**Project Documentation**

1. List of Classes.
   1. AccountAbstractFactory - An Abstract class containing all the required methods to generate the required classes for the output processor and Account classes.
      1. getData() – abstract method to get the data object with datavariables
      2. getStoreData() – abstract method to create and return storeData class
      3. getIncorrectIdMsg() - abstract method to create and return IncorrectIdMsg class
      4. getIncorrectPinMsg() - abstract method to create and return IncorrectPinMsg class
      5. getTooManyAttemptsMsg() - abstract method to create and return TooManyAttemptsMsg class
      6. getDisplayMenu() - abstract method to create and return DisplayMenu class
      7. getMakeDeposit() - abstract method to create and return MakeDeposit class
      8. getDisplayBalance() - abstract method to create and return DisplayBalance class
      9. getPromptForPin() - abstract method to create and return PromptForPin class
      10. getMakeWithdraw() - abstract method to create and return MakeWithdraw class
      11. getPenality() - abstract method to create and return Penalty class
      12. getIncorrectLockMsg() - abstract method to create and return IncorrectLockMsg class
      13. getIncorrectUnlockMsg() - abstract method to create and return IncorrectUnlockMsg class
      14. getNoFundsMsg() - abstract method to create and return NoFundsMsg class
   2. Account1ConcreteFactory – A concrete class that implements the methods of Account abstract factory and this factory is used by Account-1
      1. getData() – method to get the data object with data variables for Data1
      2. getStoreData() – method to create and return storeData1 class
      3. getIncorrectIdMsg() - method to create and return IncorrectIdMsg1 class
      4. getIncorrectPinMsg() - method to create and return IncorrectPinMsg1 class
      5. getTooManyAttemptsMsg() - method to create and return TooManyAttemptsMsg1 class
      6. getDisplayMenu() - method to create and return DisplayMenu1 class
      7. getMakeDeposit() - method to create and return MakeDeposit1 class
      8. getDisplayBalance() - method to create and return DisplayBalance1 class
      9. getPromptForPin() - method to create and return PromptForPin1 class
      10. getMakeWithdraw() - method to create and return MakeWithdraw1 class
      11. getPenality() - method to create and return Penalty1 class
      12. getIncorrectLockMsg() - method to create and return IncorrectLockMsg1 class
      13. getIncorrectUnlockMsg() - method to create and return IncorrectUnlockMsg1 class
      14. getNoFundsMsg() - method to create and return NoFundsMsg1 class
   3. Account2ConcreteFactory - A concrete class that implements the methods of Account abstract factory and this factory is used by Account-2
      1. getData() – method to get the data object with data variables for Data2
      2. getStoreData() – method to create and return storeData2 class
      3. getIncorrectIdMsg() - method to create and return IncorrectIdMsg2 class
      4. getIncorrectPinMsg() - method to create and return IncorrectPinMsg2 class
      5. getTooManyAttemptsMsg() - method to create and return TooManyAttemptsMsg2 class
      6. getDisplayMenu() - method to create and return DisplayMenu2 class
      7. getMakeDeposit() - method to create and return MakeDeposit2 class
      8. getDisplayBalance() - method to create and return DisplayBalance2 class
      9. getPromptForPin() - method to create and return PromptForPin2 class
      10. getMakeWithdraw() - method to create and return MakeWithdraw2 class
      11. getPenality() - method to create and return Penalty2 class
      12. getIncorrectLockMsg() - method to create and return IncorrectLockMsg2 class
      13. getIncorrectUnlockMsg() - method to create and return IncorrectUnlockMsg2 class
      14. getNoFundsMsg() - method to create and return NoFundsMsg2 class
   4. Account – A parent class that is defined to create template for Accounts
      1. Constructor() - Constructor to intialize and set the value for the abstractfactory
   5. Account1 – A child class that implements Account class and defines all the methods required for account1
      1. Constructor() - Constructor to intialize and set the value for the abstractfactory
      2. Open() - methods store the input data for open in data class
      3. Login() - method validates the userId and performs login
      4. Logout() - method logs out the user
      5. Pin() - method validates the pin and sets the context according to the balance
      6. Deposit() - method deposits the money and updates the context based on balance
      7. Withdraw() - method withdraws the money and updates the context based on balance
      8. Balance() - method displays the balance
      9. Lock()- method validates the pin and updates the context
      10. Unlock() - method validates the pin and updates the context
   6. Account2 – A child class that implements Account class and defines all the methods required for account2
      1. Constructor() - Constructor to intialize and set the value for the abstractfactory
      2. OPEN() - methods store the input data for open in data class
      3. LOGIN() - method validates the userId and performs login
      4. LOGOUT() - method logs out the user
      5. PIN() - method validates the pin and sets the context according to the balance
      6. DEPOSIT() - method deposits the money and updates the context based on balance
      7. WITHDRAW() - method withdraws the money and updates the context based on balance
      8. BALANCE() - method displays the balance
      9. Susptend()- method suspends the account by updating the context
      10. Activate() - method activate the account by updating the context
      11. Close() - method closes the account by updating the context
   7. DisplayBalance – A parent class defined to create a template to display balance
      1. displayBalance() – A template method defined to display balance
   8. DisplayBalance1 – A child class that implements DisplayBalance class and is used by Account 1 to display balance.
      1. displayBalance() – A method defined to display balance in Account1
   9. DisplayBalance2 – A child class that implements DisplayBalance class and is used by Account 2 to display balance.
      1. displayBalance() – A method defined to display balance in Account2
   10. DisplayMenu – A parent class defined to display all the available transaction options
       1. displayMenu() – A template method to display available transactions
   11. DisplayMenu1 – A child class defined to display all the available transaction options for Account1
       1. displayMenu() – A method to display available transactions in Account1
   12. DisplayMenu2 – A child class defined to display all the available transaction options for Account2
       1. displayMenu() – A method to display available transactions in Account2
   13. IncorrectIdMsg – A parent class defined to print the incorrect id msg when incorrect id was entered during the login.
       1. incorrectIdMsg() – A template defined to print the incorrect id msg when incorrect id was entered during the login.
   14. IncorrectIdMsg1 – A child class defined to print the incorrect id msg when incorrect id was entered during the login of Account1
       1. incorrectIdMsg() – A method defined to print the incorrect id msg when incorrect id was entered during the login of Account1
   15. IncorrectIdMsg2 – A child class defined to print the incorrect id msg when incorrect id was entered during the login of Account2
       1. incorrectIdMsg() – A method defined to print the incorrect id msg when incorrect id was entered during the login of Account2
   16. IncorrectLockMsg – A parent class defined to Print Incorrect lock message.
       1. incorrectLockMsg() - A template defined to Print Incorrect lock message.
   17. IncorrectLockMsg1 – A child class defined to Print Incorrect lock message for account 1
       1. incorrectLockMsg() - A method defined to Print Incorrect lock message for Account 1
   18. IncorrectPinMsg – A parent class defined to Print Incorrect Pin message when incorrect pin is entered
       1. incorrectPinMsg() - A template defined to Print Incorrect Pin message when incorrect pin is entered
   19. IncorrectPinMsg1 – A child class defined to Print Incorrect Pin message when incorrect pin is entered in Account 1
       1. incorrectPinMsg() - A method defined to Print Incorrect Pin message when incorrect pin is entered for account1
   20. IncorrectPinMsg – A child class defined to Print Incorrect Pin message when incorrect pin is entered in Account 2
       1. incorrectPinMsg() - A method defined to Print Incorrect Pin message when incorrect pin is entered for account2
   21. IncorrectUnlockMsg – A parent class defined to Print Incorrect Unlock msg.
   22. IncorrectUnlockMsg1 – A child class defined to Print Incorrect Unlock msg for Account1
   23. MakeDeposit – A parent class defined to perform the steps required to deposit money.
   24. MakeDeposit1 – A child class defined to perform the steps required to deposit money in Account 1.
   25. MakeDeposit2 – A child class defined to perform the steps required to deposit money in Account 2.
   26. MakeWithdraw– A parent class defined to perform the steps required to withdraw money from Account 1.
   27. MakeWithdraw1– A child class defined to perform the steps required to withdraw money from Account 1.
   28. MakeWithdraw2 – A child class defined to perform the steps required to withdraw money from Account 2.
   29. NoFundsMsg – A parent class defined to display msg when trying to withdraw with low/no balance
   30. NoFundsMsg1 – A child class defined to display Below minimum Balance when trying to withdraw with low balance in Account 1.
   31. NoFundsMsg2 – A child class defined to display No funds when trying to withdraw with no balance in Account 2.
   32. Penalty – A parent class defined to implement the penalty i.e deduct money from accounts
   33. Penalty1 – A child class defined to implement the penalty i.e deduct money from account1
   34. PromptForPin – A parent class defined to print message to prompt pin from user.
   35. PromptForPin1 – A child class defined to print message to prompt pin from Acccount1 user.
   36. PromptForPin2 – A child class defined to print message to prompt pin from Account2 user.
   37. StoreData – A parent class defined to store initial login credentials entered by user.
   38. StoreData1 – A child class defined to store initial login credentials entered by Account 1 user.
   39. StoreData2 – A child class defined to store initial login credentials entered by Account2 user.
   40. TooManyAttemptsMsg – A parent class defined to display message when Login attempts are exceeded.
   41. TooManyAttemptsMsg1 – A child class defined to display message when Login attempts are exceeded for Account1.
   42. TooManyAttemptsMsg2 – A parent class defined to display message when Login attempts are exceeded for Account2.
   43. Data – A parent class defined to store the temporary and permanent data.
       1. No methods defined for this class
   44. Data1 - A child class defined to store the temporary and permanent data for account1
       1. Getters and setters for all the variables are defined
   45. Data2 - A child class defined to store the temporary and permanent data for account2
       1. Getters and setters for all the variables are defined
   46. State – A parent class defined to create a template for all the states required for MDA.
   47. Start – A child class of State class defined to initialize the transactions for Accounts.
   48. Idle – A child class of State class defined to process login for Accounts.
   49. CheckPin – A child class of State class defined to validate login for Accounts.
   50. Locked – A child class of State class defined to Lock Accounts.
   51. Overdraw – A child class of State class defined to perform the transactions for Accounts that are overdrawn
   52. Ready – A child class of State class defined to perform the transactions for Accounts.
   53. StateS1 – A child class of State class defined to help transition of states between ready and Ready and lock and unlock for Accounts.
   54. Stop – A child class of State class defined to stop the transactions for Accounts.
   55. Suspend – A child class of State class defined to suspend accounts.
   56. Driver – A main class defined to invoke various transactions on the Accounts
       1. Main() - main method invokes the first step of execution i.e selection of account
       2. accountSelection() - Takes input from user and executes the operations specific to the account
       3. recursiveAccount1() - a recursive method for account1 defined to recursively execute until user explicitly requests for exit
       4. recursiveAccount2() - a recursive method for account2 defined to recursively execute until user explicitly requests for exit
       5. account1Operations() - a utlitly method to print operations for account1
       6. account2Operations() - a utlitly method to print operations for account2
   57. MDAContex – A context class defined to facilitate state transitions.
       1. open() - execute the open method on current state and update the state to idle for valid executions
       2. login() - execute the login method on current state and update the state to checkpin for valid executions
       3. incorrectLogin() - execute the incorrectLogin method on current state
       4. incorrectpin() - execute the incorrect method on based on number of attemts and update the state for valid executions
       5. correctPinBelowMin() - method to change state based on balance
       6. correctPinAboveMin() - method to change state based on balance
       7. deposit() - method to change state based on balance and deposit money
       8. belowMinBalance() - method to change state based on balance
       9. aboveMinBalance() - method to change state based on balance
       10. logout() - method to logout based on current state
       11. getStateList() - method to get a list of all the classes available
       12. close() - method to close the account
       13. activate() - method to activate the account
       14. suspend() - method to suspend the account
       15. incorrectUnlock() - method to change state based on balance
       16. unlock() - method to change state based on balance
       17. incorrectLock() - method to change state based on balance
       18. lock() - method to change state based on balance
       19. withdrawBelowMinBalance() - method to change state based on balance
       20. balance() - method to change fetch balance
       21. withdraw() - method to change state based on balance
   58. OutputProcessor – A class defined to execute various Output processes.
       1. storeData() - Invoke store data to update the dataclass
       2. incorrectIdMsg() - Invoke to display relavent msg
       3. incorrectPinMsg() - Invoke to display relavent msg
       4. tooManyAttemptsMsg() - Invoke to display relavent msg
       5. displayMenu() - Invoke to display menu
       6. makeDeposit() - Invoke to make deposit
       7. displayBalance() - Invoke to display relavent msg with balance
       8. promptForPin() - Invoke to display relavent msg
       9. makeWithdraw() - Invoke to withdraw money
       10. penalty() - Invoke to apply penalty
       11. incorrectLockMsg() - Invoke to display relavent msg
       12. incorrectLockMsg() - Invoke to display relavent msg
       13. noFundsMsg() - Invoke to display relavent msg