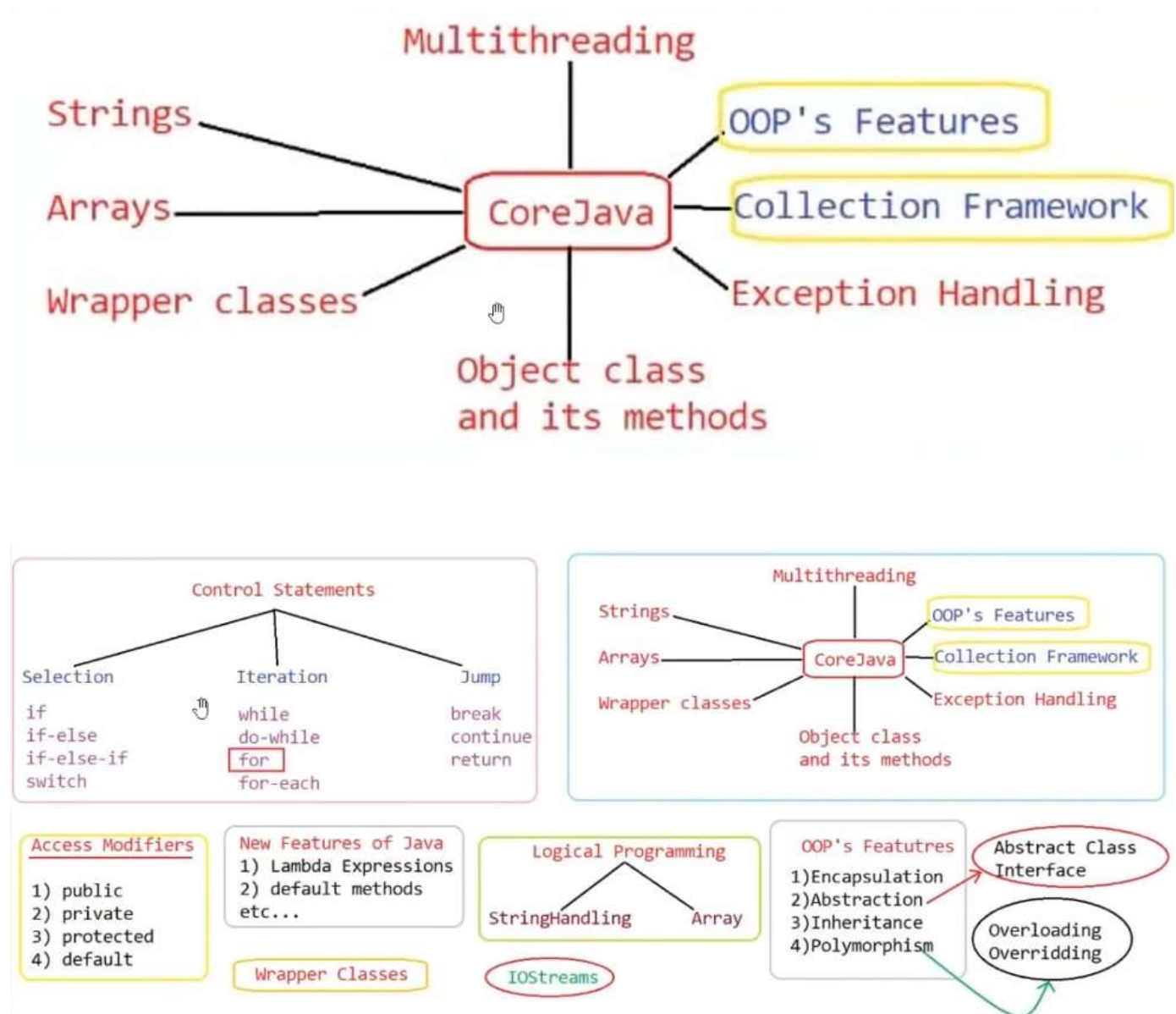
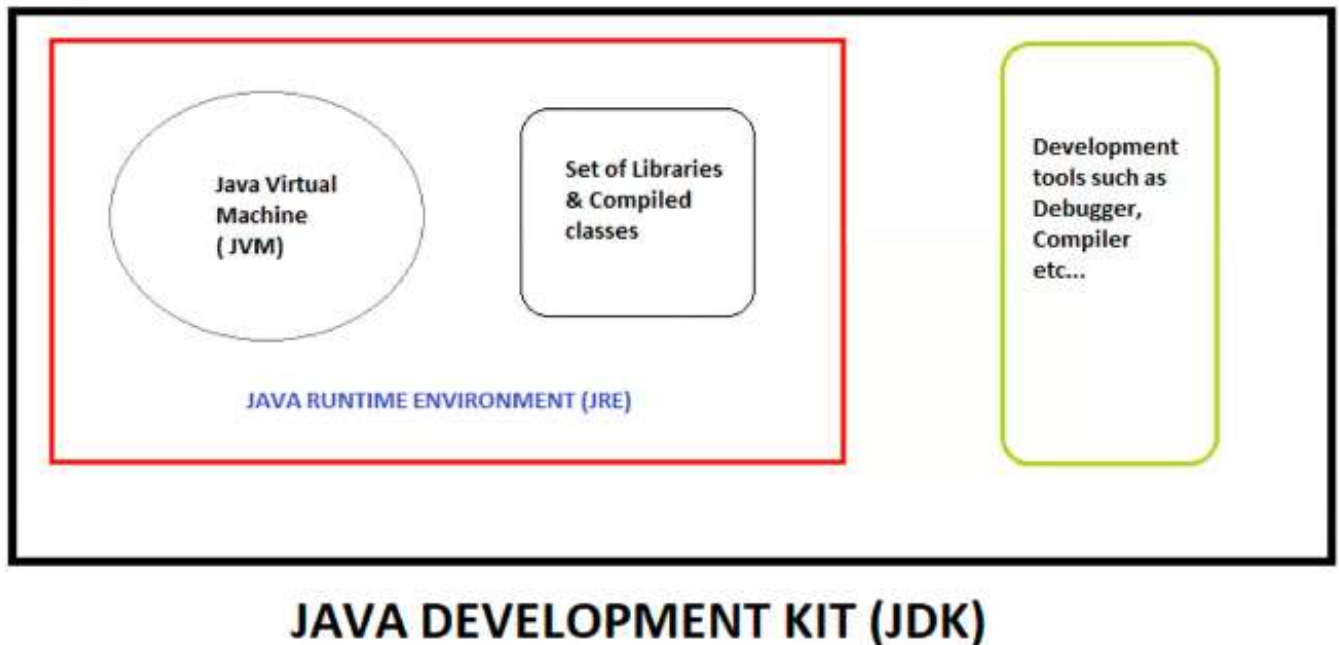


Important topics for interviewer



Difference between JVM, JRE & JDK?



JVM vs. JRE vs. JDK

Java Virtual Machine (JVM):

JVM is a virtual machine which provides a runtime environment for executing the Java bytecode

Java Runtime Environment (JRE):

JRE is an environment within which the JVM runs and has class libraries and other files that Java Virtual Machine uses at the time of execution.

JRE = Java Virtual Machine (JVM) + Libraries to run the application

Java Development Kit (JDK):

JDK is the parent set of the JRE and has everything that JRE contains along with the development tools such as a compiler, debugger etc.

JDK = Java Runtime Environment (JRE) + Development tools

Every java predefined class will be present inside JRE.

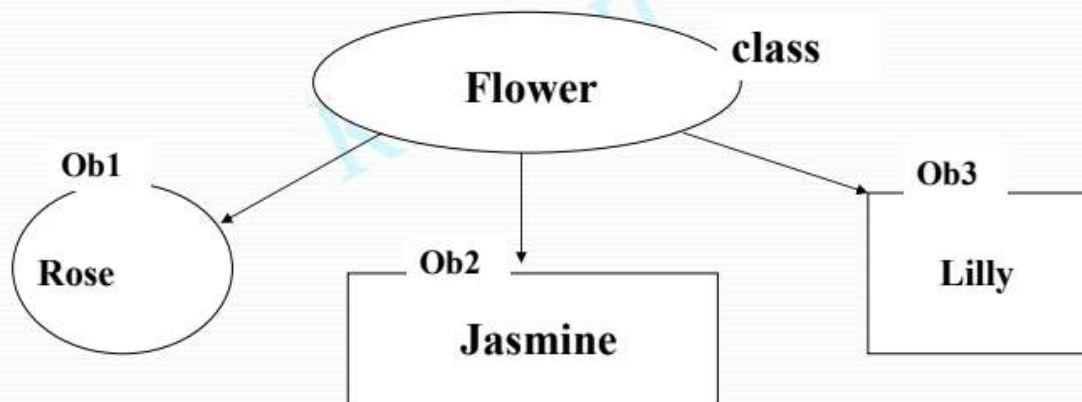
Why you want to choose java developer as a career

The Java Buzzwords

- The key considerations were summed up by the Java team in the following list of buzzwords:
 - ❖ Simple
 - ❖ Object-oriented
 - ❖ Robust
 - ❖ Multithreaded
 - ❖ Architecture-neutral
 - ❖ Interpreted & High performance
 - ❖ Distributed
 - ❖ Dynamic
 - ❖ Secure
 - ❖ Portability

CLASS

- Class is blue print or an idea of an Object
- From One class any number of Instances can be created
- It is an encapsulation of attributes and methods

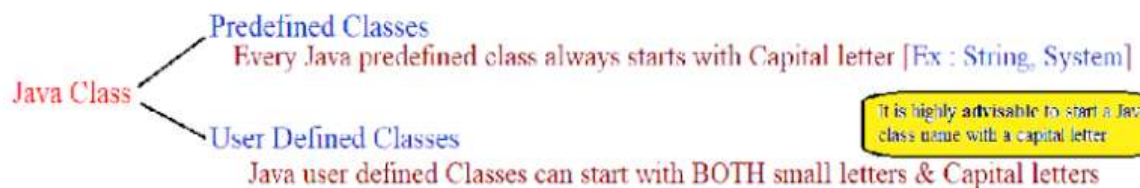


Understanding Java Class:

- 1) A Class is an Idea of an Object.
- 2) An Object is a real world entity.
- 3) Syntax of a Java Class:

```
<AccessModifier>class<ClassName>
{
}
}
```

4)



It is highly advisable to start a Java class name with a capital letter

```

1 package com.pack1;
2
3 public class LaptopFactory
4 {
5     String processor;
6     int ram;
7     int screenSize;
8
9     public LaptopFactory(String laptop_processor, int laptop_ram, int laptop_screenSize)
10    {
11        processor=laptop_processor;
12        ram=laptop_ram;
13        screenSize=laptop_screenSize;
14    }
15    public static void main(String[] args)
16    {
17        LaptopFactory lenovo=new LaptopFactory("i5",256,15);
18        LaptopFactory dell=new LaptopFactory("i7", 512, 14);
19
20        System.out.println("Lenovo Details : "+lenovo.processor+" "+lenovo.ram+" "+lenovo.screenSize);
21        System.out.println("dell Details : "+dell.processor+" "+dell.ram+" "+dell.screenSize);
22    }
23 }

```

```

Lenovo Details : i5 256 15
dell Details : i7 512 14

```

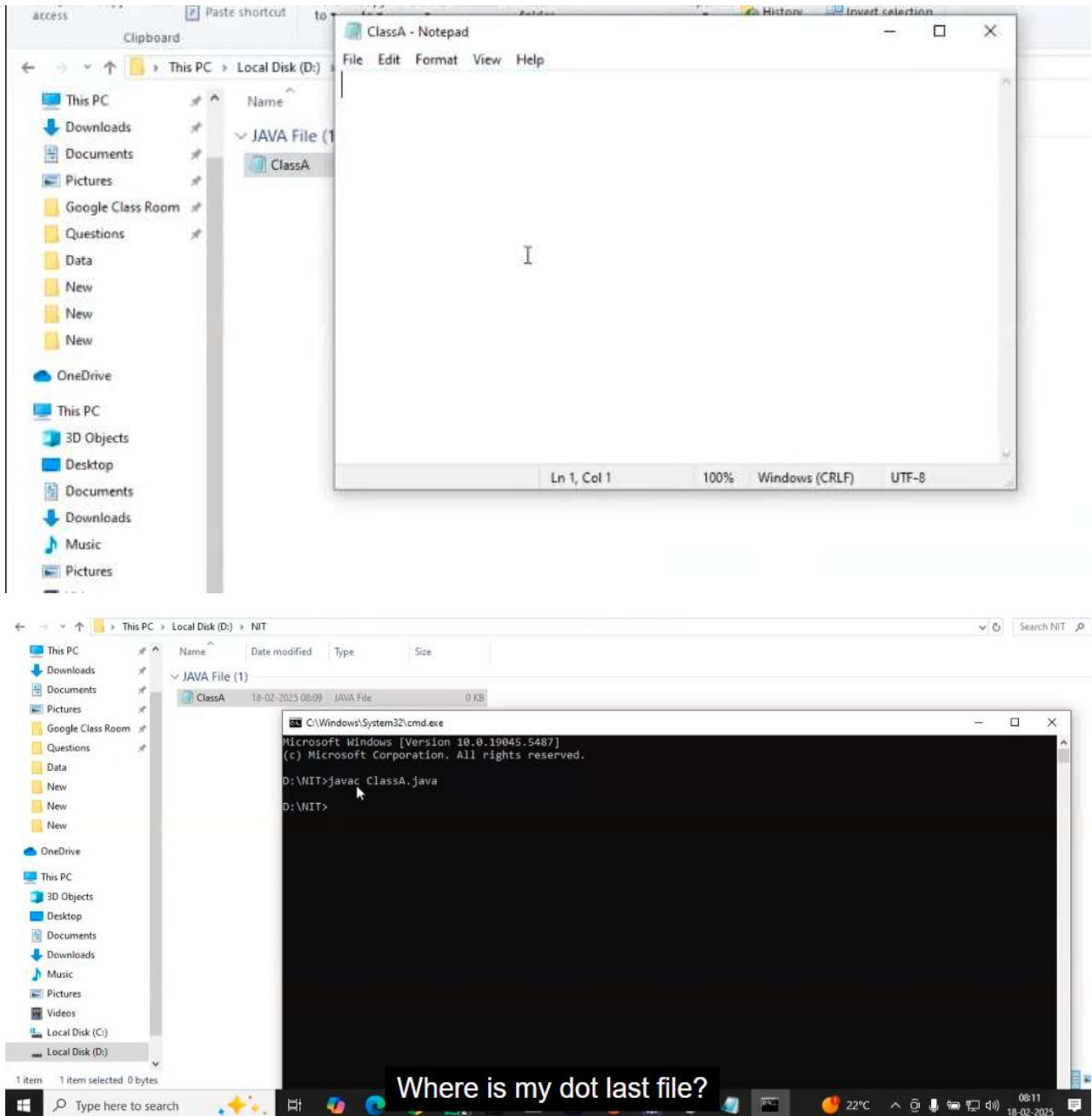
We need to keep in mind what is that object which we want to create and we need those features, variables, methods, constructors, blocks related to that object and these written in side a class

Basic Questions

- 1) Can we create an empty java file and compile and run it?
- 2) Can we create an empty java class and compile and run it?
- 3) Is it mandatory that java class name and File Name should be same?
- 4) What are user defined Classes/ Methods and Pre Defined Classes/Methods?
- 5) Does JVM executes user defined methods automatically?
- 6) How many user defined methods we can write in a class and what is the order of their execution?
- 7) How can we call the user defined methods present in class?

1. Can we create an empty java file and compile and run it?

Ans: we can create an empty java file, we can compile .class file here but .class file is not generated why because there is no class as there is no .class file (.class file name should be same as class name) we will not be able to run it.

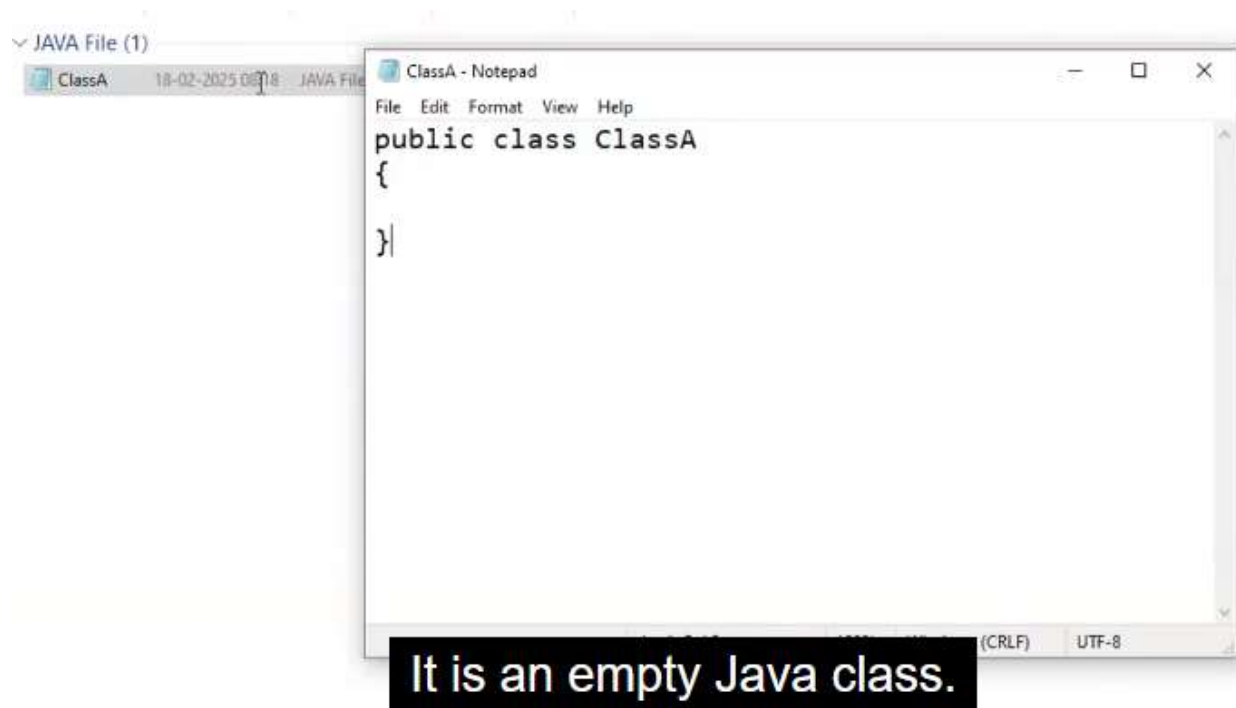


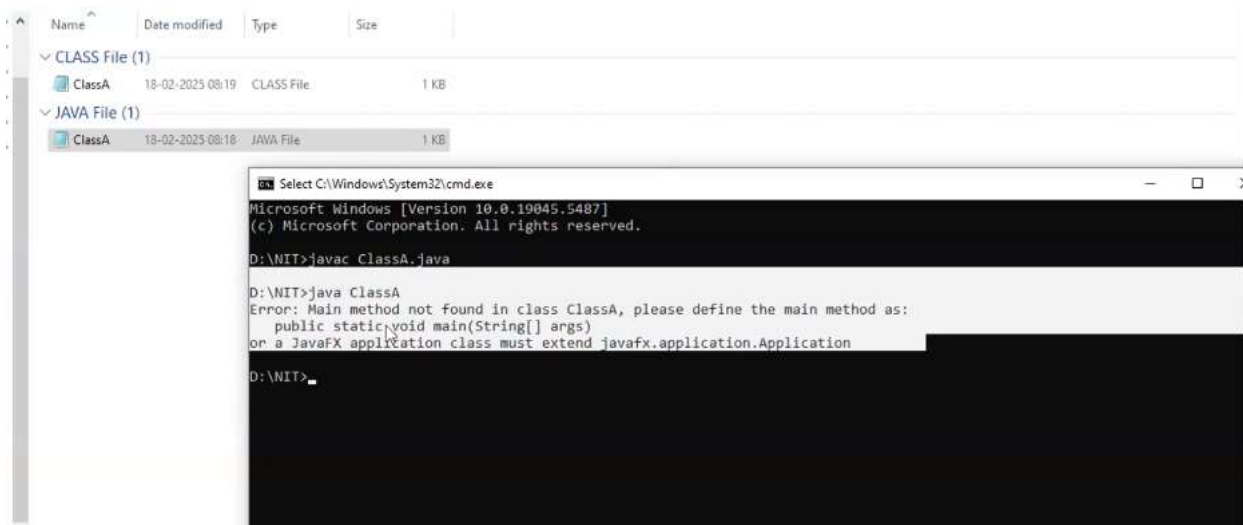
.class file is not generated, .class file name will be same as class name, but we don't have any class here.

2. Can we create an empty java class and compile and run it?

Ans: yes, we can create an empty java class, we can compile a .class file will be generated but we will not be able to run it. Why because there is no main method

Every java program execution will start from main method and end with main method.





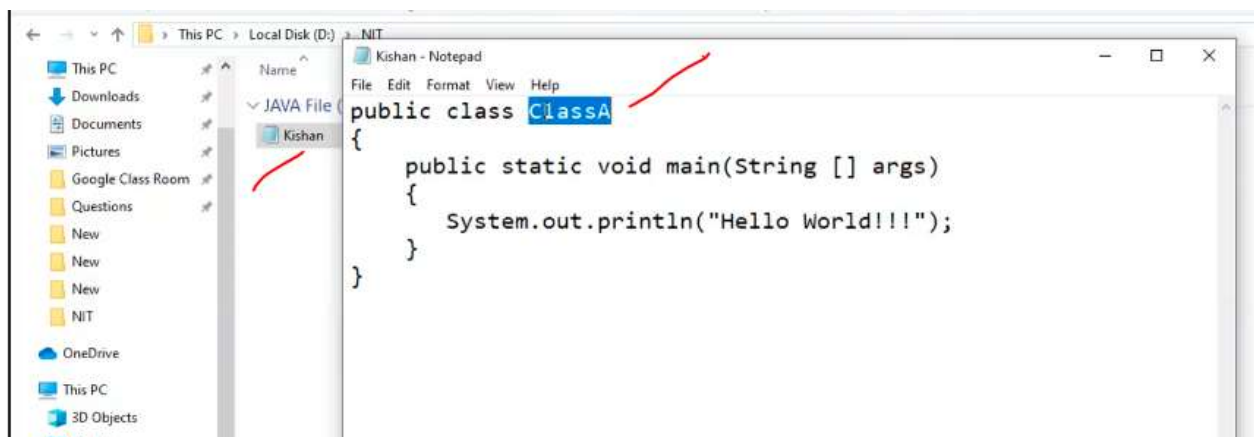
3. Is it mandatory that java class name and file name should be same?

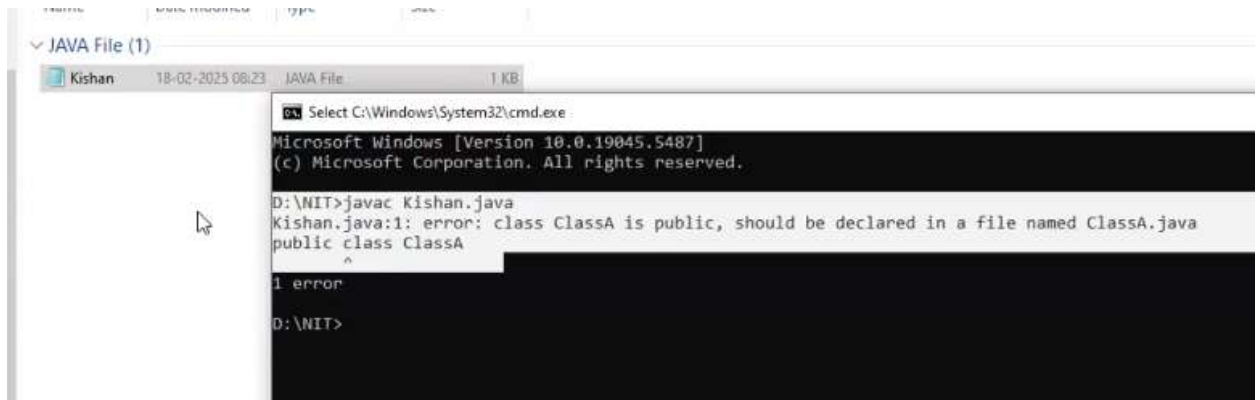
Ans: It is mandatory if the class is declared as public.

It is not mandatory if the class is declared as default(package-private).

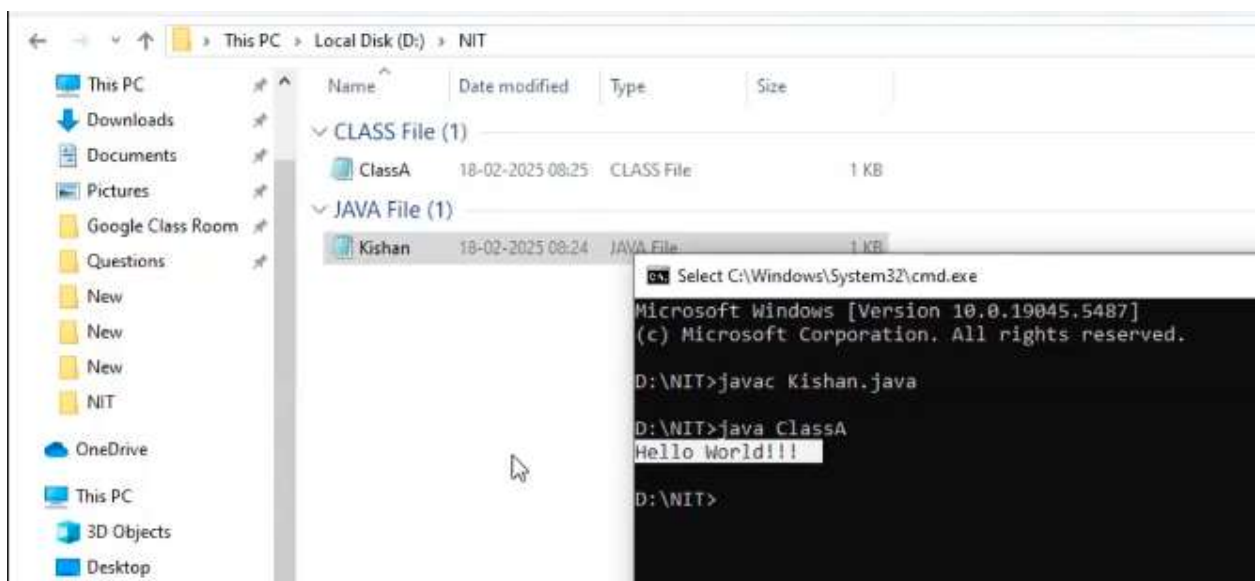
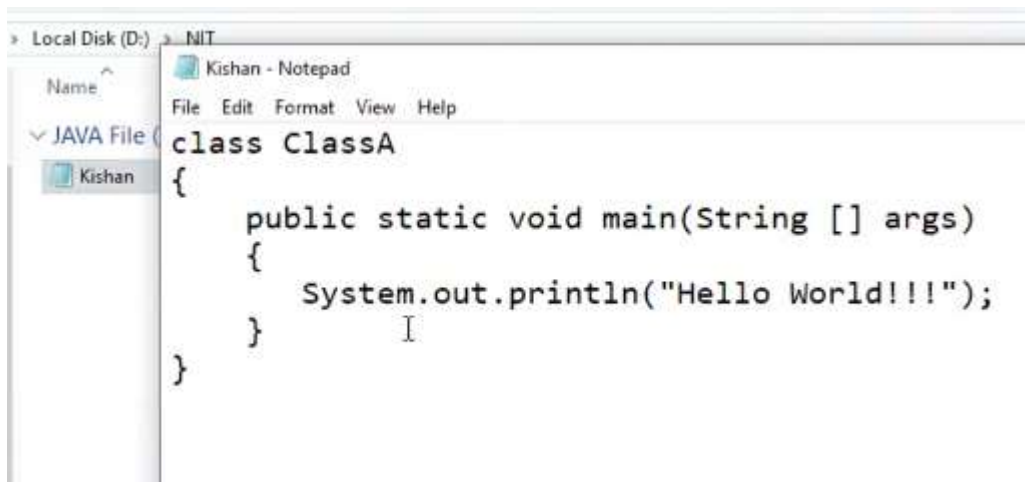
But it is highly advisable that same file name as class name.

Case-1





Case-2



4. What is user defined classes/methods and predefined classes/methods?

Ans: The classes which were given by Sun Microsystems and the methods which are present inside that class are known as pre-defined methods

The classes which are given by the programmers and the methods which are present in those are user defined methods

5. Does JVM execute user defined methods automatically?

Ans: NO, every method will be executed only whenever we are calling it. It may be a predefined method or user defined method except main method.

6. How many user defined methods we can write in a class and what is the order of their execution?

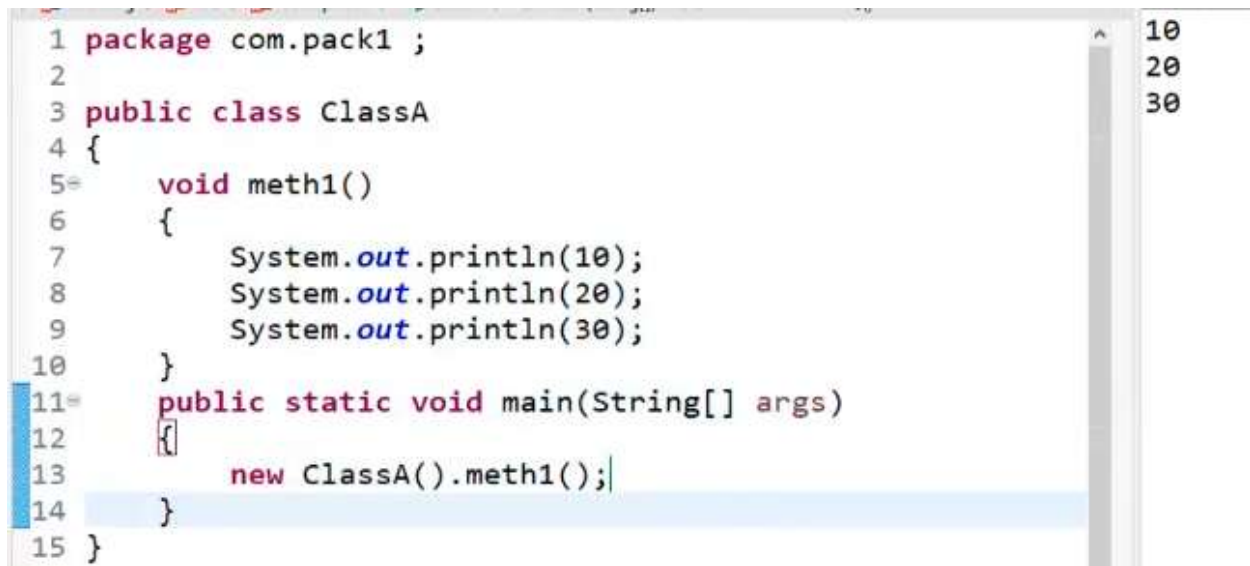
Ans: We can write any number of methods in a class.

In which order we are calling in that order only those will be getting executed obviously main method will executes first

And after that, based on our requirements, the other methods will be executed.

7. How can we call the user defined methods present in a class?

Ans: We need to call a user defined methods present in a class with help of its respective Class Object. To create an object first we need to write new keyword and initialize that object with the help of a constructor. Constructor name will be the same as the class name.



```
1 package com.pack1 ;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.println(10);
8         System.out.println(20);
9         System.out.println(30);
10    }
11    public static void main(String[] args)
12    {
13        new ClassA().meth1();
14    }
15 }
```

```

1 package com.pack1 ;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.print(10);
8         System.out.print(20);
9         System.out.print(30);
10    }
11    public static void main(String[] args)
12    {
13        new ClassA().meth1();
14    }
15 }
16

```

102030 I

```

1 package com.pack1 ;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.print("Java is ");
8         System.out.print("Object Oriented Programming
9         System.out.println(" language");
10        System.out.println("Java is Awesome");
11        System.out.println("I Love Java");
12    }
13    public static void main(String[] args)
14    {
15        new ClassA().meth1();
16    }
17 }
18

```

Java is Object Oriented Programming language
Java is Awesome
I Love Java