JAVA PROGRAMS

Q1. Write a Program to Add, Subtract , Multiple and Divide two numbers

Q2. Write a Program to Area and Perimeter of Rectangle

Q3. Write a Program to exchange/swap two number values

E.g  a  = 100 , b = 200

Output Will be a = 200, b = 100

Hint : Use third variable to swap two numbers

Q4. Calculate HRA , DA and TA on Salary

Suppose Salary is USD 9000.

Now HRA will be 30% of Salary, DA would be 20% of Salary and TA would be 10% of the Salary

And the Gross Salary would be Salary + hra + da + ta

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q1. WAP to take Student 3 Subject marks

a)      English , Maths , Science

b)      Now Calculate the total marks of a student

c)       And Percentage of a student

d)      Print Each Subject Marks, Total Marks and Percentage

e)      Print Grade of a Student, if Student get more than 50% Marks (Pass)

Otherwise Fail

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q1. WAP to Count even and odd numbers between 1 to 100

Hint : Use Modulus Operator

Q2. WAP to Print sum of 1 to n numbers , n is input by user

Example : If user enter 4 in n, result would be

1+ 2+ 3 + 4 = 10

Q3. WAP to print factorial of a number

Eg. User Enter 5

Factorial will be 5 \* 4 \* 3 \* 2 \* 1 = 120

Q4. Write a Program to print Sum of the following Series

Eg. 12 + 22 + 32 + ….  + n2

Q5. Create a Restaurant App , which will ask the choice of food, example

1.       Burger

2.       Pizza

3.       Sandwich

4.       Coke

5.       Exit

If user give the choice between 1 to 4 , so system keep asking again , till the choice is not (5 .exit)

Hint :Create the same example with multiple if else and switch , also to solve this example you need a loop

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

. WAP to print the Following Outputs

a)      \*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

b)      1

12

123

1234

12345

c)       12345

1234

123

12

1

d)      A

AB

ABC

ABCD

ABCDE

e)      12345

   1234

      123

        12

          1

Q2. WAP to do the Sum of the Series

1/1 + 1/2 + 1/3 + 1/4 + 1/5  +…. + 1/n

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

1

12

123

1234

12345

A

AB

ABC

ABCD

ABCDE

\*\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

12345

1234

123

12

1

ABCDE

ABCD

ABC

AB

A

\*\*\*\*\*

 \*\*\*\*

  \*\*\*

   \*\*

     \*

12345

  1234

     123

       12

         1

ABCDE

   ABCD

      ABC

         AB

           A

\*\*\*\*\*\*\*\*\*\*

\*\*\*\*   \*\*\*\*

\*\*\*     \*\*\*

\*\*        \*\*

\*           \*

1234554321

1234   4321

123     321

12        21

1           1

A         A

AB     BA

ABC  CBA

ABCDDCBA

public class NestedLoops {

public static void main(String[] args) {

/\*for(int k = 1 ; k<=3 ; k++)

{System.out.println(k);

System.out.println("After K Loop");

}\*/

/\*for(int i = 1;i<=3 ;i++){

for(int j = 1 ; j<=3 ; j++){

System.out.println("I is "+i+" and j is "+j);

}

}\*/

/\*for(int i = 3;i>=1 ;i--){

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

System.out.println("I is "+i+" and j is "+j);

}

}\*/

// Star

/\*

 \*

 \*\*

 \*\*\*

 \*\*\*\*

 \*\*\*\*\*

 \*/

/\*for(int i = 1; i<=5 ; i++){

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

System.out.print("\*");

}

System.out.println();

}\*/

/\*for(int i = 5; i>=1 ; i--){

for(int k=5;k>i;k--){

System.out.print(" ");

}

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

System.out.print("\*");

}

System.out.println();

}\*/

for(int i = 5;i>=1 ; i--){

//for(int i = 1;i<=5 ; i++){

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

System.out.print("\*");

}

for(int k = 1; k<= (5\*2)-(i\*2);k++){

System.out.print(" ");

}

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

System.out.print("\*");

}

System.out.println();

}

}

}

Attachments area

NEW PROGRAM

import java.util.Scanner;

public class WaveDemo2 {

public static void main(String[] args) {

System.out.println("Enter the Times to Repeat Waves");

Scanner scanner = new Scanner(System.in);

int waveRange = scanner.nextInt();

int leftValue = 5;

int rightValue = leftValue+1;

int rowLeftValue = 0;

int rowRightValue = 0;

for(int left =4, common=1, right =1; common<=5 ; common++,left--,right++,leftValue--,rightValue++){

rowLeftValue = leftValue;

rowRightValue = rightValue;

for(int i =1 ; i<=waveRange; i++){

// Left Side Printing Code

for(int space = left; space>=1; space-- ){

System.out.print(" ");

} // space close

//rightValue = leftValue;

for(int t = 1; t<=1; t++){

System.out.print(rowLeftValue<=9?"0"+rowLeftValue:rowLeftValue);

//System.out.print("\*");

}

// Right Side Printing Code

for(int space = 1; space<=(right-1)\*2; space++ ){

System.out.print(" ");

} // space close

for(int t = 1; t<=1; t++){

//System.out.print("\*");

System.out.print(rowRightValue<=9?"0"+rowRightValue:rowRightValue);

}

for(int space = left; space>=1; space-- ){

System.out.print(" ");

}

rowLeftValue = rowLeftValue + 10;

rowRightValue = rowRightValue + 10;

}

System.out.println("");

}

}

}

Enter the Times to Repeat Waves

6

    0506        1516        2526        3536        4546        5556

   04  07      14  17      24  27      34  37      44  47      54  57

  03    08    13    18    23    28    33    38    43    48    53    58

 02      09  12      19  22      29  32      39  42      49  52      59

01        1011        2021        3031        4041        5051        60

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

public class LabeledLoopDemo {

public static void main(String[] args) {

amit:

for(int i = 1 ; i<=3 ; i++){

for(int j = 1; j<=3; j++)

{

if(i==j){

break amit;

//continue outerLoop;

}

else

{

System.out.println("I is "+i+" and J is "+j);

}

}

}

System.out.println("Exit from a Loop");

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

public class EnhanceForLoopDemo {

public static void main(String[] args) {

//int x[] = {1,2,3,4,5,90,78 , 555};

//int []x={1,2,3,4};

/\*int x[] = new int[10];

x[0]=100;

x[1]=200;

\*/

int x[]=new int[]{1,2,3,4,5};

//int e[][]= new int[3][3];

//int [][]e= new int[3][3];

//int []e[]=new int[3][3];

int [][]e = new int[5][];

e[0]= new int[1000];

e[1]= new int[2000];

e[2]= new int[3000];

e[1][1100]=1;

/\*for(int i = 0; i<=4;i++){

System.out.println(x[i]);

}

\*/

/\*for(int i = 0; i<x.length;i++){

System.out.println(x[i]);

}\*/

// Enhance for loop (java 1.5)

for(int y : x){

System.out.println(y);

}

}

}

**Q1)** A HighSchool application has two classes: the Person superclass and the Student subclass. Using inheritance, in this lab you will create two new classes, Teacher and CollegeStudent. A Teacher will be like Person but will have additional properties such as *salary* (the amount the teacher earns) and *subject* (e.g. “Computer Science”, “Chemistry”,  “English”, “Other”). The CollegeStudent class will extend the Student class by adding a *year*(current level in college) nd *major* (e.g. “Electrical Engineering”, “Communications”, “Undeclared”).

The inheritance hierarchy would appear as follows:

1.         Add methods to “set” and “get” the instance variables in the Person class. These would consist of: getName, getAge, getGender, setName,  setAge, and setGender.

2.         Add methods to “set” and “get” the instance variables in the Student class. These would consist of: getIdNum, getGPA, setIdNum, and setGPA.

3.         Write a Teacher class that extends the parent class Person.

a.  Add instance variables to the class for *subject* (e.g. “Computer Science”, “Chemistry”,, “English”, “Other”) and*salary* (the teachers annual salary). *Subject* should be of type String and *salary* of type double. Choose appropriate names for the instance variables.

b.  Write a constructor for the Teacher class. The constructor will use five parameters to initialize myName, myAge, myGender, *subject*, and *salary*.  Use the super reference to use the constructor in the Person superclass to initialize the inherited values.

c.  Write “setter” and “getter” methods for all of the class variables. For the Teacher class they would be: getSubject, getSalary, setSubject, and setSalary.

d.  Write the toString() method for the Teacher class. Use a super reference to do the things already done by the superclass.

4.         Write a CollegeStudent subclass that extends the Student class.

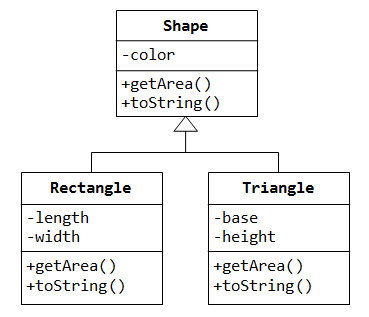
a.   Add instance variables to the class for *major* (e.g. “Electrical Engineering”, “Communications”, “Undeclared”) and *year* (e.g. FROSH = 1,  SOPH = 2, …). *Major* should be of type String and *year* of type int. Choose appropriate names for the instance variables.

b.   Write a constructor for the CollegeStudent class. The constructor will use seven parameters to initialize myName, myAge, myGender,  myIdNum, myGPA, *year*, and *major*. Use the super reference to use the constructor in the Student superclass to initialize the inherited values.

c.   Write “setter” and “getter” methods for all of the class variables. For the CollegeStudent class they would be: getYear, getMajor, setYear,  and setMajor.

d.   Write the toString() method for the CollegeStudent class. Use a super reference to do the things already done by the superclass.  
5.         Write a testing class with a main() that constructs all of the classes (Person, Student, Teacher, and CollegeStudent) and calls their toString()  method

**Q-2)** Consider the Following Image



Our program uses many kinds of shapes, such as triangle, rectangle and so on. You should design a super class called Shape, which defines the public interface (or behaviours) of all the shapes as mentioned in the above class diagram. And, we would like all the shapes to have a method called getArea(), which returns the area of that particular shape.

The subclasses override the getArea() method inherited from the super class, and provide the proper implementations for getArea().

Finally, create a **TestShape** class in our application, then create references of Shape, and assign them instances of subclasses. And call the *getArea()* methods of Rectangle & Triangle by invoking Shape references.

**Q-3)**

1. Design a new class called “CodeTimer”.  This class should contain the following data fields:
   1. A private long value named startTime
   2. A private long value named endTime
   3. Get methods for both startTime and endTime (i.e. getStartTime() and getEndTime()
   4. A no argument constructor that initializes startTime with the current time (hint: use System.currentTimeMillis() )
   5. A method named start() that resets that startTime to the current time
   6. A method named stop() that sets the endTime to the current time
   7. A method named getElapsedTime() that returns the elapsed time between startTime and endTime

**Java objective questions-**

<http://javaquestion.tripod.com/JavaQuestionAnswer.htm>

<https://www3.ntu.edu.sg/home/ehchua/programming/java/J2a_BasicsExercises.html>

<http://www.homeandlearn.co.uk/exercises/programming_exercises.html>

<https://www.cs.bham.ac.uk/~mdr/teaching/RedHotChilli/ex5A.html> ---- Inheritance

<http://java2novice.com/java-interview-programs/is-prime-number/>