**Rohitkaran Adusmalli**

**JAVA**

**Classes and Methods**

**Lab Exercise No:**14

**Exercise Objective(s):***Construction of an object*

**Exercise:***Create a class called Calculator which has 4 different methods add, diff, mul and div which*

*accepts two numbers as parameters. Create an object to access these methods and invoke*

*these methods with two numbers and display the result in the corresponding methods.*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to a class with 4 methods and access those through an object.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class Calculator{

//Method to add

public void add(int num1,int num2)

{

int num3=num1+num2;

System.out.println("Addition is=" + num3);

}

//Method to subtract

public void sub(int num1,int num2)

{

int num3=num1-num2;

System.out.println("Subtraction is="+ num3);

}

//Method to Multiply

public void mul(int num1,int num2)

{

int num3=num1\*num2;

System.out.println("Multiplication is="+ num3);

}

//Method to Divide

public void div(int num1,int num2)

{

int num3=num1/num2;

System.out.println("Division is="+ num3);

}

public static void main(String args[])

{

//Creating an object of class calculator

Calculator ob=new Calculator();

ob.add(4,2);

ob.sub(4,2);

ob.mul(4,2);

ob.div(4,2);

}

}

**Lab Exercise No:**15

**Exercise Objective(s):***Construction of an object, constructors*

**Exercise:***Create a class called Sample. Write a program to display the no of objects created for that*

*class or the no of times that class is instantiated.*

**Recommended duration:***15Mins*

**Solution :**

/\*\*

\* This is a program to display the no of objects created for that

\* class or the no of times that class is instantiated.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class Sample {

public static int count;

//Incrementing the no. of object

Sample(){

count++;

}

public static void main(String[] args) {

// TODO Auto-generated method stub

Sample ob1=new Sample();

Sample ob2=new Sample();

Sample ob3=new Sample();

//Priting the count of objects

System.out.println(ob3.count);

}

}

**Lab Exercise No:**16

**Exercise Objective(s):***Construction of an object, this keyword, accessors(getters) and mutators(setters),*

*public and private access specifiers, instance and class member variables*

**Exercise:***Create a class called Student with the following details: RollNo, StudName, MarksInEng,*

*MarksInMaths and MarksInScience. Write getters and setters for the all variables. RollNo*

*should be automatically generatedwhenever a newstudent is added. Create a class called Standard with 8 students’ details and write separate method for each of the following tasks and invoke the same.*

1. *To display the entire roll no and the name of the students in the class in the ascending order of roll no.*
2. *To display the roll no and the name of the student who has got the highest percentage.*
3. *To display the roll no and the name of the student who scored highest mark*

*inmathematics.*

1. *To display the roll no and the name of the student in the ascending order of the total marks in mathematics and science alone.*
2. *To display the roll no, name, total marks, percentage and rank of all the students in the descending order of rank.*

**Recommended duration:***40Mins*

**Solution :**

/\*\*

\* This is a program to understand access specifiers, getters and setters,instances.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class Student {

private static int temp=0;

private int rollNo;

private String studName = new String();

private int marksInEng;

private int marksInMaths;

private int marksInScience;

public Student() {

//To alot the roll number of newly generated students

temp++;

this.rollNo = temp;

}

public int getRollNo() {

return rollNo;

}

public void setRollNo(int rollNo) {

this.rollNo = rollNo;

}

public String getStudName() {

return studName;

}

public void setStudName(String studName) {

this.studName = studName;

}

public int getMarksInEng() {

return marksInEng;

}

public void setMarksInEng(int marksInEng) {

this.marksInEng = marksInEng;

}

public int getMarksInMaths() {

return marksInMaths;

}

public void setMarksInMaths(int marksInMaths) {

this.marksInMaths = marksInMaths;

}

public int getMarksInScience() {

return marksInScience;

}

public void setMarksInScience(int marksInScience) {

this.marksInScience = marksInScience;

}

public class Standard {

public void Standard() {

}

//Function for task 1 given

public void task1(Student[] s){

System.out.println("The details of students is as follows :");

for(int i=0;i<s.length;i++) {

System.out.println(s[i].getRollNo() + " | " + s[i].getStudName());

}

}

}

public static void main(String[] args) {

// TODO Auto-generated method stub

//Creating a data of 8 students

Student[] s = new Student[8];

Student s1 = new Student();

s1.setStudName("Rohitkaran");

s1.setMarksInEng(99);

s1.setMarksInMaths(95);

s1.setMarksInScience(94);

s[0] = s1;

Student s2 = new Student();

s2.setStudName("Swathy");

s2.setMarksInEng(92);

s2.setMarksInMaths(96);

s2.setMarksInScience(84);

s[1] = s2;

Student s3 = new Student();

s3.setStudName("Rohiet");

s3.setMarksInEng(91);

s3.setMarksInMaths(90);

s3.setMarksInScience(94);

s[2] = s3;

Student s4 = new Student();

s4.setStudName("Ajinkya");

s4.setMarksInEng(88);

s4.setMarksInMaths(98);

s4.setMarksInScience(94);

s[3] = s4;

Student s5 = new Student();

s5.setStudName("Shraddha");

s5.setMarksInEng(98);

s5.setMarksInMaths(91);

s5.setMarksInScience(94);

s[4] = s5;

Student s6 = new Student();

s6.setStudName("Mrunal");

s6.setMarksInEng(99);

s6.setMarksInMaths(97);

s6.setMarksInScience(94);

s[5] = s6;

Student s7 = new Student();

s7.setStudName("Siddhant");

s7.setMarksInEng(98);

s7.setMarksInMaths(95);

s7.setMarksInScience(99);

s[6] = s7;

Student s8 = new Student();

s8.setStudName("Shreyash");

s8.setMarksInEng(98);

s8.setMarksInMaths(100);

s8.setMarksInScience(84);

s[7] = s8;

Standard obj = new Standard();

obj.task1(s);

}

}

**Lab Exercise No:**17

**Exercise Objective(s):***String class, String immutability*

**Exercise:***Write class that declares the following String.*

***“The quick brown fox jumps over the lazy dog”.***

*Perform the following modifications to the above string using appropriate methods.*

1. *Print the character at the 12th index.*
2. *Check whether the String contains the word “is”.*
3. *Add the string “and killed it” to the existing string.*
4. *Check whether the String ends with the word “dogs”.*
5. *Check whether the String is equal to “The quick brown Fox jumps over the lazy Dog”.*
6. *Check whether the String is equal to “*THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG*”.*
7. *Find the index position of the character “a”.*
8. *Find the last index position of the character “e”.*
9. *Find the length of the String.*
10. *Check whether the String matches to “The quick brown Fox jumps over the lazy Dog”.*
11. *Replace the word “The” with the word “A”.*
12. *Split the above string into two such that two animal names do not come together.*
13. *Print the animal names alone separately from the above string.*
14. *Print the above string in completely lower case.*
15. *Print the above string in completely upper case.*

**Recommended duration:***30Mins*

**Solution :**

/\*\*

\* This is a program to manipulate a given string and print it in different ways.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class SeventeenSol {

public static void main(String[] args) {

// TODO Auto-generated method stub

String s="The quick brown fox jumps over the lazy dog";

//Printing the character at 12th position

System.out.println("The character at 12th index "+ s.charAt(11));

//Checking if 'is' is present in s

String word="";

boolean flag=false;

for(int i=0;i<s.length();i++)

{

char ch=s.charAt(i);

if(Character.isLetter(ch))

{

word=word+ch;

}

else

{

if(word.equals("is"))

{

System.out.println("The string contains 'is'");

flag=true;

break;

}

else

{

word="";

}

}

}

if (flag==false)

System.out.println("The string does not contain 'is'");

//Adding "and killed it" to the original string

String a=s+" and killed it";

System.out.println("After appending: "+a);

//To check if string ends with dog

word="";

for(int i=s.length()-1;i<s.length();i--)

{

char ch=s.charAt(i);

if(Character.isLetter(ch))

{

word=ch+word;

}

else

{

if(word.equals("dog"))

{

System.out.println("Ends with 'dog'");

break;

}

else

{

System.out.println("Does not end with 'dog'");

}

}

}

// Checking whether the String is equal to “The quick brown Fox jumps over the lazy Dog”

if(s.compareTo("The quick brown Fox jumps over the lazy Dog")==0)

System.out.println("Equal");

else

System.out.println("Not Equal");

//Checking whether the String is equal to “THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG”

if(s.compareTo("THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG")==0)

System.out.println("Equal");

else

System.out.println("Not Equal");

//Finding the index position of the character a

System.out.println("The index position of character 'a' is :"+s.indexOf("a"));

//Finding the last index position of the character e

System.out.println("The last index position of character 'e' is :"+s.lastIndexOf("e"));

//Finding the length of the String.

System.out.println("The length of the string is :" + s.length());

//Checking whether the String matches to “The quick brown Fox jumps over the lazy Dog”

if(s.compareTo("The quick brown Fox jumps over the lazy Dog")==0)

System.out.println("Equal");

else

System.out.println("Not Equal");

//Replacing the word “The” with the word “A”

word="";

String t="";

for(int i=0;i<s.length();i++)

{

char ch=s.charAt(i);

if(Character.isLetter(ch))

{

word=word+ch;

}

else

{

if(word.equals("The")||word.equals("the"))

{

t=t+" A";

word="";

}

else

{

t=t+" "+word;

word="";

}

}

}

System.out.println("After replacing :"+ t);

//Printing the animal names alone separately from the above string

t=s+" ";

word="";

System.out.println("The animals names are :");

for(int i=0;i<t.length();i++)

{

char ch=t.charAt(i);

if(Character.isLetter(ch))

{

word=word+ch;

}

else

{

if(word.equals("fox")||word.equals("dog"))

{

System.out.println(word);

word="";

}

else

{

word="";

}

}

}

//Printing the above string in completely lower case

System.out.println("Given string in lowercase : "+s.toLowerCase());

//Printing the above string in completely upper case

System.out.println("Given string in uppercase: "+s.toUpperCase());

}

}

**Lab Exercise No:**18

**Exercise Objective(s):***String class, String immutability*

**Exercise:***Write a program to demonstrate the difference between equals and == operator with*

*appropriate example.*

**Recommended duration:***15Mins*

**Solution :**

/\*\*

\* This is a program to to demonstrate the difference between equals and == operator with

\* appropriate example.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class EighteenSol {

public static void main(String[] args) {

// TODO Auto-generated method stub

//Using String object

String s1 = new String("HELLO");

String s2 = new String("HELLO");

if(s1==s2) {

System.out.println("False");

}else if(s1.equals(s2)) {

System.out.println("True");

}

//Assigning the value at the beginning

String s3 = "World";

String s4 = "World";

if(s3==s4) {

System.out.println("False");

}

}

}

**Lab Exercise No:**19

**Exercise Objective(s):***Arrays*

**Exercise:***Write a program to declare an array with 8 elements and copy the 8 elements into another*

*array and display the same.*

**Recommended duration:***15Mins*

**Solution :**

/\*\*

\* This is a program to declare an array with 8 elements and copy the 8 elements into another

\* array and display the same.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class NineteenSol {

public static void main(String[] args) {

// TODO Auto-generated method stub

int a[] = {15, 23, 90, 123};

//Creating an array of the same size as given

int b[] = new int[a.length];

// Copying elements from array A to B

for (int i=0; i<a.length; i++)

b[i] = a[i];

//Printing both the arrays

System.out.println("Array a : ");

for (int i=0; i<a.length; i++)

System.out.print(a[i] + " ");

System.out.println("");

System.out.println("Array b : ");

for (int i=0; i<b.length; i++)

System.out.print(b[i] + " ");

}

}

**Lab Exercise No:**20

**Exercise Objective(s):***Arrays*

**Exercise:***Write a program to display the sum and the average of elements in the array.*

**Recommended duration:***15Mins*

**Solution :**

/\*\*

\* This is a program to display the sum and the average of elements in the array.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class TwentySol {

public static void main(String[] args) {

// TODO Auto-generated method stub

int arr[] = {20,30,25,75};

double sum=0;

double average=0.0;

//Finding the sum of array elements

for(int i=0;i<arr.length;i++){

sum+=arr[i];

}

System.out.println("The sum of array elements is : " + sum);

//Finding the average of array elements

average=sum / arr.length;

System.out.println("The average of array elements is : " + average);

}

}

**Lab Exercise No:**21

**Exercise Objective(s):***Arrays*

**Exercise:***Write a program to construct two matrices and display the sum of those.*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to construct two matrices and display the sum of those.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class TwentyOneSol {

public static void main(String[] args) {

// TODO Auto-generated method stub

int matrixOne[][]= {{1,2,2},{2,1,2},{1,0,1}};

int matrixTwo[][]= {{2,5,6},{10,21,22},{7,8,13}};

int result[][] = new int[matrixOne.length][matrixTwo.length];

System.out.println("The result of addition of two matrices is : ");

//Adding the two matrices

//And displaying the result

for(int i=0;i<matrixOne.length;i++) {

for(int j=0;j<matrixTwo.length;j++) {

result[i][j] = matrixOne[i][j] + matrixTwo[i][j];

System.out.print(result[i][j] + " ");

}

System.out.println("");

}

}

}

**Lab Exercise No:**22

**Exercise Objective(s):***Arrays*

**Exercise:***Write a program to display the square of the elements of a two dimensional array.*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to display the square of the elements of a two dimensional array.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

import java.util.Scanner;

public class TwentyTwo {

public static void main(String[] args) {

// TODO Auto-generated method stub

System.out.println("Enter the no. of rows and columns : ");

Scanner scan = new Scanner(System.in);

int row = scan.nextInt();

int col = scan.nextInt();

int arr[][] = new int[row][col];

int square[][] = new int[row][col];

System.out.println("Enter the value of the array : ");

//Taking input from user and squaring right away

for(int i=0;i<row;i++) {

for(int j=0;j<col;j++) {

arr[i][j] = scan.nextInt();

square[i][j] = arr[i][j] \* arr[i][j];

}

}

//Displaying the square matrix

System.out.println("The final squared matrix is : ");

for(int i=0;i<row;i++) {

for(int j=0;j<col;j++) {

System.out.print(square[i][j] + " ");

}

System.out.println();

}

}

}

**Lab Exercise No:**23

**Exercise Objective(s):***Arrays*

**Exercise:***Write a program to construct an array with 10 elements and to find the number of*

*occurrences of each element in the Array.*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to count the occurrences of the array elements

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class TwentyThree {

public static void countFreq(int arr[], int n)

{

boolean visited[] = new boolean[n];

// Traversing through array elements and counting the no. frequencies

System.out.println("The frequency of each element is as follows :");

for (int i = 0; i < n; i++) {

// Skip this element if already passed

if (visited[i] == true)

continue;

// Counting the no. of occurrences

int count = 1;

for (int j = i + 1; j < n; j++) {

if (arr[i] == arr[j]) {

visited[j] = true;

count++;

}

}

System.out.println(arr[i] + " " + count);

}

}

public static void main(String[] args) {

// TODO Auto-generated method stub

int arr[] = { 10, 20, 20, 10, 10, 20, 5, 20 ,5 , 30};

int n = arr.length;

countFreq(arr, n);

}

}

**Lab Exercise No:**24

**Exercise Objective(s):***Overloading*

**Exercise:***Create a class called shape with the following methods*

1. *area*
2. *perimeter*

*Overload the area and perimeter method to calculate for both square and rectangle.*

*Create a main class and invoke the area method to calculate the area of the square and*

*rectangle. Also invoke the perimeter method to calculate the perimeter of the square*

*and rectangle.*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to understand overloading.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class Shape {

public void area(int side)

{

int area=side\*side;

System.out.println("The area of square is : "+area);

}

public void area(int length,int breadth)

{

int area=length\*breadth;

System.out.println("The area of rectangle is :"+area);

}

public void perimeter(int side)

{

int perimeter=4\*side;

System.out.println("The perimeter of square is :"+perimeter);

}

public void perimeter(int length,int breadth)

{

int perimeter=2\*(length+breadth);

System.out.println("The perimeter of rectangle is :"+perimeter);

}

public static void main(String[] args) {

// TODO Auto-generated method stub

int l=5,b=10;

int s=8;

Shape obj = new Shape();

obj.area(s);

obj.area(l,b);

obj.perimeter(s);

obj.perimeter(l,b);

}

}

**Lab Exercise No:**25

**Exercise Objective(s):***Overloading*

**Exercise:** *Create a class called employee with the following data members*

1. *empName*
2. *empId*
3. *empAge*
4. *empdesgn*
5. *empLocation*
6. *empExpInYrs*

*All these data members should be initialized using constructors. Use constructor overloading*

*and demonstrate by creating different employee objects with*

1. *Employee name alone*
2. *Employee name and id*
3. *Employee name, id and age*
4. *Employee name, id and designation*
5. *Employee name, id, age and designation*
6. *Employee name, id, age and location*
7. *Employee name, id, age and experience*
8. *Employee name, id, designation and experience*
9. *Employee name, id, designation, location and experience*
10. *Employee name, id, age, designation, location and experience*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to understand overloading in java.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class Employee {

String empName;

private int empId;

private int empAge;

private String empDesgn;

private String empLocation;

private int empExpInYrs;

public Employee(String empName) {

this.empName=empName;

empId=12;

empAge=21;

empDesgn="Trainee Software Engineer";

empLocation="Pune";

empExpInYrs=0;

}

public Employee(String empName,int empID) {

this.empName=empName;

this.empId=empID;

empAge=21;

empDesgn="Trainee Software Engineer";

empLocation="Pune";

empExpInYrs=0;

}

public Employee(String empName,int empID, int empAge) {

this.empName=empName;

this.empId=empID;

this.empAge=empAge;

empDesgn="Trainee Software Engineer";

empLocation="Pune";

empExpInYrs=0;

}

public Employee(String empName,int empID, int empAge, String empDesgn) {

this.empName=empName;

this.empId=empID;

this.empAge=empAge;

this.empDesgn=empDesgn;

empLocation="Pune";

empExpInYrs=0;

}

public Employee(String empName,int empID, int empAge, String empDesgn, String empLocation) {

this.empName=empName;

this.empId=empID;

this.empAge=empAge;

this.empDesgn=empDesgn;

this.empLocation=empLocation;

empExpInYrs=0;

}

public Employee() {

// TODO Auto-generated constructor stub

}

public String getEmpName() {

return empName;

}

public void setEmpName(String empName) {

this.empName = empName;

}

public int getEmpId() {

return empId;

}

public void setEmpId(int empId) {

this.empId = empId;

}

public int getEmpAge() {

return empAge;

}

public void setEmpAge(int empAge) {

this.empAge = empAge;

}

public String getEmpDesgn() {

return empDesgn;

}

public void setEmpDesgn(String empDesgn) {

this.empDesgn = empDesgn;

}

public String getEmpLocation() {

return empLocation;

}

public void setEmpLocation(String empLocation) {

this.empLocation = empLocation;

}

public int getEmpExpInYrs() {

return empExpInYrs;

}

public void setEmpExpInYrs(int empExpInYrs) {

this.empExpInYrs = empExpInYrs;

}

public static void main(String[] args) {

// TODO Auto-generated method stub

//Calling various parameterized constructors with different inputs

Employee obj = new Employee();

obj.Employee("Rohitkaran");

obj.employee("Rohitkaran",1);

obj.employee("Rohitkaran",12,21);

obj.employee("Rohitkaran",12,21,"TSE");

obj.employee("Rohitkaran",12,21,"TSE","pune" );

obj.employee("Rohitkaran",12,21,"TSE","Pune",1);

}

}

**Lab Exercise No:**26

**Exercise Objective(s):***Overloading*

**Exercise:***Create a class called Calculator which has 4 different methods add, diff, mul and div which*

*accepts two numbers as parameters. Overload the methods such that the parameters can be*

*of the following pattern.*

1. *Both are of int data type.*
2. *Both are of double data type.*
3. *First parameter is of int data type and second parameter is of double data type.*
4. *First parameter is of double data type and second parameter is of int data type.*

*Create object to access these methods and invoke these methods with different type of*

*numbers and display the result in the corresponding methods.*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to a class with 4 methods and overloaded methods with different inputs.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class Calculator{

//Method to add

public void add(int num1,int num2)

{

int num3=num1+num2;

System.out.println("Addition is=" + num3);

}

//Method to subtract

public void sub(int num1,int num2)

{

int num3=num1-num2;

System.out.println("Subtraction is="+ num3);

}

//Method to Multiply

public void mul(int num1,int num2)

{

int num3=num1\*num2;

System.out.println("Multiplication is="+ num3);

}

//Method to Divide

public void div(int num1,int num2)

{

int num3=num1/num2;

System.out.println("Division is="+ num3);

}

//Both inputs are double

public void add(double num1,double num2)

{

double num3=num1+num2;

System.out.println("Addition is=" + num3);

}

//One input is integer and other is double

public void sub(double num1,int num2)

{

double num3=num1-num2;

System.out.println("Subtraction is="+ num3);

}

//One input is double while the other one is integer

public void mul(int num1,double num2)

{

double num3=num1\*num2;

System.out.println("Multiplication is="+ num3);

}

public static void main(String args[])

{

//Creating an object of class calculator

Calculator ob=new Calculator();

ob.add(4,2);

ob.add(20.5, 45.6);

ob.sub(4,2);

ob.sub(20.6,15);

ob.mul(4,2);

ob.mul(120, 11.2);

ob.div(4,2);

}

}

**Lab Exercise No:**27

**Exercise Objective(s):***Initializers*

**Exercise:***Write a class called Computer such that the object of that class should be created only when*

*the class is loaded.*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to use initializers.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class Computer {

public static int a;

{

System.out.println("Getting Initialized");

System.out.println("It runs here before before class is loaded.");

}

Computer(){

this.a = 10;

}

public static void main(String[] args) {

// TODO Auto-generated method stub

Computer obj = new Computer();

System.out.println(obj.a);

}

}

**Lab Exercise No:**28

**Exercise Objective(s):***Var-args*

**Exercise:***In the calculator (Lab exercise - 14) program, make the add and diff method to accept var-args*

*and demonstrate.*

**Recommended duration:***20Mins*

**Solution :**

/\*\*

\* This is a program to use varargs.

\*

\* @author Rohitkaran

\*/

package com.hsbc.day2;

public class TwentyEightSol {

//Defining a varargs method

public static void varagrs(String str, int ...a)

{

System.out.println("String: " + str);

System.out.println("Number of arguments is: "+ a.length);

for (int i =0;i<a.length;i++ )

System.out.println(a[i] + " ");

}

public static void main(String[] args) {

// TODO Auto-generated method stub

varagrs("Rohitkaran", 90, 10);

varagrs("Vishal", 80, 70);

}

}