has >

|  |
| --- |
| AccountOperations |

|  |
| --- |
| Customer |

|  |
| --- |
| SavingsAccount |

|  |
| --- |
| EmployeeOperations |

|  |
| --- |
| Account |

has >

|  |
| --- |
| Employee |

has >

|  |
| --- |
| FixedAccount |

|  |
| --- |
| ITransactions |

|  |
| --- |
| CustomerOperations |

|  |
| --- |
| Bank |

|  |
| --- |
| ITransactions |
| void deposit(double amount)  void withdraw(double amount)  void transfer(Account a, double amount) |

|  |
| --- |
| AccountOperations |
| void insertAccount(Account a)  void removeAccount(Account a)  Account getAccount(int accountNumber)  void showAllAccounts( ) |

|  |
| --- |
| CustomerOperations |
| void insertCustomer(Customer c)  void removeCustomer(Customer c)  Customer getCustomer(int nid)  void showAllCustomers( ) |

|  |
| --- |
| EmployeeOperations |
| void insertEmployee(Employee e)  void removeEmployee(Employee e)  Employee getEmployee(int empId)  void showAllEmployees( ) |

|  |
| --- |
| Customer |
| String name  int nid  Account accounts[ ] |
| void setName(String name)  void setNid(int nid)  String getName( )  int getNid( ) |

|  |
| --- |
| Employee |
| String name  String empId  double salary |
| void setName(String name)  void setNid(int nid)  void setSalary(double salary)  String getName( )  int getNid( )  double getSalary( ) |

|  |
| --- |
| Account |
| int accountNumber  double balance |
| void setAccountNumber(int accountNumber)  void setBalance(double balance)  int getAccountNumber( )  double getBalance( )  abstract void showInfo( ) |

|  |
| --- |
| SavingsAccount |
| double interestRate |
| void setInterestRate(double interestRate)  double getInterestRate( ) |

|  |
| --- |
| FixedAccount |
| int tenureYear |
| void setTenureYear(int tenureYear)  int getTenureYear( ) |

|  |
| --- |
| Bank |
| Customer customers[ ]  Employee employees[ ] |
|  |

|  |
| --- |
| **Start** |
| The ***Start*** class contains the main method. Inside the main method the application prompts the user to choose among the following options:   1. Employee Management 2. Customer Management 3. Customer Account Management 4. Account Transactions 5. Exit   Upon choosing any of the option the user is given some further options to choose from:  *Options for Employee Management:*   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | Insert New Employee |  | Remove Existing Employee |  | Show All Employees |   *Options for Customer Management:*   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | Insert New Customer |  | Remove Existing Customer |  | Show All Customer |   *Options for Customer Account Management:*   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | Insert New Account |  | Remove Existing Account |  | Show All Accounts |   Options for Employee Management:   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  | Deposit Money |  | Withdraw Money |  | Transfer Money |   The application will prompt the user to give input for the necessary values as per the choice. The application will repeatedly go on according to the user choices. |

Requirements:• Object Oriented Programming Principles must be followed.  
• The whole program must be organized using at least three user defined packages.  
• Whenever a Transaction is made, the details of the transaction has to be written in a text file.  
• The program must be able to handle exceptions by itself.

Project Submission Instruction:  
 • You will have to upload the project in your VUES account in a link named “Project”.

* Any one member will upload the project on behalf of the group.
* You have to upload zip files. Put all the .java files into a zip and rename the zip folder by your id then upload it to the portal.

Project Submission Deadline:

19.04.2020