## Stored Procedures (SQL Task)

## Database HomePro

1. Create and test the Stored procedure to get list of all customers.

Name: **HomePro.GetAllCustomers\_<YourName>**.

1. Create and test the procedure to get list of customers without schedules.

Name: **HomePro.GetCustomersNoSchedule\_<YourName>**

List of columns: **CustomerId,FirstName,LastName,Email,Phone,ZipCode,Age**

1. Create procedure to get list of customers who is older than given age.

Name: **HomePro.GetCustomersBy Age\_<YourName>**

Parameter: @Age int

List of columns: **CustomerId,FirstName,LastName,Email,Phone,ZipCode, Age**

1. Add validation to stored procedure **HomePro.GetCustomersByAge\_<YourName>** that parameter @Age contains Proper value. Call the SP with 3 different parameters:
   1. EXEC HomePro.GetCustomersByAge\_<Name> @Age = 0
   2. EXEC HomePro.GetCustomersByAge\_<Name> @Age = 20
   3. EXEC HomePro.GetCustomersByAge\_<Name> @Age = 200
2. Create procedure to get list of customers without schedules after the given date:

Parameter: @DateNedeed datetime.

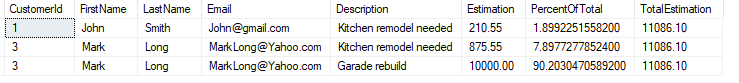
List of columns: **CustomerId,FirstName,LastName,Email,Phone,ZipCode,Age**

The procedure must verify that parameter is in allowed range. Say between ‘2010-01-01’ and ‘2020-01-10’.

Call the SP with 3 different parameters on order the test data validation and compare the result

* @DateNedeed = ‘2000-12-31’
* @DateNedeed = ‘2012-10-21’
* @DateNedeed = ‘2015-10-01’

1. Create procedure to get list of customers with quotes. In addition, calculate the percentage which the current purchase takes in the total sales throughout the whole company.  
   Name: **HomePro.GetEstimationsWithPercentage\_<YourName>**

Expected result.  


1. Create procedure to get list of customers with quotes and discount applied to an eligible age.

Parameters:

@Discount numeric(10,2) - value of discount which may be applied. Value range 0 – 1.

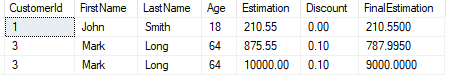
@EligibleAge int - The minimal age of client who is eligible for discount.

Expected result.

EXEC **HomePro.GetEstimationsWithDiscountByAge \_<YourName>**

@Discount = 0.10,

@EligibleAge = 30



Run the SP with following parameters:

* @Discount = -0.5, @EligibleAge = 30
* @Discount = 2, @EligibleAge = 10

1. Create the procedure to get list of customers with quotes with discount applied to purchase amount.

Parameters:

@Discount numeric(10,2) - value of discount which may be applied. Vales range 0 – 1.

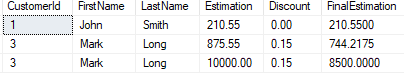
@EligibleAmount numeric(10,2) - The amount of purchase is eligible for discount.

Expected result.

EXEC **HomePro.GetEstimationsWithDiscountByAmount \_<YourName>**

@Discount = 0.15,

@EligibleAmount = 50



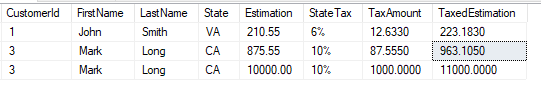
1. Create procedure to get list of customers with quotes and Tax applied According to state rate.

Virginia 6%

California 10%

The rest of States 5%

Expected result.



Assume that your teammate is developer and you are tester. Choose a couple of procedures are whiten by “Developer” and execute them with different set of parameters to validate their behavior.

## Database Bank

1. Create and test the procedure to get list of all clients.

Name: **Bank.GetAllClients\_<YourName>**

List of columns: **ClientId,FirstName,LastName,Phone,Email,State,Age,Type**

1. Create procedure to get list of customers without Account.

Name: **Bank.GetClientsNoAccount\_<YourName>**

List of columns: **ClientId,FirstName,LastName,Phone,Email,State,Age,Type**

1. Create the procedure to get list of customers who is in given range of age.

Name: **Bank.GetClientsBy Age\_<YourName>.**

Parameters: @AgeBegin int, @AgeEnd int. The SP must verify that values of parameters “Age” are in the allowed range, which is between 18 and 100 and that @AgeBegin > @AgeEnd

Call the SP with 3 different parameters:

* 1. EXEC HomePro.GetCustomersByAge\_<Name> @AgeBegin = 0, @AgeEnd = 50
  2. EXEC HomePro.GetCustomersByAge\_<Name> @AgeBegin = 20, @AgeEnd = 50
  3. EXEC HomePro.GetCustomersByAge\_<Name> @AgeBegin = 50, @AgeEnd = 40

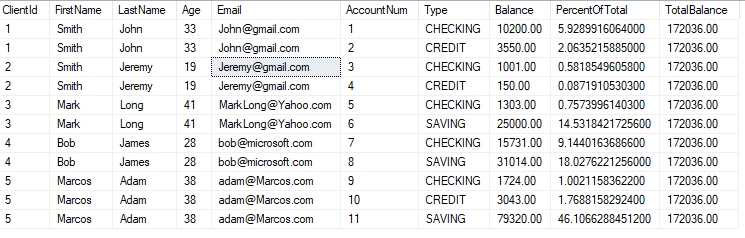
1. Create procedure to get list of clients without given account type:

Parameter: @AccountType CHAR(10). Account Type must be one of: “CHECKING”, “CREDIT”, “SAVING”. The procedure must verify the parameter value.

List of columns: **ClientId,FirstName,LastName,Phone,Email,State,Age,Type**

1. Create the procedure to get list of Clients with Account’s Balances and calculate the percentage which each balance takes in the total balance throughout the bank.  
   Name: **Bank.GetAccountsWithBalances\_<YourName>**

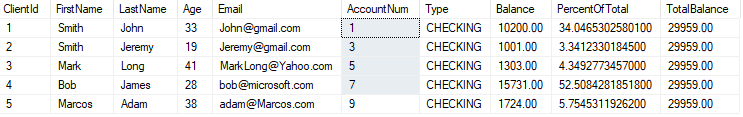
Expected result.



1. Create the procedure to get list of Clients with Account’s Balances and calculate the percentage of the account’s balance in the total Balance throughout the bank for the given AccountType as a parameter  
   Name: **Bank.GetAccountsWithBalancesByType\_<YourName>**

Expected result.

**EXEC Bank.GetAccountsWithBalancesByType\_<YourName>** @AccountType = 'CHECKING';



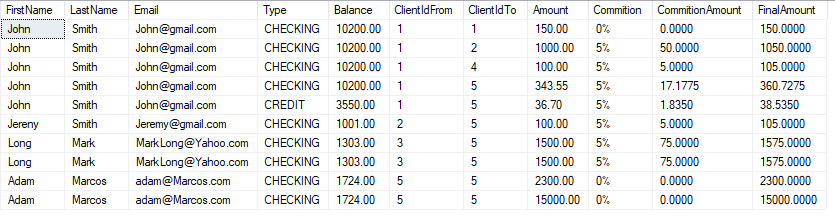
**Bank.GetAccountsWithBalancesByType\_<YourName>** @AccountType = 'SAVING';

**Bank.GetAccountsWithBalancesByType\_<YourName>** @AccountType = 'CREDIT';  
**Bank.GetAccountsWithBalancesByType\_<YourName>** @AccountType = 'BLABLABLA';

1. Create the procedure to get list of Clients with transactions they paid and calculate the bank’s commission on each transaction. The commissions rules are. If client moved money to his own account then commission 0% otherwise 5%

Name: **Bank.GetTransactionsWithCommision\_<YourName>**

Expected result.



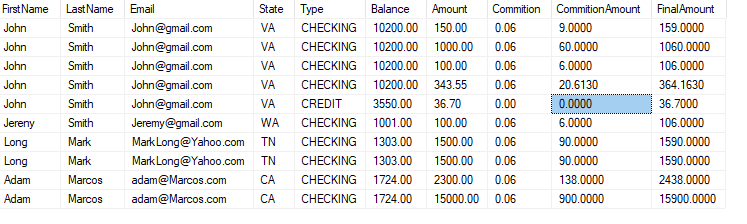
1. Create the procedure to get list of Clients with transactions they paid and calculate the bank’s commission on each transaction. The commission’s rules are. The SP has 2 parameters:

@AccountType char(10) – type of account which is eligible for bank commission @ Commission numeric(10,2) – Commission size. Value in range from 0 to 1.

Expected result.

**EXEC Bank.GetAccountsWithCommitionByAccoutnType\_<YourName>**

@AccountType = 'CHECKING', @Commission = 0.06



Run the SP with different parameters:

**EXEC Bank.GetAccountsWithCommitionByAccoutnType\_<YourName>**

@AccountType = 'CREDIT', @Commission = 0.09

Send your homework to my email [Andrew.a4100@gmail.com](mailto:Andrew.a4100@gmail.com) for validation