WEEK-01

Learn and Understand Basic Concepts of Object oriented programming

Object oriented programming is a programming paradigm based on the concepts of 'objects', which can contain data in the form of fields (attributes or properties) and code in the form of procedures (methods or functions). The key concepts of OOP includes :-

Class :- a blue print for creating objects. it defines the properties and behaviours that objects of the class will have

Object :- An instance of a class. It represents a specific entity in a program

Encapsulation :- the bundling of the data with the methods that operate on that data, or the restriction of direct access to some of the objects components.

Inheritance :- The ability of a class to inherit properties and behaviours from another class

Polymorphism :- the ability to present the same interface for different data types.

Learn and Understand the features of Java

Platform independence :- Java programs can run on any device that has Java Virtual Machine (JVM) installed, regardless of the underlying architecture.

Object oriented :- Java is designed to be object oriented, facilitating modular and reusable code

Simple :- Java syntax is designed to be simple and easy to learn, especially for those familiar with C or C++

Secure :- Java's security feature such as the sandbox environment, helps to protect system from malicious code

Robust :- Java's strong memory management, exception handling, and type safety contribute to its robustness.

multithreaded :- java supports concurrent programming by providing built in support for multithreading

Learn and understand Installation of Java JDK and configure the same

You can download it from http://www.oracle.com/java/technologies/downloads/

Step1 - click on the jdk-8u271-windows64.exe file and click on next button

Step2 - if you want to change the path change it or else let it be default depending on your convienence. And Click on next button.

Step3 - This gives the window for installing wait till it goes to another window

Step4 - if you want to change the JRE path you can change it or let it be default depending on your convenience and click on to Next.

Step5 - wait until it Completes

Step6 - now you have successfully installed JDK software only when you click finish/close button After installing your JDK your cook is not done still lot more things to do.

Setting up your path and classpath for making your programs to run anywhere in the drive

Right click on my computer and click on properties
Click on the advanced tab in the system properties
dialog box. and then click for environment variables
button

If you don't have path in the display then go for
new otherwise go for edit button
In the section you have 2 part one is user variable
and system variables

Because the JDK is installed in this path you
should take the fullpath till bin because bin
contains javac tools finally click on to Ok

Now open a fresh command prompt and give
javac - version and java - version
Now you can see the version displayed now
path is set successfully and you can run anywhere
in your hard disk drives

Learn and Understand Java tokens with Suitable Example.

Java tokens are the basic building blocks of a java program. They include identifiers, keywords literals, operators, separators and comments. Here are some examples:

Identifiers : 'myVariable', 'calculateTotal', 'MAX_VALUE'

Keywords : 'class', 'public', 'static', 'void'

Literals :- '5' (integer literal)
3.14 (floating point literal)
"Hello" (String literal)

Operators : '+' (addition), '-' (subtraction),
'*' multiplication, '/' (division)

Seperators : ';' (semicolon)
'{' (left curly braces) '}' (right curly braces),
',' (comma)

Comments : // This is single line comment ;
'/* This is multi-line comment */'

Learn and understand Installation of Eclipse IDE

1 Go to oracle java download page for Eclipse java download

2 follow the installation instructions and install the Eclipse JDK.

On clicking the download packages, the following page will be displayed with multiple eclipse versions

Select the IDE for Java developers and select the download link from one of the below depending on the operating system of your machine.

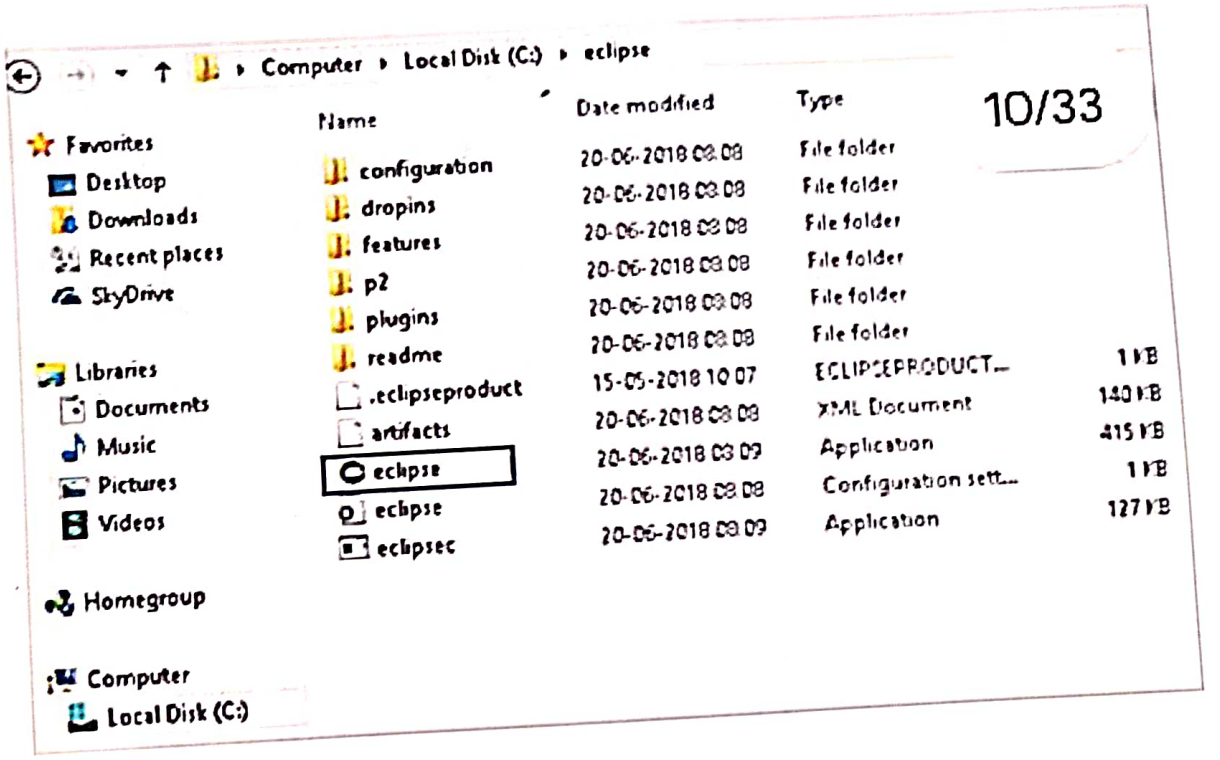
Once you select the OS type download link, you will be redirected to the following page

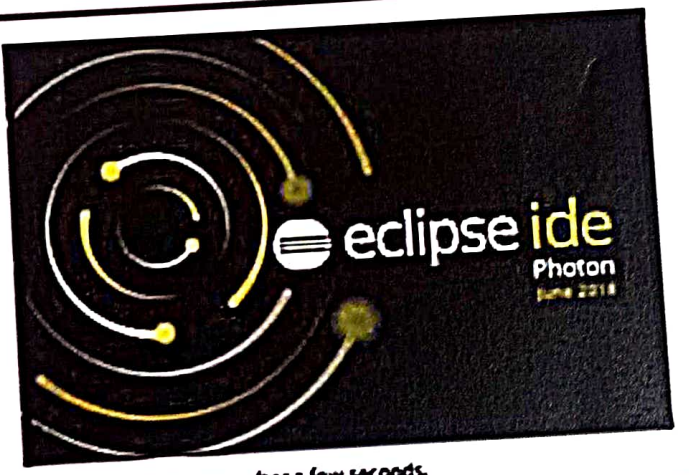
you should see the below folder structure once you are done with the above steps

Setting up the Eclipse workspace

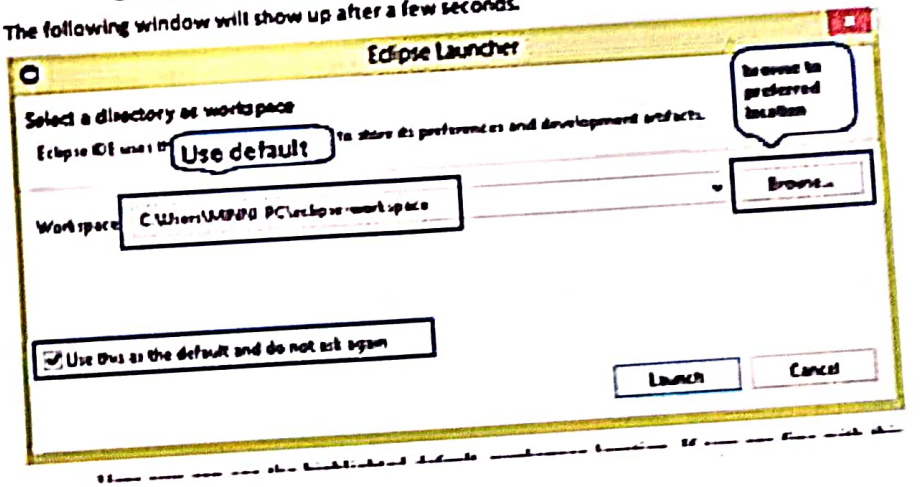
Open the Eclipse workspace once it is launched you can see the default view

Close the window.





The following window will show up after a few seconds.



Here you can see two sub items under your project name. The IDE system library

Learn and understand to design and implement Simple programs

(a) Hello world

```
package Week1;
public class Hello_world
{
    public static void main (Strings [] args)
    {
        System.out.println(" Hello world \n
            welcome to programming with Java);
        System.out.println (" ");
    }
}
```

Output

Hello world...
Welcome to programming with Java

(b) Odd or Even

```
Package Week-1;
Pubic Class Even_or_Odd
{ public static void main (String [] args)
  {
      System.out.println ("the given number
       " + num + " is odd or even");
      int num = 68;
      if (num % 2 == 0)
          System.out.println (" The given
      number " + num + " is even");
      else
          System.out.println (" The given
      number " + num + " is odd ");
  }
}
```

Output

This program checks the number is odd or even
The given number 68 is even

06. Learn and Understand to design and implement simple Programs

a)Hello world

```
package Week1;

public class Hello_World {

    public static void main(String[] args)
    {
        System.out.println("Hello World...\n Welcome to programming with Java");
        System.out.println("");
    }
}
```

Output:-

```
Hello World...
 Welcome to programming with Java
```

b) Odd or Even

```
package Week1;

public class Even_or_Odd {
    public static void main(String[] args) {
        System.out.println("This program checks the no is odd or even");
        int num=68;
        if(num%2==0)
            System.out.println("The given number "+num+" is even");
        else
            System.out.println("The given number "+num+" is odd");
    }

}
```

Output:-

```
This program checks the no is odd or even
The given number 68 is even
```

Main.java

Run

```java
public class Even_or_Odd
{
    public static void main(String[] args)
    {
        System.out.println("This program checks the no is odd or even");
        int num=68;
        if(num%2==0)
            System.out.println("The given number "+num+" is even");
        else
            System.out.println("The given number "+num+" is odd");
    }

}

```

Main.java

```java
// Online Java Compiler
// Use this editor to write, compile and run your Java code online

class Hello_World {
    public static void main(String[] args) {
        System.out.println("Hello, World!");
    }
}
```

```
java -cp /tmp/2c16UFj9ti Even_or_Odd
This program checks the no is odd or even
The given number 68 is even
```

```
java -cp /tmp/2c16UFj9ti HelloWorld
Hello, World!
```