Question 1

Correct

Mark 1.00 out of 1.00

Flag question

If the access specifier of the display method in super class is protected, then what could be the valid specifier for the overriding display method in sub class?

Select one:

- protected and default
- private and protected
- protected, default and public
- protected and public ✔

Question 2

Correct

Mark 1.00 out of 1.00

Flag question

_____ is the best example for Runtime Polymorphism.

Select one:

- Method Overriding ✔
- Method Overloading
- Both the options

Question 3

Correct

Mark 1.00 out of 1.00

Flag question

Observe the code.

```
interface A1 {
      public abstract void  a11();
      public abstract void  a12();
}
interface A2 extends A1 {
       void  a21();
       void  a22();
}
public class C  implements A2  {
}
```

In the above scenario, which methods should class C override?

Select one:

- void a21() { }
  void a22(){ }
- public void a11() { }
  public void a12() { }
  void a21() { }
  void a22(){ }
- public void a11() { }
  public void a12() { }
  public void a21() { }
  public void a22(){ } ✔
- public void a21() { }
  public void a22(){ }

Question 4

Correct

Mark 1.00 out of 1.00

Flag question

Observe the statements.

Statement 1 : An abstract class cannot have non abstract methods

Statement 2 : An abstract class should have a minimum of one abstract method in its class.

Select one:

- Both Statement 1 and Statement 2 are true
- Statement 2 alone is true
- Statement 1 alone is true
- Statement 1 and Statement 2 are false ✔

Question 5

Correct

Mark 1.00 out of 1.00

Flag question

If the sub class overrides the method in the parent class, the overridden method can be invoked by using either 'this' or 'super' keyword. State true or false.

Select one:

- True
- False ✔

Question 6

Correct

Mark 1.00 out of 1.00

Flag question

Predict the output.

```
class Parent {
   public void display()   {
      System.out.println("In Parent");
   }
}
class Test extends Parent {
   public void display() {
      System.out.println("In child");
   }
   public void testMethod() {
      System.out.println("In test method");
   }
   public static void main(String[] args) {
      Parent p = new Test();
      p.display();
      p.testMethod();
   }
}
```

Select one:

- In Parent
  In child
  In test method
- Compile Time error ✔
- In child
  In test method
- Runtime error

Question 7

Correct

Mark 1.00 out of 1.00

Flag question

Predict the output.

```
abstract class Sample {
   public int x;
   Sample()     {
      x = 10;
   }
   abstract final public void display();
}

class Test extends Sample {
   final public void display()   {
      System.out.println("x = " + x);
   }
   public static void main(String[] args)   {
      Test t = new Test();
      t.display();
   }
}
```

Select one:

- Runtime error
- x = 10
- x = 0
- Compile Time error ✔