Class Diagram (UML)

Relationships:
- Customer ----▷ AccountOperations
- Bank ----▷ EmployeeOperations
- Bank ----▷ CustomerOperations
- Bank has > Customer
- Bank has > Employee
- Bank has > Account (has >)
- SavingsAccount ——▷ Account
- FixedAccount ——▷ Account
- Account ----▷ ITransactions

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( )
```

Account
```
int accountNumber
double balance

void setAccountNumber(int accountNumber)
void setBalance(double balance)
int getAccountNumber( )
double getBalance( )
abstract void showInfo( )
```

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( )
```

FixedAccount
```
int tenureYear

void setTenureYear(int tenureYear)
int getTenureYear( )
```

SavingsAccount
```
double interestRate

void setInterestRate(double interestRate)
double getInterestRate( )
```

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:

    1. Insert New Employee   2. Remove Existing Employee   3. Show All Employees   4. Go Back

Options for Customer Management:

    1. Insert New Customer   2. Remove Existing Customer   3. Show All Customers   4. Go Back

Options for Customer Account Management:

    1. Insert New Account   2. Remove Existing Account   3. Show All Accounts   4. Go Back

Options for Account Transactions:

    1. Deposit Money   2. Withdraw Money   3. Transfer Money   4. Show Transactions   5. Go Back

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 two 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.