*QUEST NUMBER 1*

***INSTRUCTIONS:***

1. Error based question must be answered in a word file and submitted on drive or github.
2. Scenario based questions must be coded and whole java project folder must be submitted.

***ERROR BASED QUESTIONS:***

Number 1: Find at least two errors in the following program

**package** test;

**import** java.util.Scanner;

**class** First

{

Scanner inp = **new** Scanner(System.***in***);

**public** **int** setValue =0;

**private** **int** value = 0;

First()

{

setValue = inp.nextFloat();

value = setValue;

}

**public** **void** display()

{

System.***out***.println("Variable initialized is: ",setValue);

}

}

**public** **class** Main

{

**public** **static** **void** main(String[] args)

{

First f1 = **new** First();

f1.value = f1.value\*f1.value;

f1.display();

}

}

Number 2: Find all errors in following programs.

Hint: Look about super() in java before attempting.

**class** Parent {

**int** x;

Parent(**int** x) {

**this**.x = x;

}

**void** display() {

System.***out***.println("Value of x: " + x);

}

}

**class** Child **extends** Parent

{

**int** y;

Child(**int** x, **int** y)

{

**this**.x = x;

**this**.y = y;

}

**void** display()

{

System.***out***.println("Value of x: " + x + ", Value of y: " + y);

}

}

**class** GrandChild1 **extends** Child {

**int** z;

GrandChild1(**int** x, **int** y, **int** z) {

**this**.x = x;

**this**.y = y;

**this**.z = z;

}

**void** display() {

System.***out***.println("Value of x: " + x + ", Value of y: " + y + ", Value of z: " + z);

}

}

**class** GrandChild2 **extends** Child,GrandChild1

{

GrandChild2(**int** x, **int** y)

{

**this**.x = x;

**this**.y = y;

}

**void** display() {

System.***out***.println("Value of x: " + x + ", Value of y: " + y);

}

}

**public** **class** Main {

**public** **static** **void** main(String[] args) {

Child child = **new** Child(10, 20);

child.display();

GrandChild1 gc1 = **new** GrandChild1(10, 20, 30);

gc1.display();

GrandChild2 gc2 = **new** GrandChild2(10, 20);

gc2.display();

}

}

***SCNEARIO BASED QUESTIONS:***

**Question 1: Banking System**

**Scenario:**

You're designing a banking system. Utilize interfaces and abstract classes to represent various account types such as savings account, checking account, and investment account, ensuring that common functionalities can be implemented while allowing for specific implementations for each account type?

**Common Functionalities:**

- Tax

- Transactions

- Pin Number

- etc

**Question 2: School Management System**

**Scenario:**

Develop a software application for a school management system. Design the classes representing different entities such as students, teachers, and courses. Common methods or functionalitis must be inherited while allowing for specific implementations for each entity type.

**Hints**

- Department

- Names

- Timings

- Assigning user ids.

.