|  |  |  |  |
| --- | --- | --- | --- |
| Up-gradation of Account class | | | |
| Account | accId, accName, accBalance  use all the previous methods and constructors |  | |
| Extended part  (We want to preserve history of transactions for each account object. To do so we need to develop one more class name Transaction) | | | |
| Transaction | Account sender  Account receiver  int amount  String additionalInfo  1 constructor// Parameterized  Transaction(Account sender, Account receiver, int amount)  showInfo() |  | |
| In the Account class use one/two more member variable to store transaction information and necessary member functions. | | | |
|  | Transaction [] listOfTransaction // M.V  inttotalNumberOfTransaction // M.V  void showAllTranscation() |  | |
|  | | |
|  | | |
|  | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| Book | String bookName  String bookAuthor  String bookId  String bookType  int bookCopy // how many copy | Member variables  declare private |
| 2 Constructor (Empty, valued)  void showInfo()  void addBookCopy(int x)// how many copy of book |  |
|  | To count the total number of book object use static modifier to count book object.  If static is covered in your theory class only then you precede this. | |
|  | static int bookCounter  static void showTotalBookInfi() | |
|  | | |
|  |  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| Library  Reuse Book Class from lab\_02 | String libName  String libAddress  Book [] listOfBook  int totalBook; | Member variables  declare private |
| 2 constructors (empty and valued)  void showLibInfo() // show library info and all book info as well  void addNewBook(Book book) // add a new book into lib  void deleteBook(Book book) // delete book object  void addNewBookCopy(Book book, int copy) |  |