**Interfaces Questions**

Question 1: What is wrong with the following interface?

public interface SomethingIsWrong {

void aMethod(int aValue) {

System.out.println("Hi Mom");

}

}

**Answer : It has a method implementation in it. Only default and static methods have implementations.**

**2) Can you identify the error in the below code?**

interface A

{

    private int i;

}

**Answer**: Illegal modifier for field i. Only public, static and final are allowed.

**3) What will be the output of the following program?**

interface A

{

    void myMethod();

}

class B

{

    public void myMethod()

    {

        System.out.println("My Method");

    }

}

class C extends B implements A

{

}

class MainClass

{

    public static void main(String[] args)

    {

        A a = new C();

        a.myMethod();

    }

}

My Method

**4. Why the below code is showing compile time error?**

interface X

{

    void methodX();

}

class Y implements X

{

    void methodX()

    {

        System.out.println("Method X");

    }

}

Interface methods must be implemented as public. Because, interface methods are public by default and you should not reduce the visibility of any methods while overriding.

5. **Does below code compile successfully? If not, why?**

interface A

{

    int i = 111;

}

class B implements A

{

    void methodB()

    {

        i = 222;

    }

}

No, because interface fields are static and final by default and you can’t change their value once they are initialized. In the above code, methodB() is changing value of interface field A.i. It shows compile time error.

**6. Is the following code written correctly?**

class A

{

    //Class A

}

interface B extends A

{

    //Interface B extending Class A

}

No. An interface can extend another interface not the class.

7. **What will be the output of the following program?**

interface P

{

    String p = "PPPP";

    String methodP();

}

interface Q extends P

{

    String q = "QQQQ";

    String methodQ();

}

class R implements P, Q

{

    public String methodP()

    {

        return q+p;

    }

    public String methodQ()

    {

        return p+q;

    }

}

public class MainClass

{

    public static void main(String[] args)

    {

        R r = new R();

        System.out.println(r.methodP());

        System.out.println(r.methodQ());

    }

}

QQQQPPPP  
PPPPQQQQ

**8. Can interfaces have constructors? – No**

**9.** **Is the below program written correctly? If yes, what will be the output?**

class A implements B

{

    public int methodB(int i)

    {

        return i =+ i \* i;

    }

}

interface B

{

    int methodB(int i);

}

public class MainClass

{

    public static void main(String[] args)

    {

        B b = new A();

        System.out.println(b.methodB(2));

    }

}

4

**10. Can you find out the errors in the following code?**

interface A

{

    {

        System.out.println("Interface A");

    }

    static

    {

        System.out.println("Interface A");

    }

}

Interfaces can’t have initializers.

11. **How do you access interface field ‘i’ in the below code?**

class P

{

    interface Q

    {

        int i = 111;

    }

}

P.Q.i

12. **What will be the output of the following program?**

interface X

{

    char c = 'A';

    char methodX();

}

class Y implements X

{

    {

        System.out.println(c);

    }

    public char methodX()

    {

        char c = this.c;

        return ++c;

    }

}

public class MainClass

{

    public static void main(String[] args)

    {

        Y y = new Y();

        System.out.println(y.methodX());

        System.out.println(y.c);

        System.out.println(X.c);

    }

}

A  
B  
A  
A

13. **What will be the output of the following program?**

interface One

{

    String s = "FINAL";

    String methodONE();

}

interface Two

{

    String methodONE();

}

abstract class Three

{

    String s = "NOT FINAL";

    public abstract String methodONE();

}

class Four extends Three implements One, Two

{

    public String methodONE()

    {

        String s = super.s + One.s;

        return s;

    }

}

public class MainClass

{

    public static void main(String[] args)

    {

        Four four = new Four();

        System.out.println(four.methodONE());

        One one = four;

        System.out.println(one.s);

    }

}

NOT FINALFINAL  
FINAL

**14. What will be the output of the following program?**

interface A

{

    int methodA();

}

interface B

{

    int methodB();

}

interface C

{

    int methodC();

}

class D implements A, B, C

{

    int i = 999+111;

    public int methodA()

    {

        i =+ i / i;

        return i;

    }

    public int methodB()

    {

        i =- i \* i;

        return i;

    }

    public int methodC()

    {

        i = ++i - --i;

        return i;

    }

}

public class MainClass

{

    public static void main(String[] args)

    {

        D d = new D();

        System.out.println(d.i);

        System.out.println(d.methodA());

        System.out.println(d.methodB());

        System.out.println(d.methodC());

    }

}

1110  
1  
-1  
1

**15. Is the below program written correctly? If yes, what will be the output?**

interface X

{

    void methodX();

    interface Y

    {

        void methodY();

    }

}

class Z implements X, X.Y

{

    {

        methodX();

        System.out.println(1);

    }

    public void methodX()

    {

        methodY();

        System.out.println(2);

    }

    public void methodY()

    {

        System.out.println(3);

    }

}

public class MainClass

{

    public static void main(String[] args)

    {

        Z z = new Z();

        z.methodX();

        z.methodY();

        X x = z;

        x.methodX();

    }

}

3  
2  
1  
3  
2  
3  
3  
2

**16. Can you identify the error in the below code?**

interface X

{

    void methodX();

}

interface Y extends X

{

    void methodY();

}

class Z implements Y

{

    public void methodY()

    {

        System.out.println("Method Y");

    }

}

Class Z must implement methodX() also.

**17. Is the below program written correctly? If yes, what will be the output?**

interface I

{

    class C

    {

        int i;

        public C(int i)

        {

            this.i = ++i;

        }

        int methodC()

        {

            return ++i;

        }

    }

}

public class MainClass

{

    public static void main(String[] args)

    {

        I.C c = new I.C(000);

        System.out.println(c.methodC());

    }

}

2

**18. What will be the output of the following program?**

class A { }

class B extends A { }

class C extends B { }

interface ABC

{

    void method(A a);

}

interface PQR

{

    void method(B b);

}

class M implements ABC, PQR

{

    public void method(A a)

    {

        System.out.println(2);

    }

    public void method(B b)

    {

        System.out.println(3);

    }

}

public class MainClass

{

    public static void main(String[] args)

    {

        M m = new M();

        m.method(new A());

        m.method(new B());

        m.method(new C());

    }

}

2  
3  
3