

Introduction to Basic Java Program

OBJECTIVES:-

- Write, compile and run Java programs.
- Declare and define classes and objects
- Declare fields, methods, and constructors
- Read input from commandline

1. Install and set environment variables

<https://www.youtube.com/watch?v=FVxKbAukRxk>

2. General syntax of java program:

```
access-specifier class ClassName {  
    // fields  
    fieldType fieldName;  
  
    // methods  
    public returnType methodName() {  
        statements;  
    }  
}
```

Program#1

i. Now, write your code in a text-editor (sublime text, notepad++, etc):

```
class First  
{  
    public static void main(String abc[])  
    {  
        System.out.println("My first Java Program");  
    }  
}
```

ii. Save the program in D:\java with the name First(same as the name of class) and Extension (.java) for e.g. First.java

iii. In order to compile and run the program, do the following in the command prompt:

```
C:\Documents and Settings\bidur>d:
D:\>cd java
D:\java>_
```

Now to compile the program use the command `javac` that converts the java source code into binary program consisting of byte codes.

`javac First.java`

If the program contains no errors, the compiler generates a *byte code program* (*.class file*) from your source file. The compiler stores the byte code program in a file with the same name as source file, but with the extension `.class`.

The Java interpreter inspects the byte code and checks it to ensure that the security restrictions are met and then execute the program within the java virtual machine.

`java First`

If there is no exception in the program the output is printed on the command prompt.

```
D:\java>javac First.java
D:\java>java First
My first Java Program
D:\java>
```

`System.out.print` command in java is same as `cout` in C++ . `System.out.println` command is similar to `System.out.print` but also prints a newline as well. Run the following programs and observe the output.

Program#2

WAP to do simple arithmetic operation in main method

```
class CalculateSum
{
    public static void main(String args[])
    {
        int num1,num2,sum;
        num1 = 100;
        num2 = 200;
        sum= num1 + num2;
        System.out.println("Sum of two numbers = "+sum);
    }
}
```

Program#3

WAP to demonstrate Class called Motorbike. Provide its properties and behavior (method) as well.

```
public class Motorbike {

    int speed;
    String model;

    public Motorbike(String model) {
        this.model = model;
    }

    public void accelerate() {
        // add 1 km/hr
        speed = speed + 1;
    }

    public void stop() {
        // set current speed to zero
        speed = 0;
    }

    public void printSpeed() {
        // display the current speed of this car
        System.out.println("Current speed of " + model + " is " + speed + " mph");
    }

    public static void main(String[] args) {
        // create new Honda car
        Motorbike honda = new Motorbike("Honda");
        // create new Pulsar car
        Motorbike pulsar = new Motorbike("Pulsar");
        // print current speed of Honda
        honda.printSpeed();

        // call the accelerate method twice on Honda
        honda.accelerate();
        honda.accelerate();

        // call the accelerate method once on Pulsar
        pulsar.accelerate();

        // print current speed of Honda
        honda.printSpeed();
        // print current speed of Pulsar
        pulsar.printSpeed();

        // now park the Honda car
        honda.stop();
        // print current speed of Honda
        honda.printSpeed();
    }
}
```

Program#4

WAP to read an numbers from console and display their sum

```
import java.io.*;

public class ReadData {

    public int getInteger(){
        System.out.println("Write an Integer number:");
        String line = null;
        int value = 0;
        try {
            BufferedReader is = new BufferedReader(new InputStreamReader(System.in));
            line = is.readLine();
            value = Integer.parseInt(line);
        } catch (Exception e) {
            System.err.println("Unexpected IO ERROR: " + e);
        }

        return value;
    }

    public static void main(String[] args) {

        ReadData rd = new ReadData();

        int n1 = rd.getInteger();
        int n2 = rd.getInteger();
        int sum = n1+n2;

        System.out.println(" Integer: " + sum);
    }
}
```

Program#5

Write a program with a class which saves the data for a bank account

```
import java.util.Scanner;
public class BankAccount
{
    String id;
    float balance;
    int transactionCount;
    String name;

    public BankAccount(String id, float balance, String name )
    {
        this.id = id;
        this.balance = balance;
        this.transactionCount = 0;
        this.name = name;
    }

    public void readAccountDetails()
    {
        Scanner b = new Scanner (System.in);
        System.out.println("Enter name: ");
        name = b.nextLine();
        System.out.println("Enter id: ");
        id = b.nextLine();
        System.out.println("Enter balance: ");
        balance = b.nextFloat();
    }

    String getAccountDetails()
    {
        return("ID: " + id + "Balance: " + balance + "Name: " + name );
    }

    void deposit(float amount)
    {
        balance = balance + amount;
        transactionCount++;
    }

    void withdraw(float amount)
    {
        balance = balance - amount;
        transactionCount++;
    }

    public static void main(String[] args) {
```

```
BankAccount acc1 = new BankAccount("", 1000 , "" );
acc1.readAccountDetails();
System.out.println( "Details \n"+ acc1.getAccountDetails());
acc1.deposit(50);
acc1.getAccountDetails();
acc1.withdraw(25);
acc1.getAccountDetails();
    }
}
```

Program#6

Write a program to count the number of instances of a class using class variable(i.e. static variable).

```
class SampleClass {

    // Set count to zero initially.
    static int count = 1;
    float data;

    public SampleClass() {

        // increment the count on each call to the constructor via any instance
        count = count + 1;

        System.out.print(" number: " + count);
    }
}

public class ObjectCountDemo {

    public static void main(String[] args) {
        SampleClass stuff1 = new SampleClass();
        stuff1.data = SampleClass.count;
        SampleClass stuff2 = new SampleClass();
        stuff2.data = SampleClass.count;
        stuff2.data += stuff1.data;
        System.out.println(" Final Data: " + stuff2.data + " "+ stuff1.data );
    }
}
```

Program#7

Write a program that reads two numbers from the command line, the number of hours worked by an employee and their base pay rate. Add warning messages if the pay rate is less than the minimum wage (12.5 an hour) or if the employee worked more than the number of hours in a week.

Compile: <javac Salary.java>

Run: <java Salary 30 50>

```
class Salary {

    public static void main (String[] args) {

        double hours = Double.valueOf(args[0]).doubleValue();
        double rate = Double.valueOf(args[1]).doubleValue();
        double pay;

        pay = rate * hours;

        System.out.println("The paycheck is " + pay + " dollars.");
        if ( rate < 12.5) {
            System.err.println("This employee is not getting the legally required minimum
wage.");
        }
        if ( hours > 7*24) {
            System.err.println("Did this employee really work " + hours + " hours?");
        }

    }

}
```