Question 1

Correct

Mark 1.00 out of 1.00

⚑ Flag question

```
public class ArithmeticOperation{
  private void add(int operand1, int operand2)
  {
    System.out.println(operand1 + operand2);
  }
}
public class Addition extends ArithmeticOperation
{
  public void show()
  {
    add(10, 12);
  }
  public static void main(String args[])
  {
    Addition ob = new Addition();
    ob.show();
  }
}
```

What will be the output of above code when compiled and executed?

Select one:

○ Runtime error as add method is not defined in MethodOverriding class

○ Will compile and display 1012

○ Will print false

○ Will compile and display 32

◉ Compile time error ✔

Question 2

Correct

Mark 1.00 out of 1.00

⚑ Flag question

What will be the output of the following program ?

```
class FourWheeler
{
  public FourWheeler()
  {
    System.out.println("Class FourWheeler");
  }
}
class Car extends FourWheeler
{
  public Car()
  {
    System.out.println("Class Car");
  }
}
class Audi extends Car
{
  public Audi()
  {
    super();
    System.out.println("Class Audi");
  }
}
class Driver
{
  public static void main(String args[])
  {
    Audi cc=new Audi();
  }
}
```

Select one:

○ Class Audi
  Class Car
  Class FourWheeler

○ Compile Time Error

○ Exception occurs

◉ Class FourWheeler
  Class Car
  Class Audi ✔

Question 3

Correct

Mark 1.00 out of 1.00

⚑ Flag question

10. class FourWheeler {

11. [private] ✔ void display() { }

12. }

13. class Car extends FourWheeler {

14. [protected] ✔ void display() { }

15. }

Which method at line 14, will correctly complete class Car?

Question 4

Correct

Mark 1.00 out of 1.00

⚑ Flag question

Given:
class FourWheeler
{
public FourWheeler ()
{
System.out.print(1);
}
}
class Car extends FourWheeler
{
public Car()
{
System.out.print(2);
}
}
class Audi extends Car
{
public Audi()
{
System.out.print(3);
}
}
public class Driver
{
public static void main(String[] argv)
{
new Audi();
}
}
What is the result when this code is executed?

Select one:

○ The code runs with no output

○ 321

○ 3

◉ 123 ✔

Question 5

Correct

Mark 1.00 out of 1.00

⚑ Flag question

The class Employee is declared in a package **mainpack** and the Derived classes PermanentEmployee and TemporaryEmployee are declared in another package **subordpack**. The basicPay attribute should be accessed only by means of a derived class object.

How to ensure that the basicPay attribute is not accessed directly by the other classes in the subordpack?

Employee.java

package [mainpack ⇕] ✔ ;

public class Employee{

[protected ⇕] ✔ **int basicPay;**

}

/*PermanentEmployee.java*/

package [subordpack ⇕] ✔ ;

public class PermanentEmployee extends Employee{

.....

}

//TemporaryEmployee.java

package [subordpack ⇕] ✔ ;

public class TemporaryEmployee extends Employee{

.....

}

Question 6

```
class FourWheeler{
protected FourWheeler getObject()
{
   //logic with return statement
}
}
class Car extends FourWheeler
{
        protected Car getObject() {}   ✔
}
class Driver
{
public static void main(String args[])
{
   FourWheeler object = new Car();
   object.getObject();
}
}
```

```
private FourWheeler getObject() {}                    Car getObject() {}
```

```
protected void getObject() {}
```

Question 7

Say that there are three classes: Computer, AppleComputer, and IBMComputer. What are the likely relationships between these classes

Select one:

○ Computer, AppleComputer and IBMComputer are sibling classes.

○ Computer is a superclass, AppleComputer is a subclasses of Computer, and IBMComputer is a sublclas of AppleComputer

○ IBMComputer is the superclass, AppleComputer and Computer are subclasses of IBMComputer.

◉ Computer is the super class, AppleComputer and IBMComputer are subclasses of computer ✔

Question 8

Constructor of the superclass [can ⇅] ✔ be invoked by from its subclass.

A default member of a super class in one package [cannot ⇅] ✔ be accessed in its own subclass which is in a different package.

A protected member of a super class in one package [can ⇅] ✔ be accessed by its own subclass which is in a different package.

Question 9

State True or False
Child class objects can be instantiated when the parent class constructor is protected

Select one:
◉ True ✔
○ False

Question 10

[Default Access] ✔ is the most restrictive access modifier that will allow members of one class to have access to members of another class in the same package.

[Public]   [Synchronized]   [Protected]   [Abstract]

Given a method in a public class, [private ⇅] ✔ access modifier must be used to restrict access to that method to only the other members of the same class.

Interpret which of the following statements are correct with respect to inheritance relationship in java?

Select one or more:

☐ object of subclass referenced by super class type can access newly defined sub class variables

☑ object of subclass referenced by super class type can invoke overridden sub class methods ✔

☑ object of subclass referenced by super class type can invoke super class methods ✔

☐ object of subclass referenced by super class type can invoke newly defined sub class methods

☑ object of subclass referenced by super class type can access super class variables ✔

Given a method in a public class, [private ⇅] ✔ access modifier must be used to restrict access to that method to only the other members of the same class.