**Interfaces Questions**

Question 1: What is wrong with the following interface?

public interface SomethingIsWrong {

void aMethod(int aValue) {

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

}

}

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

interface A

{

    private int i;

}

**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();

    }

}

**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");

    }

}

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

interface A

{

    int i = 111;

}

class B implements A

{

    void methodB()

    {

        i = 222;

    }

}

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

class A

{

    //Class A

}

interface B extends A

{

    //Interface B extending Class A

}

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());

    }

}

**8. Can interfaces have constructors?**

**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));

    }

}

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

interface A

{

    {

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

    }

    static

    {

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

    }

}

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

class P

{

    interface Q

    {

        int i = 111;

    }

}

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);

    }

}

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);

    }

}

**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());

    }

}

**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();

    }

}

**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");

    }

}

**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());

    }

}

**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());

    }

}