

Lab Task 1: Installation of IDE and write the basic program in JAVA.

OBJECTIVES:

- i. Getting acquainted with JDK and Netbeans IDE
- ii. Writing the first Java program using Netbeans IDE
- iii. To be familiar with the basic concept of class and object

PROBLEM

- a. Install JDK and Netbeans. Create a Java with Maven project. Write your first java program to print 'Hello World'.
- b. Create a student class with properties id, name, email, cgpa, hometown. Create two objects for the class. Finally display the details of those objects using various techniques.

Solution:

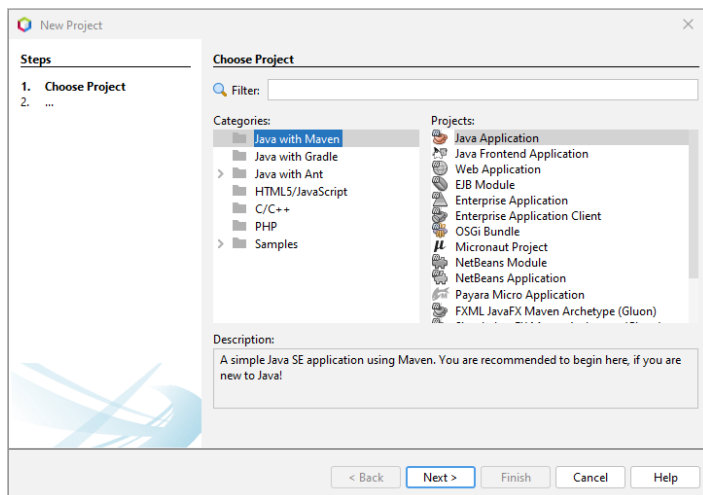
- a. Install JDK and Netbeans using the following video tutorial:

https://www.youtube.com/watch?v=vt7_6HwCFOU

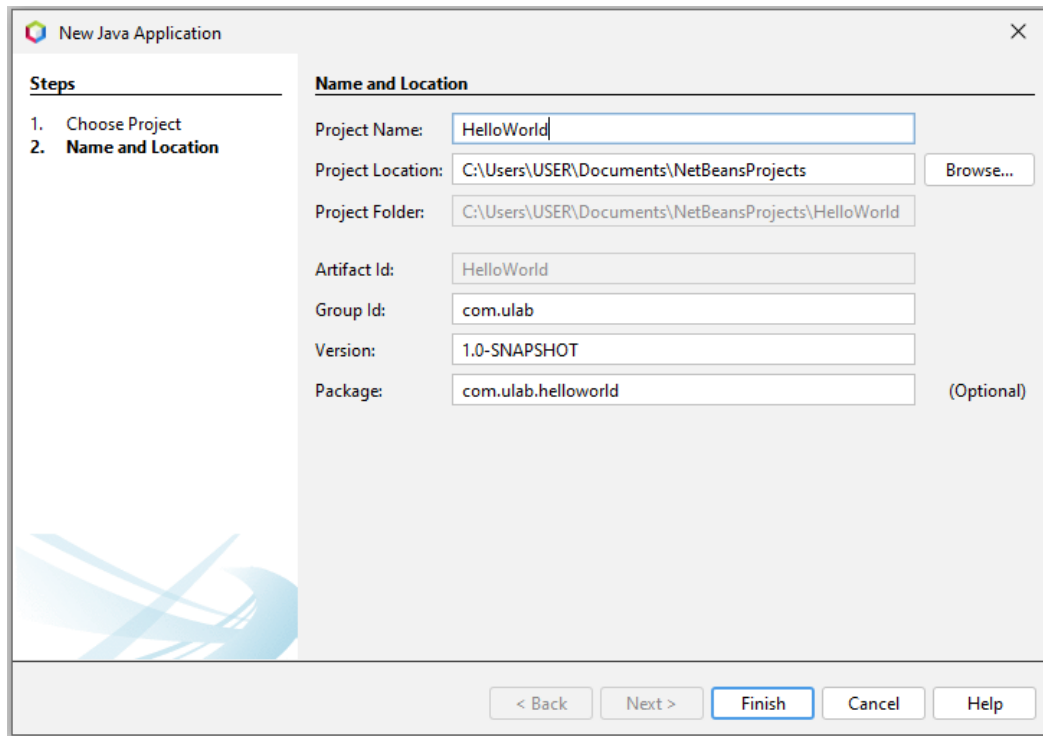
Writing first Java program using netbeans:

Take the steps below to set up a new Java project.

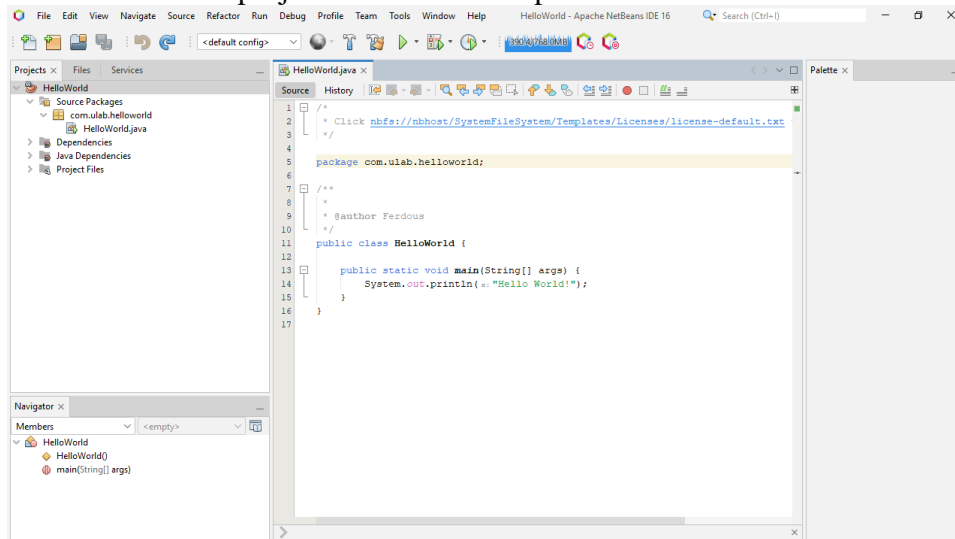
1. In the IDE, choose File > New Project or click the "New Project" button in the toolbar.
2. In the New Project wizard, select Java Application, as shown in the figure below. Then click Next.



3. In the Name and Location page of the wizard, type HelloWorld in the Project Name field, (as shown in the figure below):



4. Click Finish. The project is created and opened.



5. Click the play icon (Run Project). You should have the output window like this.



```
Output - Run (HelloWorld) ×
cd C:\Users\USER\Documents\NetBeansProjects\HelloWorld; "JAVA_HOME=C:\\Program Files\\Java\\jdk-19" cmd /c "%C:\\Program F
Scanning for projects...

-----< com.ulab:HelloWorld >-----
Building HelloWorld 1.0-SNAPSHOT
-----[ jar ]-----

--- maven-resources-plugin:2.6:resources (default-resources) @ HelloWorld ---
Using 'UTF-8' encoding to copy filtered resources.
skip non existing resourceDirectory C:\Users\USER\Documents\NetBeansProjects\HelloWorld\src\main\resources

--- maven-compiler-plugin:3.1:compile (default-compile) @ HelloWorld ---
Changes detected - recompiling the module!
Compiling 1 source file to C:\Users\USER\Documents\NetBeansProjects\HelloWorld\target\classes

--- exec-maven-plugin:3.0.0:exec (default-cli) @ HelloWorld ---
Hello World!

BUILD SUCCESS

Total time: 4.701 s
Finished at: 2023-05-16T10:03:17+06:00
-----
```

- b. Create a student class with properties id, name, email, cgpa, hometown. Create two objects for the class. Finally display the details of those objects using various techniques.

Prerequisites:

- i. Basic knowledge of Java programming language.
- ii. Understanding of classes, objects, and constructors in Java.

Learning outcomes:

- i. Understanding how to create a class with properties and methods in Java.
- ii. Understanding how to create objects and set their properties.
- iii. Understanding how to access and display object properties.

Problem Analysis:

we need to create a Student class with properties such as id, name, email, cgpa, and hometown. Then, we need to create two objects of the Student class and display their details using methods and without using methods.

Background Theory:

Classes in Java:

It is a blueprint or template for creating objects.

- Class is not a real-world entity. It is just a template or blueprint or prototype from which objects are created.
- Class does not occupy memory.
- Class is a group of variables of different data types and a group of methods.

Objects in Java:

It is a basic unit of Object-Oriented Programming and represents real-life entities. An object consists of :

- **State:** It is represented by attributes of an object. It also reflects the properties of an object.
- **Behavior:** It is represented by the methods of an object. It also reflects the response of an object with other objects.
- **Identity:** It gives a unique name to an object and enables one object to interact with other objects.

The problem is to create a Student class in Java with properties such as id, name, email, cgpa, and hometown. Two objects of the class need to be created, initializing the properties one by one using the objects. Finally, the details of those objects should be displayed using various techniques.

Algorithm Design:

1. Create a Java class named "Student" with properties: id (integer), name (string), email (string), cgpa (double), and hometown (string).
2. Write a display method to display the properties for a student object.
3. In the main method:
 - a. Create two Student objects.
 - b. Initialize the properties of the two objects.
4. Display the details of the first object without using the display method.
5. Display the details of the second object using the display method.

Code:

```
package student;
public class Student {
    int id;
    String name;
    String email;
    double cgpa;
    String hometown;

    void display()
    {
        System.out.println("ID: "+id);
        System.out.println("Name: "+name);
        System.out.println("Email: "+email);
        System.out.println("CGPA "+cgpa);
        System.out.println("Home Town: "+hometown);
    }
}
```

Figure 1: Class with properties and method

```

public static void main(String[] args) {
    Student s1=new Student();
    s1.id=101;
    s1.name="Sakib";
    s1.email="sakib@gmail.com";
    s1.cgpa=3.55;
    s1.hometown="Dhaka";

    Student s2=new Student();
    s2.id=102;
    s2.name="Tamim";
    s2.email="tamim@gmail.com";
    s2.cgpa=3.45;
    s2.hometown="Khulna";

    s1.display();

    System.out.println("ID: "+s2.id);
    System.out.println("Name: "+s2.name);
    System.out.println("Email: "+s2.email);
    System.out.println("CGPA "+s2.cgpa);
    System.out.println("Home Town: "+s2.hometown);
}
}

```

Figure 2: Main method

```

ID: 101
Name: Sakib
Email: sakib@gmail.com
CGPA 3.55
Home Town: Dhaka
ID: 102
Name: Tamim
Email: tamim@gmail.com
CGPA 3.45
Home Town: Khulna

```

BUILD SUCCESS

Total time: 2.214 s
 Finished at: 2023-05-27T13:36:59+06:00

Figure 3: Output

Practice Problems

1. Create a class called BankAccount with instance variables accountNumber and balance. Add methods to deposit and withdraw money from the account. Create objects of BankAccount and perform deposit and withdrawal operations.
2. Create a class rectangle with properties such as length and width. Add methods to calculate the perimeter and area of the rectangle. Create objects and display their corresponding perimeter and area.
3. Create a class called movie which as properties such as title, genre, leadactor, director, release year, rating and review. Create two movie objects and display their properties. If the rating is <5 , the review should be “Not Good”. Otherwise, the review would be “Good”.