

In order to see the output of our program we need to perform '2' tasks

1) Compilation

2) Running

Compilation: [Java Compiler ==> javac FileName.java]

1) Compilation means checking whether the code we have written according to the Java language syntax or not.

2) After successful compilation Java compiler is going to generate a .class file.

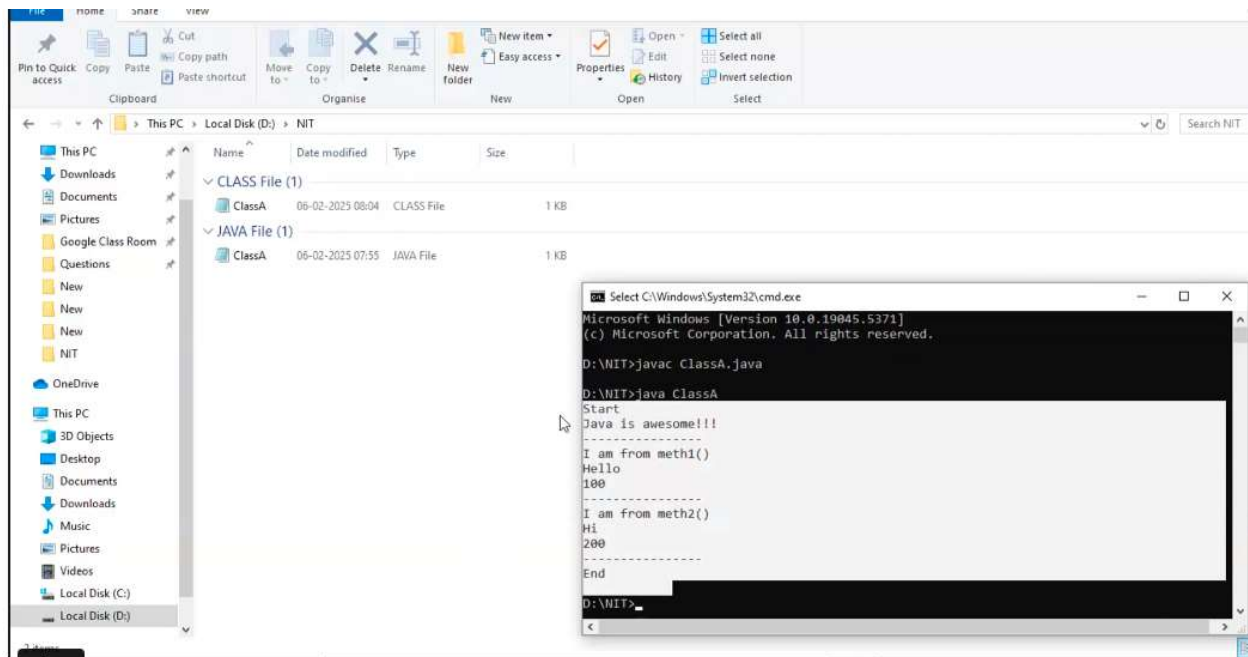
3) The generated .class file name will be same as Class name

4) The generated .class file consists of 'bytecode' instructions which are understandable only by the machines humans cannot understand those instructions.

Running : [Java Virtual Machine : JVM ==> java generated.className]

1) In order to run our Java program, we need to send the generated .class file as an input to the JVM.

2) JVM is going to check whether all the byte code instructions which are present in the .class file are correct (or) wrong, if correct we will be getting output if wrong, we will be getting Exception



IDE – Integrated Development Environment

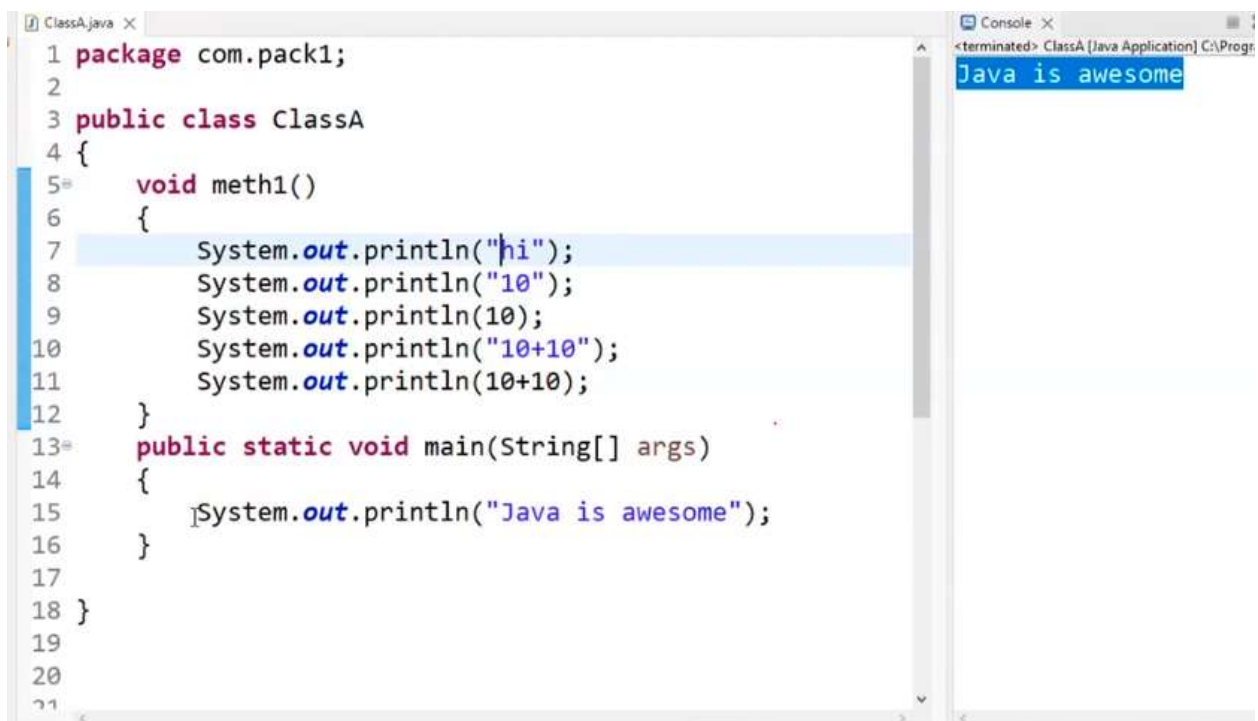
IDE is place where we can write the code effectively

There are lots of IDE in the market

Eclipse, my eclipse, net beam & intel j

package com.pack1;

Package consists of similar types of classes, interfaces and enum.



The screenshot shows a Java IDE with two windows. The left window, titled 'ClassA.java', contains the following code:

```
1 package com.pack1;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.println("hi");
8         System.out.println("10");
9         System.out.println(10);
10        System.out.println("10+10");
11        System.out.println(10+10);
12    }
13    public static void main(String[] args)
14    {
15        System.out.println("Java is awesome");
16    }
17 }
18
19
20
21
```

The right window, titled 'Console', shows the output of the program: 'Java is awesome'.

```
ClassA.java X Console X
1 package com.pack1;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.println("hi");
8         System.out.println("10");
9         System.out.println(10);
10        System.out.println("10+10");
11        //System.out.println(10+10);
12    }
13    public static void main(String[] args)
14    {
15        ClassA aobj=new ClassA();
16        aobj.meth1();
17    }
18
19 }
20
21
```

<terminated> ClassA [Java Appli

hi
10
10
10+10

```
ClassA.java X Console X
1 package com.pack1;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.println("hi");
8         System.out.println("10");
9         System.out.println(10);
10        System.out.println("10+10");
11        System.out.println(10+10);
12    }
13    public static void main(String[] args)
14    {
15        ClassA aobj=new ClassA();
16        aobj.meth1();
17    }
18
19 }
20
21
```

<terminated> ClassA [

hi
10
10
10+10
20

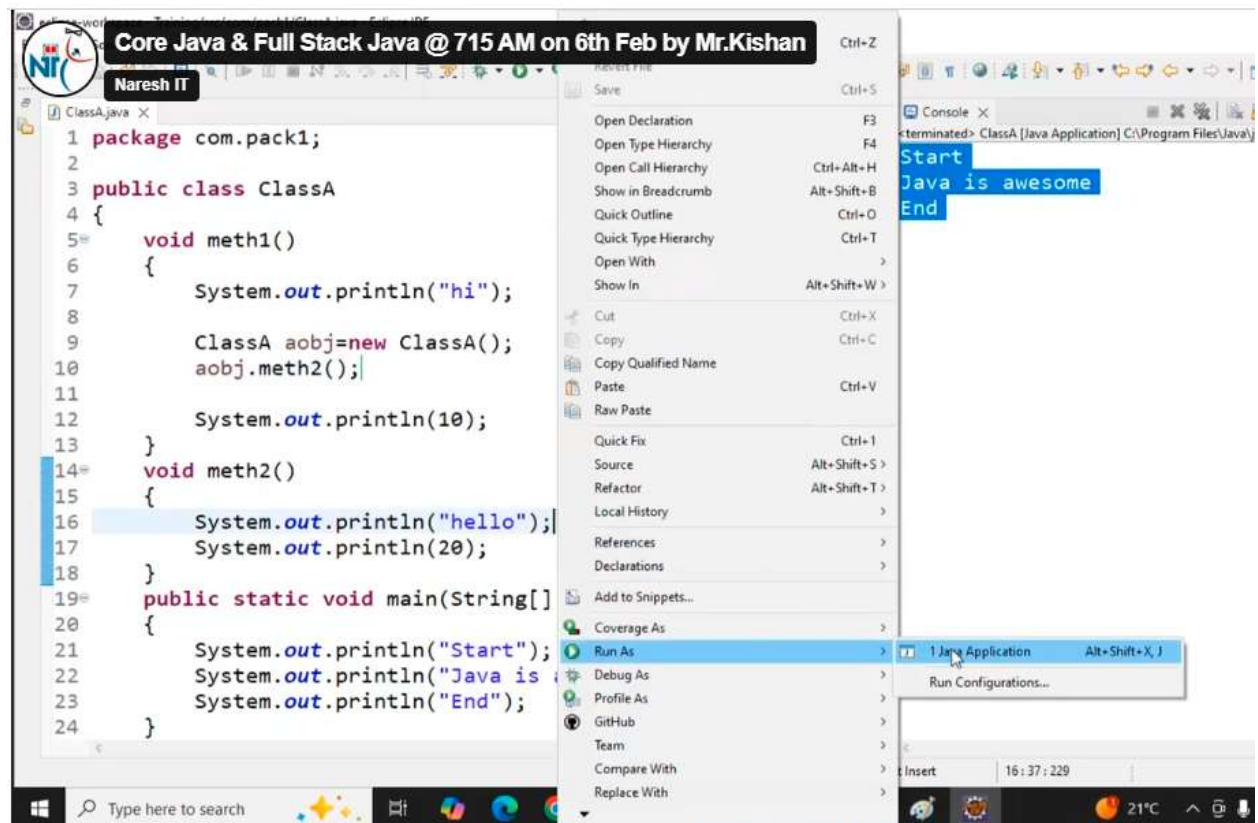


The screenshot shows an IDE with a file named `ClassA.java` open. The code is as follows:

```
1 package com.pack1;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.println("hi");
8         System.out.println(10);
9     }
10    void meth2()
11    {
12        System.out.println("hello");
13        System.out.println(20);
14    }
15    public static void main(String[] args)
16    {
17        System.out.println("Start");
18        System.out.println("Java is awesome");
19        System.out.println("End");
20    }
21
22 }
23
24
```

The console on the right shows the output of the program:

```
<terminated> ClassA [Java Application]
Start
Java is awesome
End
```



Core Java & Full Stack Java @ 715 AM on 6th Feb by Mr.Kishan

Naresh IT

```
1 package com.pack1;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.println("hi");
8
9         ClassA aobj=new ClassA();
10        aobj.meth2();
11
12        System.out.println(10);
13    }
14    void meth2()
15    {
16        System.out.println("hello");
17        System.out.println(20);
18    }
19    public static void main(String[] args)
20    {
21        System.out.println("Start");
22        System.out.println("Java is awesome");
23        System.out.println("End");
24    }
```

Console

```
<terminated> ClassA [Java Application] C:\Program Files\Java
Start
Java is awesome
End
```

Core Java & Full Stack Java @ 715 AM on 6th Feb by Mr.Kishan

Naresh IT

```
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.println("hi");
8
9         ClassA aobj=new ClassA();
10        aobj.meth2();
11
12        System.out.println(10);
13    }
14    void meth2()
15    {
16        System.out.println("hello");
17        System.out.println(20);
18    }
19    public static void main(String[] args)
20    {
21        System.out.println("Start");
22        System.out.println("Java is awesome");
23        aobj.meth1();
24        System.out.println("End");
25    }
```

Console

```
<terminated> ClassA [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe
Start
Java is awesome
End
```

Writable Smart Insert 23:9 [41] 08


```
1  public class ClassA
2  {
3      void meth1() // main()
4      {
5          System.out.println("hi");
6          ClassA aobj=new ClassA(); // Creating an Object
7          aobj.meth2();
8          System.out.println(10);
9      }
10     void meth2() //meth1()
11     {
12         System.out.println("hello");
13         System.out.println(20);
14     }
15     public static void main(String[] args)
16     {
17         System.out.println("Start");
18         ClassA aobj=new ClassA(); // Creating an Object
19         System.out.println("Java is awesome");
20         aobj.meth1(); // Calling a method
21         System.out.println("End");
22     }
23 }
24
```

Console X
<terminated> ClassA [Java Application] C:\
Start
Java is awesome
hi
hello
20
10
End

```
1  package com.pack1;
2
3  public class ClassA
4  {
5      void meth1()
6      {
7          System.out.println("int value : "+10);
8      }
9      public static void main(String[] args)
10     {
11         ClassA aobj=new ClassA();
12         aobj.meth1();
13     }
14 }
15
16
17
18
19
20
21
22
23
24
```

Console X
<terminated> ClassA [Java Application] C:\Program Files\Java\
int value : 10