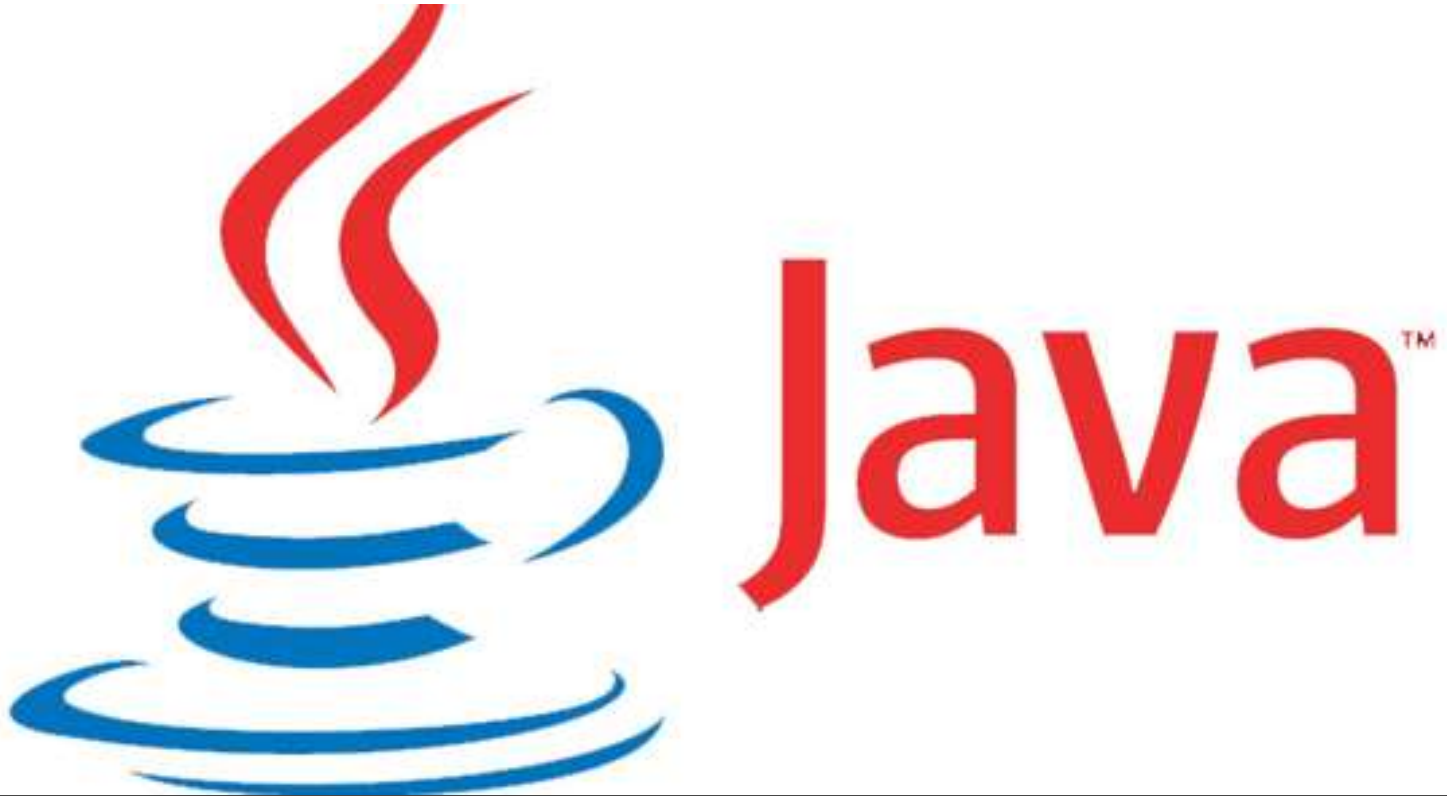


Session2- **Java7**

Exercises

Session2: Exercises

Multiple Choice Question

Inheritance And Polymorphism

Q46. What type of inheritance does Java have?

A. single inheritance

B. double inheritance

C. multiple inheritance

D. class inheritance

Inheritance And Polymorphism

Q47. What restriction is there on using the *super* reference in a constructor?

A. It can only be used in the parent's constructor.

B. Only one child class can use it.

C. It must be used in the last statement of the constructor.

D. It must be used in the first statement of the constructor.

Inheritance And Polymorphism

Q48. A class Animal has a subclass Mammal. Which of the following is true?

A. Because of single inheritance, Mammal can have no subclasses.

B. Because of single inheritance, Mammal can have no other parent than Animal.

C. Because of single inheritance, Animal can have only one subclass.

D. Because of single inheritance, Mammal can have no siblings.

Inheritance And Polymorphism

Q49. Say that there are three classes: Computer, AppleComputer, and IBMComputer. What are likely relationships between these classes?

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

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

C. Computer, AppleComputer and IBMComputer are sibling classes.

D. Computer is a superclass, AppleComputer is a subclasses of Computer, and IBMComputer is a subclass of AppleComputer

Inheritance And Polymorphism

Q50. **class A**

```
{
        int i;
         A(int i)
        {
                this.i=i;
        }
}
```

class B extends A{}

Which of the following is true about the code above?

A. The code does not compile because there is no constructor defined in B

B. The code does not compile because no-argument constructor is not defined in A

C. The code does not compile because no-argument constructor is not defined in B

D. The code compiles fine

Inheritance And Polymorphism

Q51. **package a;**

public class A{

protected int i;

}}

package b;

public class B extends A{

void f(A a){}

}

Which of the following is accessible in the method f()?

A. this.i

B. a.i

C. super.i

D. None of the above.

Inheritance And Polymorphism

Q52. **class A {**

public static void f() {

System.out.println("fA"); }}

class B extends A{

public void f() {

System.out.println("fB"); }

public static void main(String[] args) {

A a= new B();

a.f(); }}

What will happen on compilation or execution of code?

A. fA

B. fB

C. Code will not compile

D. Code will throw runtime error

Inheritance And Polymorphism

Q53. **class D {}**

class T {

D d;

public Object getD(){ //line 1

return new Object(); //line 2

}}

class S extends T {

public D getD(){ // line 3

return new D();

}}

Identify the problems in the code?

A. Code will have compilation error at line 1

B. Code will have compilation error at line 2

C. Code will have compilation error at line 3

D. Code will compile successful

Inheritance And Polymorphism

Q54. **abstract class A {**

```
    static void addNo(int x, int y) {
    System.out.println(x+y); }
    public static void main(String[] a) {
    A.addNo(5, 10);
}}
```

What will happen on compilation or execution of code?

A. Code does not compile because abstract class must have at least one abstract method

B. Code does not compile because abstract class cannot have static method

C. Code does not compile because abstract class cannot have be instantiated

D. Code prints 15 on execution

Inheritance And Polymorphism

Q55. **class T extends String {**

public static void main(String[] a) {

String t = new T();

System.out.println(t);

}}

What is the result of compilation and execution of the code?

A. Code does not compile because inheritance from **String** is prohibited

B. Code does not compile because **String** class does not have no-argument constructor

C. Code does not compile because of invalid conversion of subclass object into super class object.

D. Code compiles clean

Inheritance And Polymorphism

Q56.
```
class A {
        int i;
        A() { i=0; }
       A(int i) { this.i=i; }
}
class B extends A {
        int j;
        B(int j) { this.i=10; this.j=j; }  //line 1
        B(){ this(5);  }
       public static void main(String[] a) {
                A b= new B();
                System.out.println(b.i+b.j);//line 2
}}
```

What is the result of compilation/execution of the code?

A. Compilation error at line 1

B. Compilation error at line 2

C. Prints: 15

D. Prints: 5

Inheritance And Polymorphism

Q57. **class B {String s1 = "Bs1"; String s2 = "Bs2";}**

```
class A extends B {
        String s1 = "As1";
        public static void main(String args[]) {
          A x = new A(); B y = (B)x;
          System.out.println(x.s1+" "+x.s2+" "+y.s1+" "+y.s2);
 }}
```

The code prints

A. Bs1 Bs2 Bs1 Bs2

B. As1 Bs2 As1 Bs2

C. As1 Bs2 Bs1 Bs2

D. As1 As2 Bs1 Bs2

Inheritance And Polymorphism

Q58. **class B {**
static void display(){ System.out.println("method B"); }
}
class A extends B {
static void display(){ System.out.println("method A"); }
public static void main(String args[]) {
//line 1
x.main();
}}

Which of the following statements in line 1 will print "method B"?

A. B x = new A();

B. A b=(A)new A();

C. B b=(B)(A)new A();

D. A b=(A) new B();

Inheritance And Polymorphism

Q59.
```
package p;
class B
{
        void main(){
        System.out.println("main B");
        }
}
```

The class inheriting from B which is in another package can override main() method as

A. public void main()

B. private Object main()

C. public int main()

D. protected void main()

Inheritance And Polymorphism

Q60.
```
class A {
    String s = "A";
    public void print(){System.out.print(s);}
    }
    class B extends A {
            String s = "B";
            public  void print(){ System.out.print(s);  }
            public static void main(String[] args) {
                    A b=new B();
                    b.print();
    }}
```

What happens on compilation and execution of the code?

A. Prints A

B. Prints B

C. Compilation error because of invalid overriding

D. Runtime error because of ClassCastException

Inheritance And Polymorphism

Q61. class A

```
{
        String s = “classA";
        final private void print(){
        System.out.print(s);
        }
}
```

Class inheriting from “A“ can have print() method declaration as

A. private void print()

B. final public void print()

C. void print()

D. None of the above since print method is final and cannot be overridden

Inheritance And Polymorphism

Q62. **class Tree {**

```
1.  int leaves;
2.  @Override
3.  public int equals(Object o){
4.  if(leaves==((Tree)o).leaves)
5.  return 0;
6.  else return 1;
7.  }}
```

What are the problems with the code listed above:

A. There is a warning by compiler for incorrect equals method

B. There is a compilation error because of incorrect overriding of equals method

C. A compilation error occurs at Line 5

D. There is no problem with the code

Inheritance And Polymorphism

Q63. **interface X{static void doIt();}**

public class Test1 implements X{

public static void doIt(){ System.out.println("OK");}

public static void main(String st[]){

Test1.doIt();

}}

Which of the following is true about the code above?

A. The code does not compile because of a problem in interface

B. The code does not compile of a problem in Test1 class

C. The code compiles and prints “OK” at execution time

D. The code compiles but throws a runtime error

Inheritance And Polymorphism

Q64. **interface X{**

```
int k=90;
int doIt();}
public class Test1 implements X{
int l=doIt();
// insert method declarartion for doIt()
{ return X.k;}
public static void main(String st[]){
        Test1 t=new Test1();
        t.doIt();
        System.out.print(t.l);
}}
```

Which of the following declaration(s) for doIt() in Test1 will compile the code?

A. int doIt()
B. public int doIt()
C. public static int doIt()
D. Code has syntax error and will not compile for any declaration of doIt()

Inheritance And Polymorphism

```
Q65.  interface X { int k=90; }                    \\Line1
      public class Test1 implements X{             \\Line2
      static{ System.out.print(k); }               \\Line3
      public static void main(String str[]) {      \\Line4
      }}
`
```

Inheritance And Polymorphism

Q66. Which of the following is true about interfaces?

A. Interface is an abstract class

B. Interface is always public

C. Method of interface must be abstract

D. Interface automatically inherits from Object class

Inheritance And Polymorphism

Q67. Given:

1. public interface Status {

2. /* insert code here */ int MY_VALUE = 10;

3. }

Which three are valid on line 2? (Choose three.)

A. final

B. abstract

C. public

D. static

Inheritance And Polymorphism

Q68. Which Man class properly represents the relationship "Man has a best friend who is a Dog"?

A. class Man extends Dog { }
B. class Man implements Dog { }
C. class Man { private BestFriend dog; }
D. class Man { private Dog bestFriend; }

Inheritance And Polymorphism

Q69. Given:

```
public interface A {
String DEFAULT_GREETING = “Hello World”;
public void method1(); }
```

A programmer wants to create an interface called B that has A as its parent.

Which interface declaration is correct?

A. public interface B extends A { }
B. public interface B implements A {}
C. public interface B instanceOf A {}
D. public interface B inheritsFrom A { }

Inheritance And Polymorphism

Q70. **Given:**

```
class TestA {
public void start() { System.out.println(”TestA”); } }
public class TestB extends TestA {
public void start() { System.out.println(”TestB”); }
public static void main(String[] args) {
((TestA)new TestB()).start();
}}
```

What is the result?

A. TestA

B. TestB

C. Compilation fails.

D. An exception is thrown at runtime.

Inheritance And Polymorphism

Q71. Given:

1. interface Data { public void load(); }

2. abstract class Info { public abstract void load(); }

Which class correctly uses the Data interface and Info class?

A.
```
public class Employee extends Info implements Data {
public void load() { /*do something*/ }
}
```

B.
```
public class Employee implements Info extends Data {
public void load() { /*do something*/ }
}
```

C.
```
public class Employee extends Info implements Data {
public void load() { /*do something */ }
public void Info.load() { /*do something*/ }
}
```

Inheritance And Polymorphism

Q72.

```
public class Test {
    int x= 12;
    public void method(int x) {
    x+=x;
    System.out.println(x);
    }
    public static void main(String as[]) {
            Test t = new Test();
            t.method(5);
}}
```

What is the output from line 5 of the Test class?

A. 5

B. 10

C. 12

D. 17

Inheritance And Polymorphism

Q73. **interface A {**

```
        int a = 1;              //Line  1
        public int b = 2;       // Line 2
        public static int c = 3;  // Line 3
}
```

Which field declaration results in a compile-time error?

A. 1

B. 2

C. 3

D. None of the above

Inheritance And Polymorphism

Q74. **interface I1 {String name = "I1"; }**

interface I2 {String name = "I2"; }

class X implements I1, I2 { **//line 1**

public static void main(String[] args) {

System.out.print(name); **//line 2**

}}

What is the result of attempting to compile and run the program?

A. I1

B. I2

C. Compilation error at line 1

D. Compilation error at line 2

Inheritance And Polymorphism

Q75. **interface I {final void f();}**

abstract class X implements I{

public final void f(){}

public static void main(String[] args) {

new X().f();}}

Spot all of the problems in code?

A. Interface I cannot have final method

B. Class X cannot be declared as abstract since it implements interface

C. Method f() cannot be overridden in class X

D. Instance of X cannot be created

Inheritance And Polymorphism

Q76.

```
public interface X {                                    //Line1
        void f(); }                                     //Line2
abstract class A implements X {                         //Line3
        public final void f() {}                        //Line4
        public static void main(String[] args) {        //Line5
}}
```

Which lines will cause compilation error?

A. Line 1

B. Line 3

C. Line 4

D. No compilation error

Inheritance And Polymorphism

Q77. **interface Y {}**

```
public class X{
static String s;
public static void main(String[] args) {
X a1= new X();
System.out.print(a1 instanceof X);          //Line 1
System.out.print(a1 instanceof Y);          //Line 2
System.out.print(s instanceof Y);           //Line 3
System.out.print(null instanceof Y);        //Line 4
}}
```

Which of the following is true about the code?

A. Line 1 prints true

B. Line 2 prints false

C. Line 3 gives compilation error

D. Line 4 prints false

Inheritance And Polymorphism

Q78. **interface A {int i = 1; int m1();}**

interface B extends A {int i = 10; int m1();}

class X implements B {

public int m1() {return i;}

public static void main(String[] args) {

System.out.print(new X().m1());

}}

What happens on compilation and execution of the code?

A. Prints 1

B. Prints 10

C. Compilation error

D. Runtime error

Inheritance And Polymorphism

Q79. **interface I1 { }**

interface I2 {}

class C1 implements I1{}

class C2 implements I2 {}

class C3 extends C1 implements I2 {}

Assume the following declarations.

C1 obj1; C2 obj2; C3 obj3;

Which of the following are valid assignments?

A. obj2 = obj1;

B. I1 a = obj2;

C. I2 a = obj2;

D. I2 c = obj1;

Inheritance And Polymorphism

Q80. Which of the following statements about abstract methods/classes in Java are FALSE?

A. An abstract class cannot be instantiated.

B. Constructors cannot be abstract.

C. A subclass of an abstract class must define the abstract methods.

D. Static methods cannot be declared abstract.

Inheritance And Polymorphism

Q81. The concept of multiple inheritance is implemented in Java by

A. Multiple Inheritance is not at all possible in Java

B. Extending one class and implementing one or more interfaces

C. Implementing two or more interfaces

D. Both option B and option C

Inheritance And Polymorphism

Q82. ”super” keyword can be used to ?

A. Call super class’s constructor
B. Access super class’s member
C. Both option A and option B
D. None of these

Inheritance And Polymorphism

Q83. A class can implement _____ interfaces

A. Only one

B. One or more than one

C. Minimum 2

D. A class cannot implement an interface

Inheritance And Polymorphism

Q84. A class implements an interface but does not override all the methods of that interface, then _____

A. it should be declared as abstract class

B. it should be declared as final class

C. it will compile successfully

D. None of these

Inheritance And Polymorphism

Q85. Which of the following statements is true?

A. Static, final and private methods can be abstract

B. A final class can be inherited

C. Static methods cannot be overridden

D. All are true

Session2: Exercises

Hands-On Exercises

Session2: Exercises

Q1. Write a program to add two complex number using this reference.

Q2. Write a program to prove the order of initialization for static block.

Q3. Create a class A. add public, private and protected member variable and methods. Create another class B which extends from A and add few members. Try to access a private ,public and protected members of a base class within a class B. observe the result.

Session2: Exercises

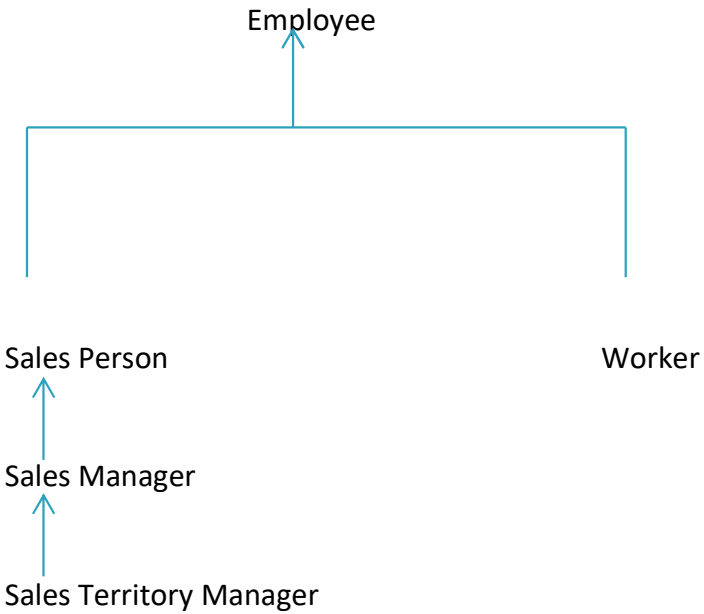
Q4. Create two classes, A and B, with default constructors (empty argument lists) that announce themselves. Inherit a new class called C from A, and create a member B inside C. Do not create a constructor for C. Create an object of class C and observe the results.

Q5. Modify above Exercise so that A and B have constructors with arguments instead of default constructors. Write a constructor for C and perform all initialization within C's constructor.

Q6. Create an inheritance hierarchy of Wipro: Wipro Technologies, Wipro Infotech , Wipro BPO etc. In the base class, provide methods that are common to all Wipro, and override these in the derived classes to perform different behaviors depending on the specific type of Wipro. Create an array of Wipro, fill it with different specific types of Wipro, and call your base-class methods to see what happens.

Session2: Exercises

Q7. Write an application to implement a Pay Roll System for an Organization



Each will have Date class to represent Date of Joining

Session2: Exercises

Q8. Create an abstract class with no abstract methods. Verify that you cannot create instances of the class.

Q9. Create an interface with 3 methods in a package. Implement this in a class in different package.

Q10. Create a class in a package. Write a protected method in it. From outside the package, try to call the method and see the result. Now inherit from your class and call the protected method from inside a method of your derived class.