the user and statement table. Here, the various user looks for various transaction according to the date and this is also many to many relationships.

UserMakesDeposit

UserMakesDeposit is a relationship that connects the User and the Deposit entity sets. This is a many to many relationships since there can be several users making several deposits.

UserMakesTransfer

UserMakesTransfer is a relationship that connects the User and the Transfer entity sets.

This is also many to many relationship. Here the user transfers the amount to the receiver's name.

UserMakesWithdrawal

UserMakesTransfer is a relationship that connects the User and the Withdrawal entity sets. This is a many to many relationship because many users can withdraw certain amount of money at any time.

The screenshot shows the MySQL Workbench interface. The left sidebar displays the 'Banking System' schema with various tables like Account, Deposit, Information, Loan, Statements, Transfer, User, etc. The central pane shows the results of a query: 'SELECT * FROM `Banking System`.Account;'. The results grid contains columns for username, password, and ssn, with 16 rows of data. A tooltip on the right explains that automatic context help is disabled and suggests using the toolbar for help. The bottom status bar indicates the query was completed successfully.

username	password	ssn
claudia73	jkbusz3	93748236
danielgomez	jhjede	98565824
giovannielin	llynn	746382649
jack12	21kcal	999228693
jacobblack	gggy44	963802783
josephmendez	uhhy	846867482
luischeck23	ujho	983647284
manthat	gndvnh12	83647207
mirandaclaudie	khdi88	983038492
philliplawre	hutyg	846385673
rezwani83	uncs89	883337726
rosejun	kighyr	984637576
scarlett12	konfem	123456789
susan34	khuh89	986382781
sushi1	1fbab	937490889
mariadekko	378uhk	746633778

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Server

Administration Schemas Query 1 User User User User

SCHEMAS

Filter objects Arren Baral Banking System Tables Account Deposit Information Loan Statements Transfer User UserAppliesLoan UserGeneratesStatement UserMakesDeposit UserMakesTransfer UserMakesWithdrawal Withdrawal Views Stored Procedures Functions

Object Info Session

Table: transfer

Columns:

- amount varchar(45)
- name varchar(45) PK
- phone_number varchar(45)

Result Grid Filter Rows: Search Export:

	checking	credit	savings
3300	3773	3886	
3874	833	3574	
2387	675	973	
388	53	978	
863	888	663	
1116	764	732	
873	873	868	
344	353	33	
390	222	444	
555	3333	224	
244	4343	522	
2434	3435	342	
34	873	3568	
455	353	334	
345	353	353	
3535	33	42954	
33	4543	4232	

User 1 Read Only

Action Output

Time	Action	Response	Duration / Fetch Time
267	21:48:53	INSERT INTO User (checking, credit, savings) VALUES ('33', '45...', 1 row(s) affected)	0.001 sec
268	21:48:55	SELECT * FROM `Banking System`.User LIMIT 0, 1000 17 row(s) returned	0.00024 sec / 0.000...

Query Completed

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Server

Administration Schemas Query 1 Statements Statements

SCHEMAS

Filter objects Arren Baral Banking System Tables Account Deposit Information Loan Statements Transfer User UserAppliesLoan UserGeneratesStatement UserMakesDeposit UserMakesTransfer UserMakesWithdrawal Withdrawal Views Stored Procedures Functions

Object Info Session

Schema: Banking System

Table: Statements

Result Grid Filter Rows: Search Export/Import:

date	transactions
2015-03-09	1
2015-04-01	45
2015-04-03	42
2015-04-04	33
2015-04-10	2
2015-04-23	3
2015-04-30	6
2016-05-02	45
2016-05-04	42
2016-05-05	33
2016-05-06	2
2016-05-21	3
2016-05-23	6
2017-05-02	98
2017-05-04	78
2017-05-05	48
2017-05-06	32

Statements 1 Apply Revert

Action Output

Time	Action	Response	Duration / Fetch Time
220	21:36:48	INSERT INTO Statements (date, transactions) VALUES ('2017-0...', 1 row(s) affected)	0.0012 sec
221	21:36:56	SELECT * FROM `Banking System`.Statements LIMIT 0, 1000 19 row(s) returned	0.00027 sec / 0.000...

Query Completed

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Administration Schemas Query 1 Loan Loan

Context Help Snippets

SCHEMAS

Arren Baral Banking System

Tables Account Deposit Information Loan Statements Transfer User UserAppliesLoan UserGeneratesStatement UserMakesDeposit UserMakesTransfer UserMakesWithdrawal Withdrawal Views Stored Procedures Functions

Object Info Session

Table: Loan

Columns:

- loan_type varchar(45)
- APR varchar(45)
- SSN varchar(45) PK

Result Grid

loan_type	APR	SSN
car	1.99	111223333
car	5.99	111663333
car	9.99	222334444
truck	9.99	222355555
home	3.99	222553333
home	3.99	222553333
car	10.99	222855555
car	9.99	555667777
car	9.99	555668888
car	2.99	666005555
home	6.99	666775555
home	6.99	666778888
car	0.99	888553333
home	5.99	888990000
home	3.99	888993333
	HULL	HULL

Loan 1

Action Output

Time	Action	Response	Duration / Fetch Time
184 21:25:28	INSERT INTO Loan (loan_type, APR, SSN) VALUES ('car', '0.99', ...)	1 row(s) affected	0.00088 sec
185 21:25:31	SELECT * FROM `Banking System`.`Loan` LIMIT 0, 1000	16 row(s) returned	0.00024 sec / 0.000...

Query Completed

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Administration Schemas Query 1 Information Information

Context Help Snippets

SCHEMAS

Arren Baral Banking System

Tables Account Deposit Information Loan Statements Transfer User UserAppliesLoan UserGeneratesStatement UserMakesDeposit UserMakesTransfer UserMakesWithdrawal Withdrawal Views Stored Procedures Functions

Object Info Session

Table: Information

Columns:

- first_name varchar(45)
- last_name varchar(45)
- email_address varchar(45)
- phone_number varchar(45)

Result Grid

first_name	last_name	email_address	phone_number
Philip	Swift	philip.swift@gmail.com	1234567890
Wright	Brothers	wright.brothers@gmail.com	1234567890
Mary	Poff	mary.poff@gmail.com	1234567890
Mike	Plant	mike.plant@gmail.com	3334567890
Jack	Mill	jack.mill@gmail.com	1
Milkshake	Wade	milkshake.wade@gmail.com	1234567890
Karen	Custo	karen.custo@gmail.com	1234567890
Jose	Fernandez	jose.fernandez@gmail.com	9984567890
Hamilton	Dine	hamilton.dine@gmail.com	8844567890
Rock	Dayman	rock.dayman@gmail.com	5554567890
John	Den	john.den@gmail.com	1114567890
Rodriguez	Taylor	rodriguez.taylor@gmail.com	8884440000
Billie	Ellish	billie.ellish@gmail.com	4448889999
Khalid	Armani	khalid.armani@gmail.com	6664448888
Normani	Dany	normani.dany@gmail.com	3338885555
Zyan	Nial	zyan.nial@gmail.com	9994447777

Information 1

Action Output

Time	Action	Response	Duration / Fetch Time
134 21:12:08	INSERT INTO Information (first_name, last_name, email_address, ...)	1 row(s) affected	0.00037 sec
135 21:12:11	SELECT * FROM `Banking System`.`Information` LIMIT 0, 1000	16 row(s) returned	0.00025 sec / 0.000...

Query Completed

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Administration Schemas Query 1 Transfer

Context Help Snippets

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Table: Transfer

Columns:

- amount varchar(45)
- name varchar(45) PK
- phone_number varchar(45)

Object Info Session

Result Grid Filter Rows: Search Edit: Export/Import: Result Grid

amount	name	phone_number
800	Bille	2233377260
500	Ellie	4085437536
800	Funet	9997773336
90	Cameron	9473894627
8000	Gal	8844463352
1100	Grate	8884446652
100	Hall	8746352637
900	Hayke	8364527356
500	Jack	8877777260
400	Joss	8833377364
200	Mill	8822277260
600	Nian	7733377260
30	Nor...	8844477260
300	Philip	8833377260
500	Sanoe	9473894627
500	Sul	9994446621
40	Taylor	9364782617

Transfer 1

Action Output

Time	Action	Response	Duration / Fetch Time
244	21:45:05	INSERT INTO Transfer (amount, name, phone_number) VALUES... 1 row(s) affected	0.0022 sec
245	21:45:11	SELECT * FROM `Banking System`.Transfer LIMIT 0, 1000 17 row(s) returned	0.00026 sec / 0.000...

Query Completed

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Administration Schemas Query 1 Deposit Deposit

Context Help Snippets

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Schema: Banking System

Table: Deposit

Columns:

- PIN
- amount

Object Info Session

Result Grid Filter Rows: Search Export: Result Grid

PIN	amount
1111	3000
1234	1900
4321	9048
2422	948
4638	9460
5362	1984
9937	843
3748	444
8934	3339
9469	2453
9647	88
7988	10
9478	3564
5798	4600
7369	999

Deposit 1

Action Output

Time	Action	Response	Duration / Fetch Time
104	20:48:43	INSERT INTO Deposit (PIN, amount) VALUES ('7369', '999') 1 row(s) affected	0.00079 sec
105	20:48:47	SELECT * FROM `Banking System`.Deposit LIMIT 0, 1000 15 row(s) returned	0.0010 sec / 0.00001...

Query Completed

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Administration Schemas

Context Help Snippets

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

SCHEMAS

Tables

- Account
- Deposit
- Information
- Loan
- Statements
- Transfer
- User
- UserAppliesLoan
- UserGeneratesStatement
- UserMakesDeposit
- UserMakesTransfer
- UserWithdrawal
- Views
- Stored Procedures
- Functions
- cs157a
- sakila

Object Info Session

Table: UserMakesDeposit

Columns:

date	varchar(45)
------	-------------

Result Grid

PIN amount

1111	3000
1222	4000
2222	4000
3333	5000
4444	4000
4444	2000
5444	2000
5555	2500
6555	2500
6666	4500
7666	4500
7777	5000
8777	5000
8888	4000
9888	4000
9999	5000
NULL	NULL

Withdrawal 1

Action Output

Time	Action	Response	Duration / Fetch Time
421 22:18:33	INSERT INTO Withdrawal (PIN, amount) VALUES ('9999', '5000')	1 row(s) affected	0.00094 sec
422 22:18:36	SELECT * FROM `Banking System`.'Withdrawal LIMIT 0, 1000	17 row(s) returned	0.00024 sec / 0.000...

Query Completed

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Administration Schemas

Context Help Snippets

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

SCHEMAS

Arren Baral Banking System

Tables

- Account
- Deposit
- Information
- Loan
- Statements
- Transfer
- User
- UserAppliesLoan
- UserGeneratesStatement
- UserMakesDeposit
- UserMakesTransfer
- UserWithdrawal
- Views
- Stored Procedures
- Functions

Object Info Session

Table: Transfer

Columns:

amount	varchar(45)
name	varchar(45) PK
phone_number	varchar(45)

Result Grid

loan_amount payment_term

111100	30
16000	20
30000	10
40000	50
50000	10
80000	5
70000	8
60000	9
988880	40
20000	5
10000	4
30000	10
50000	5
90400	10
3057	6
70000	10
700000	15

UserAppliesLoan 1

Action Output

Time	Action	Response	Duration / Fetch Time
291 21:54:11	INSERT INTO UserAppliesLoan (loan_amount, payment_term) V...	1 row(s) affected	0.00079 sec
292 21:54:14	SELECT * FROM `Banking System`.'UserAppliesLoan LIMIT 0, 1...	21 row(s) returned	0.00094 sec / 0.000...

Query Completed

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Server

Administration Schemas

Query 1 UserGeneratesStatement UserGeneratesStatement

Limit to 1000 rows

1 • SELECT * FROM `Banking System`.UserGeneratesStatement;

Result Grid Filter Rows: Search Export:

date transactions

date	transactions
2015-03-07	1
2015-04-07	1
2015-04-07	9
2015-04-24	8
2015-04-25	5
2015-04-26	7
2015-04-27	9
2015-05-01	10
2015-05-29	11
2015-05-30	12
2015-05-01	13
2015-05-02	14
2015-05-03	15
2015-05-04	16
2015-05-05	17
2015-05-06	18

Object Info Session

Table: UserGeneratesStatement

Columns: date varchar(45) transactions int

Action Output

Time	Action	Response	Duration / Fetch Time
334 22:03:44	INSERT INTO UserGeneratesStatement (date, transactions) VALUES ('2015-05-06', 18)	1 row(s) affected	0.00097 sec
335 22:03:47	SELECT * FROM `Banking System`.UserGeneratesStatement LIMIT 0, 20	16 row(s) returned	0.00024 sec / 0.000...

Query Completed

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Server

Administration Schemas

Query 1 UserMakesDeposit UserMakesDeposit

Limit to 1000 rows

1 • SELECT * FROM `Banking System`.UserMakesDeposit;

Result Grid Filter Rows: Search Export:

date

date
2020-03-09 00:00:00
2020-03-09
2020-03-10
2020-03-11
2020-03-12
2020-03-13
2020-03-15
2020-03-15
2020-04-09
2020-04-09
2020-04-09
2020-04-09
2020-05-10
2020-05-14
2020-05-14
2020-05-15
2020-05-15
2020-05-09
2020-05-09
2020-05-09
2020-05-09

Object Info Session

Table: UserMakesDeposit

Columns: date varchar(45)

Action Output

Time	Action	Response	Duration / Fetch Time
361 22:07:22	INSERT INTO UserMakesDeposit (date) VALUES ('2020-05-09')	1 row(s) affected	0.0019 sec
362 22:07:26	SELECT * FROM `Banking System`.UserMakesDeposit LIMIT 0, 20	20 row(s) returned	0.00024 sec / 0.000...

Query Completed

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Administration Schemas Query 1 UserMakesTransfer UserMakesTransfer

SCHEMAS Filter objects

Tables Account Deposit Information Loan Statements Transfer User UserAppliesLoan UserGeneratesStatement UserMakesDeposit UserMakesTransfer UserMakesWithdrawal Withdrawal Views Stored Procedures Functions cs157a sakila

Object Info Session

Table: UserMakesDeposit

Columns: date varchar(45)

Result Grid Filter Rows: Search Export:

amount	name
220	Philip
20	Jack
320	Milia
30	Selena
400	Jose
50	Miguel
100	Tia
300	Portia
90	Hayle
100	Dale
100	Mil
320	Ella
200	Garnet
300	Juan
50	Sergio
100	Renia
300	Anita
90	Lole
100	Brun

UserMakesTransfer 1 Read Only

Action Output

Time	Action	Response	Duration / Fetch Time
382 22:10:56	INSERT INTO UserMakesTransfer (amount, name) VALUES ('100...')	1 row(s) affected	0.00065 sec
383 22:11:00	SELECT * FROM 'Banking System'.UserMakesTransfer LIMIT 0,...	19 row(s) returned	0.00027 sec / 0.0000...

Query Completed

The screenshot shows the MySQL Workbench interface on a Mac OS X desktop. The main window displays the 'UserMakesTransfer' table with columns 'amount' and 'name'. The table contains 19 rows of data. On the right side of the interface, there is a context help message: 'Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.' Below the main window, the Mac OS X dock is visible with various application icons.

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

Administration Schemas Query 1 UserMakesWithdrawal UserMakesWithdrawal

SCHEMAS Filter objects

Tables Account Deposit Information Loan Statements Transfer User UserAppliesLoan UserGeneratesStatement UserMakesDeposit UserMakesTransfer UserMakesWithdrawal Withdrawal Views Stored Procedures Functions cs157a sakila

Object Info Session

Table: UserMakesDeposit

Columns: date varchar(45)

Result Grid Filter Rows: Search Export:

date	amount
2018-04-09 00:00:00	100
2018-04-10 00:00:00	400
2018-04-11 00:00:00	3000
2018-04-11 00:00:00	500
2018-04-15 00:00:00	1000
2018-04-15 00:00:00	1000
2018-04-20 00:00:00	2200
2018-04-20 00:00:00	2300
2018-04-29 00:00:00	100
2018-04-30 00:00:00	4400
2018-05-05 00:00:00	3000
2018-05-05 00:00:00	500
2018-05-10 00:00:00	1000
2018-05-10 00:00:00	1000
2018-05-10 00:00:00	2200
2018-05-14 00:00:00	2300
2018-05-15 00:00:00	1100

UserMakesWithdrawal 1 Read Only

Action Output

Time	Action	Response	Duration / Fetch Time
402 22:15:44	INSERT INTO UserMakesWithdrawal (date, amount) VALUES ('2018-05-15 00:00:00', 1100)	1 row(s) affected	0.00082 sec
403 22:15:48	SELECT * FROM 'Banking System'.UserMakesWithdrawal LIMIT 0,...	17 row(s) returned	0.00025 sec / 0.0000...

Query Completed

The screenshot shows the MySQL Workbench interface on a Mac OS X desktop. The main window displays the 'UserMakesWithdrawal' table with columns 'date' and 'amount'. The table contains 17 rows of data. On the right side of the interface, there is a context help message: 'Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.' Below the main window, the Mac OS X dock is visible with various application icons.