Deadline: The time duration for the quiz is 80 minutes. However You have 24 hours to submit your quiz. Late submission will not be accepted on goggle classroom.

Name: **ID:**

Object Oriented Programming (AM/BM)
QUIZ 02

Honor Code Pledge: 'On my honor I have neither given nor received help on this quiz.'

Question 1:

A bank maintains two kinds of accounts – Savings account and Current account. The savings account provides simple interest, deposit and withdrawal facilities. The current account only provides deposit and withdrawal facilities. Current account holders should also maintain minimum balance. If balance falls below minimum level a service charge is imposed. Create an abstract class Account that stores customer name, account number type of account and abstract methods. From this derive the classes Curr_Account (double balance, double min_bal, double serviceCharge / penalty) and and Sav_Account (double balance). Include the necessary methods in order to achieve the following:

1. Define parameterized constructor in a class hierarchy.
2. Allow deposit and update the balance.
3. Display the balance.
4. Compute interest and add to balance.
5. Permit withdrawal and update the balance (check for minimum balance).
6. Apply polymorphism if required for methods in class hierarchy.
7. Create an array of super class / object and populate with subclass objects and call the overridden / object methods.
8. Write a test program to demonstrate the above said implementations.

Question 2:

Define a interface EMPInterface (void displayEMP(), void giveBonus (double amount)). Define an abstract class Employee(empID, fName, lName, salary). Define a concrete class Manager (noOfOtockOptions), subclass of Employee and define interface methods.

Perform the following

1. Define appropriate constructors in a class hierarchy.

Deadline: The time duration for the quiz is 80 minutes. However You have 24 hours to submit your quiz. Late submission will not be accepted on goggle classroom.

2. Ensure the bonus amount should not be negative and zero using exception handling mechanism (use throws and throw clauses of exception handling).
3. Create array of interface reference variables and populate with manager objects.
4. Write a test program to implement the above said requirements of interface implementation and exception handling.