



# JAVA Programming

Lab book – Week 2



Create a program that defines a class Student with the data 'name, city, and age,' as well as a method printData to display the data. To declare and access the values, create the objects s1 and s2.

```
public class Student {
String name, city;
int age;
static int m;
void printData() {
    System.out.println("Student name = "+name);
    System.out.println("Student city = "+city);
    System.out.println("Student age = "+age);
}
}
class Stest {
    public static void main(String args[]) {
        Student s1=new Student();
        Student s2=new Student();
        s1.name="Nikith";
        s1.city="Rajahmundry";
        s1.age=22;
        s2.name="Meghana";
        s2.city="Mumbai";
        s2.age=21;
        s2.printData();

        s1.printData();
        s1.m=20;
        s2.m=22;
        Student.m=27;
        System.out.println("s1.m = "+s1.m);
        System.out.println("s2.m = "+s2.m);
        System.out.println("Student.m = "+Student.m);
    }
}
```

Using parameterized constructor with two parameters id and name. While creating the objects obj1 and obj2 passed two arguments so that this constructor gets invoked after creation of obj1 and obj2.



```
class Employee {  
    int empId;  
    String empName;  
    //parameterized constructor with two parameters  
    Employee(int id, String name){  
        this.empId = id;  
        this.empName = name;  
    }  
    void info(){  
        System.out.println("Id: "+empId+" Name: "+empName);  
    }  
    public static void main(String args[]){  
        Employee obj1 = new Employee(10245,"Chaitanya");  
        Employee obj2 = new Employee(92232,"Negan");  
        obj1.info();  
        obj2.info();  
    }  
}
```

**Write a program in JAVA to demonstrate the method and constructor overloading.**

```
class Cs {  
    int p,q;  
    public Cs() {}  
    public Cs(int x, int y) {  
        p=x;  
        q=y;  
    }  
    public int add(int i, int j) {  
        return (i+j);  
    }  
    public int add(int i, int j, int k) {  
        return (i+j+k);  
    }  
    public float add(float f1, float f2) {  
        return (f1+f2);  
    }  
}
```

```
}  
public void printData() {  
    System.out.print("p = "+p);  
    System.out.println(" q = "+q);  
}  
}  
  
class ConstructorOverloading {  
    public static void main(String args[]) {  
        int x=2, y=3, z=4;  
        Cs c=new Cs();  
        Cs c1=new Cs(x, z );  
        c1.printData();  
        float m=7.2F, n=5.2F;  
        int k=c.add(x,y);  
        int t=c.add(x,y,z);  
        float ft=c.add(m, n);  
        System.out.println("k = "+k);  
        System.out.println("t = "+t);  
        System.out.println("ft = "+ft);  
    }  
}
```

**Write a Java Program to finds addition of two matrices**

```
class Add {  
public static void main(String args[])  
{  
    int [][] x={{1,2,3}, {4,5,6}, {7,8,9} };  
    int [][] y={ {11,12,13}, {14,15,16}, {17,18,19} };  
    int [][] z=new int[3][3];  
    for(int i=0; i<3; i++)  
        for(int j=0; j<3; j++) {  
            z[i][j]=x[i][j]+y[i][j];  
        }  
    for(int i=0; i<3; i++)  
    {  
        for(int j=0; j<3; j++) {  
            System.out.print(z[i][j]+" ");  
        }  
    }  
}
```



```
        System.out.print("\n");
    }
}
}
```

**Write a Java Program to find the average of numbers in an array**

```
class Avg {
public static void main(String args[]) {
    int n=args.length;
    float [] x=new float[n];
    for(int i=0; i<n; i++) {
        x[i]=Float.parseFloat( args[i]);
    }
    float sum=0;
    for(int i=0; i<n; i++)
        sum=sum+x[i];
    float avg=sum/n;
    System.out.println("Average of given numbers is
"+avg);
}
}
```

**Write a program in JAVA to create a class Bird also declares the different parameterized constructor to display the name of Birds.**

```
class Bird
{
int age;
String name;
    Bird()
    {
        System.out.println("this is the parrot");
    }
    Bird(String x)
    {
        name=x;
        System.out.println("this is the "+name);
    }
}
```



```
Bird(int y,String z)
{
    age=y;
    name=z;
    System.out.println("this is the
"+age+"years\t"+name);
}
public static void main(String arr[])
{
    Bird a=new Bird();
    // a.Bird();
    Bird b=new Bird("maina");
    Bird c=new Bird(20,"sparrow");
}
}
```

**Write a java Program to generate a Ladder of number**

```
import java.util.Scanner;
class Ladder
{
    public static void main(String arr[])
    {
        Scanner in=new Scanner(System.in);
        System.out.println("Enter the number of rows");
        int a=in.nextInt();
        for(int i=1;i<=a;i++)
        {
            for(int j=1;j<=i;j++)
                System.out.print(j);
            for(int k=i-1;k>=1;k--)
                System.out.print(k);
            System.out.print("\n");
        }
    }
}
```

**Write a program to summation of numbers from 1 to 10.**



```
// Java program to illustrate while loop

class whileLoopDemo {
    public static void main(String args[])
    {
        int x = 1, sum = 0;

        // Exit when x becomes greater than 4
        while (x <= 10) {
            // summing up x
            sum = sum + x;

            // Increment the value of x for
            // next iteration
            x++;
        }
        System.out.println("Summation: " + sum);
    }
}
```

File *Salary.java* contains most of a program that takes as input an employee's salary and a rating of the employee's performance and computes the raise for the employee. As in the pre-lab, an employee who is rated excellent will receive a 6% raise, one rated good will receive a 4% raise, and one rated poor will receive a 1.5% raise. Add the *if... else...* statements to program *Salary* to make it run as described above.

```
import java.util.Scanner;
import java.text.NumberFormat;
public class Salary
{
    public static void main (String[] args)
    {
        double currentSalary; // employee's current salary
        double raise; // amount of the raise
        double newSalary; // new salary for the employee
        String rating; // performance rating
        Scanner scan = new Scanner(System.in);
```



```
System.out.print ("Enter the current salary: ");
currentSalary = scan.nextDouble();
System.out.print ("Enter the performance rating
(Excellent, Good, or Poor): ");
rating = scan.nextLine();
// Compute the raise using if ...
newSalary = currentSalary + raise;
// Print the results
NumberFormat money =
NumberFormat.getCurrencyInstance();
System.out.println();
System.out.println("Current Salary: " +
money.format(currentSalary));
System.out.println("Amount of your raise: " +
money.format(raise));
System.out.println("Your new salary: " +
money.format(newSalary));
System.out.println();
}
}
```

Program *Rock.java* contains a skeleton for the game Rock, Paper, Scissors. Open it and save it to your directory. Add statements to the program as indicated by the comments so that the program asks the user to enter a play, generates a random play for the computer, compares them and announces the winner (and why).

```
import java.util.Random;
public class Rock
{
    public static void main(String[] args)
    {
        String personPlay; //User's play -- "R", "P", or "S"
        String computerPlay; //Computer's play -- "R", "P",
or "S"
        int computerInt; //Randomly generated number used to
determine
        //computer's play
        Scanner scan = new Scanner(System.in);
```





```
Random generator = new Random();
//Get player's play -- note that this is stored as a
string
//Make player's play uppercase for ease of comparison
//Generate computer's play (0,1,2)
//Translate computer's randomly generated play to string
    switch (computerInt)
    {
    }
//Print computer's play
//See who won. Use nested ifs instead of &&.
    if (personPlay.equals(computerPlay))
        System.out.println("It's a tie!");
    else if (personPlay.equals("R"))
        if (computerPlay.equals("S"))
            System.out.println("Rock crushes
scissors. You win!!");
        else {
            // ... Fill in rest of code
        }
    }
}
```

The file *Grades.java* contains a program that reads in a sequence of student grades and computes the average grade, the number of students who pass (a grade of at least 60) and the number who fail. Compile and run the program to see how it works.

```
//
*****
*
// Grades.java
//
// Read in a sequence of grades and compute the average
// grade, the number of passing grades (at least 60)
// and the number of failing grades.
//
*****
*
import java.util.Scanner;
public class Grades
```



```
{
//-----
-----
// Reads in and processes grades until a negative number is
entered.
//-----
-----
public static void main (String[] args)
{
double grade; // a student's grade
double sumOfGrades; // a running total of the student grades
int numStudents; // a count of the students
int numPass; // a count of the number who pass
int numFail; // a count of the number who fail
Scanner scan = new Scanner(System.in);
System.out.println ("\nGrade Processing Program\n");
// Initialize summing and counting variables
sumOfGrades = 0;
numStudents = 0;
numPass = 0;
numFail = 0;
// Read in the first grade
System.out.print ("Enter the first student's grade: ");
grade = scan.nextDouble();
while (grade >= 0)
{
sumOfGrades = sumOfGrades + grade;
numStudents = numStudents + 1;
if (grade < 60)
numFail = numFail + 1;
else
numPass = numPass + 1;
}
// Read the next grade
System.out.print ("Enter the next grade (a negative to quit):
");
grade = scan.nextDouble();
}
if (numStudents> 0)
{
System.out.println ("\nGrade Summary: ");
System.out.println ("Class Average: " +
sumOfGrades/numStudents);
System.out.println ("Number of Passing Grades: " + numPass);
System.out.println ("Number of Failing Grades: " + numFail);
}
else
System.out.println ("No grades processed.");
}
```

<pre>} }</pre>

## PART 2

### Exercise 1

**Introduction of Java** (For multiple choice questions choose more than one option if required)

1. Java's defining characteristics come from?
  - a. It's a newly created language
  - b. It's predecessors
  - c. None of the above
  - d. a&b
2. Java derives it's syntax from ..... and many of Java's object-oriented features were influenced by.....?
3. Java?
  - a. Platform-dependent
  - b. Platform-independent



### Oops in Java

8. Which type of programming language is Java?
  - a. Process-oriented model
  - b. Object-oriented programming
9. Write a simple Java program to print the statement "This is My First Java program"?
10. What is the return type of program's main() method?
11. What is the argument type of program's main() method?
12. Which of the following signatures are valid for the main() method entry point of an application?
  - a. public static void main()
  - b. public static void main(String arg[])
  - c. public void main(String [] arg)
  - d. public static void main(Stering[] arg)
  - e. public static int main(String [] arg)



## Lab 2

### Java keywords

13. Name two keywords in Java that are reserved but not used?
14. Choose the valid keywords from those listed below?

- a. bytes
- b. byte
- c. import
- d. finalist

### Operators

15. What are the operators used in Java?
16. Write a program to print the remainder of division operation (22/10 & 22.25/10)?
17. Write a program to demonstrate several assignment operators (+=, -=, \*=, /=)?

18. `int a = 1;`

`int b = 2;`

`int c;`

`int d;`

`c = ++b;`

`d = a++;`

`c++;`

What is the value of c and d after executing this code?

19. `a = 4; b = 1; Boolean c = a < b;` Value of c?

- a. true
- b. false
- c. 1
- d. 4



20. Choose valid statements in java?

- a. if(!done)
- b. if(done)
- c. if(done == 0)
- d. if(done != 0)

21. a = true, b = false

```
c = a | b;  
d = a & b;  
e = a ^ b;  
f = (! a & b) | (a & ! b);  
g = ! a;
```

What is the value of c, d, e, f, and g?

22. a = 20; b = 10;

```
x = a > b ? a : b;
```

What is the value of x?

- a. 10
- b. 20
- c. 30



23. Write Java assignment statements to evaluate the following equations.

- a.  $\text{Area} = \pi r^2 + 2\pi rh$
- b.  $\text{Energy} = \text{mass} (\text{acceleration} * \text{height} + \text{velocity}^2 / 2)$

24. Write a Java program to find area and perimeter of a square and rectangle?

25. Write a program to find out biggest number out of three numbers using relational and short-circuit OR and AND operators?

26. Write a program to swap of two integers without using temporary variable?

## Lab 3

### Control Statements

27. What are the three categories of control statements?
28. Write a program that uses an if-else-if ladder to determine which season a particular month is in?
29. Write a program that uses a switch case to determine given character is vowel or not?
30. Write a program that uses a switch case to determine which season a particular month is in?
31. Write a program for four-function calculator using switch case?(add,sub,mul,div)

## Lab 4

### Using while loop

32. Write a program to determine given number is prime or not?  
(Hint -Prime number: A number only divided by itself and 1)
33. Write a program to find even number up to range?
34. Write a program to find average of odd number below range?
35. Write a program to find midpoint of 100 and 200?
36. Write a program for palindrome using while loop? (write for both string and numbers)  
(Hint: Given number are 125 to print 521  
125 is not a palindrome) or (liril reverse is lilri)
37. Write a program to find sum of digits of a number.  
(Hint :  $12345 = 1+2+3+4+5 = 15$ )
38. Write a program for Fibonacci series  
(Hint :Give number is 6. To print this format 0 1 1 2 3 5)
39. Write multiplication tables using do-while loop?
40. Write a program to determine given number is prime or not using for loop?
41. Write a program to print the format below using nested for loop

1	0
2 3	0 0



4 5 6                      0 0 0  
7 8 9 10      and            0 0 0

## Lab 5

### Array

42. Write a program to print array elements in reverse order.
43. Write a program to find Biggest and Lowest element of an array.
44. Write a program to swap two arrays without using temporary variable.

## Lab 6

### Class and methods

45. Design a class to represent a bank account including the following members.

#### Data members

Name of the depositor

Account number

Type of account

Balance amount in the account

#### Methods

To assign a initial values

To deposit an amount after checking balance

To display the name and balance

46. Modify the above program (55) to incorporate a constructor to provide initial value and using *this* keyword.
47. Write a program for arithmetic operation using method overloading?
48. Write the above (57) program using overloading constructors?
49. Write a simple program find the volume of Box using method calls?
50. Write Example program – Inheritance and multilevel inheritance?

51. Write a program to print the integer value using method overriding and super keyword?
52. Design a package to display the name and account balance using array of object?
53. Write an above program (62) using import package?
  
54. Write a simple interface program to print an integer value using access implementation through interface references?

