

**Question 1- C**

```
public class Movie extends Cinema {  
    public Movie(String movie) { }
```

Two line give us compilation error.

**Question 2 -D**

The public access specifier indicates that the interface can be used by any class in any package. If you do not specify that the interface is public, then your interface is accessible only to classes defined in the same package as the interface.

<https://docs.oracle.com/javase/tutorial/java/IandI/interfaceDef.html>

**Question 3 – A**

```
public void playMusic() {  
    System.out.print("Play!") output result is Play!
```

**Question 4 -A**

Inheritance allow us when using object attributes and methods used to acces commonly.

<https://www.geeksforgeeks.org/inheritance-in-java/>

**Question 5 - ?**

**Question 6 – B**

An interface is a completely "abstract class" that is used to group related methods with empty bodies.

[https://www.w3schools.com/java/java\\_interface.asp](https://www.w3schools.com/java/java_interface.asp)

**Question 7 – D**

This code isn't compiling.

**Question 8 – D**

Java does not support multiple inheritance. > False statement

Java allows multiple inheritance using abstract classes. > False statement

Java allows multiple inheritance using non-abstract classes. > False statement

Java allows multiple inheritance using interfaces. > True statement. Interface allow us for using a few inheritance.

**Question 9 – ?**

**Question 10 – C**

In any object-oriented programming language, Overriding is a feature that allows a subclass or child class to provide a specific implementation of a method that is already provided by one of its super-classes or parent classes. When a method in a subclass has the same name, same parameters or signature, and same return type(or sub-type) as a method in its super-class, then the method in the subclass is said to override the method in the super-class.

<https://www.geeksforgeeks.org/overriding-in-java/>

**Question 11 – C**

This code isn't compiling.

**Question 12 – 2**

System.out.print(school.getNumberOfStudentsPerClassroom()) output result is 2.

**Question 13 – B**

Static is applied to interface method.

**Question 14 – C**

System.out.print("Sprinting!") output result is Sprinting!

**Question 15 -B**

Interface can implement another interface statement is false.

**Question 16 - D**

Code isn't compile. Exception in thread "main" java.lang.Error: Unresolved compilation problem.

**Question 17 – ?**

**Question 18 – C**

final Triangle shape = new IsoscelesRightTriangle() line g3 isn't compiling.

**Question 19 – D**

Play is must be class.

**Question 20 – C**

Interface allow us extends an abstract class.

**Question 21 – A**

Code isn't compile. Exception in thread "main" java.lang.Error: Unresolved compilation problem.

**Question 22 – ?**

**Question 23 – C**

Default value is allows modifiers to applited to an abstract method.

**Question 24 – D**

Code isn't compile. Exception in thread "main" java.lang.Error: Unresolved compilation problem.

**Question 25 - ?**

**Question 26 -**

**Question 27 -**

**Question 28 -**

**Question 29 -**

**Question 30 –**

**Question 31 –**

**Question 32 –**

**Question 33 -**

**Question 34 -**

**Question 35 -**

**Question 36 –**

**Question 37 -**

**Question 38 –**

**Question 39 –**

**Question 40 –**

**Question 41 –**

**Question 42 –**

**Question 43 –**

**Question 44 –**

**Question 45 –**

**Question 46 –**

**Question 47 –**

**Question 48 –**

**Question 49 -**

**Question 50 -**