***Dt : 12/5/2022***

***\*imp***

***Operators in Java:***

***=>Operator is a Special symbol used to perform operations.***

***=>The following are some important operators in Java:***

***1.Arithmetic Operators***

***2.Relational Operators***

***3.Logical Operators***

***4.Increment-Decrement Operators***

***1.Arithmetic Operators:***

***=>Arithmetic Operators are used to perform basic operations***

***or Fundamental operations.***

***Operator Meaning***

***+ Addition***

***- Subtraction***

***\* Multiplication***

***/ Division***

***% ModDivision***

***Ex:***

***a=7 b=2***

***a/b = 7/2 = 3***

***a%b = 7%2 = 1***

***2.Relational Operators:***

***=>Relational Operators are used to compare two values and***

***generate boolean result.***

***=>These operators are also known as Conditional Operators***

***Operator Meaning***

***> Greater Than***

***>= Greater Than or Equal to***

***< Less Than***

***<= Less Than or Equal to***

***== Is equal to***

***!= Not equal to***

***3.Logical Operators:***

***=>Logical Operators are used to compare two comparisions and***

***generate boolean result.***

***Operator Meaning***

***&& Logical AND***

***|| Logical OR***

***! Logical NOT***

***Logical AND(&&):***

***A B A&&B***

***T T T***

***F T F***

***T F F***

***F F F***

***Logical OR(||):***

***A B A||B***

***T T T***

***F T T***

***T F T***

***F F F***

***Logical NOT(!):***

***A !A***

***T F***

***F T***

***4.Increment-Decrement Operators:***

***=>Increment operator will increment the value by 1 and decrement***

***operator will decrement the value by 1.***

***Operator Meaning***

***++ Increment***

***-- Decrement***

***=======================================================***

***\*imp***

***Control Structures in Java:***

***=>The structures which are used to control the part of the***

***program are known as Control Structures.***

***=>These Control Structures are categorized into three types:***

***1.Selection Statements***

***2.Iterative Statements***

***3.Branching Statements***

***1.Selection Statements:***

***=>The Statements which are used to select one part of the***

***program for execution are known as Selection Statements or***

***Conditional Statements.***

***=>Types:***

***(a)Simple if***

***(b)if-else***

***(c)Nested if***

***(d)Ladder if-else***

***(e)switch-case***

***2.Iterative Statements:***

***=>The statements which are used to execute set-of-lines from the***

***program repeatedly on some condition are known as Iterative***

***Statements or Repeatitive Statements or Looping Structures.***

***=>Types:***

***(a)while loop***

***(b)do-while loop***

***(c)for loop***

***3.Branching Statements:***

***=>The statements which are used to transfer the control from***

***one location to another location are known as Branching Statements***

***or Transfer Statements.***

***=>Types:***

***(a)break***

***(b)continue***

***(c)return***

***(d)exit***

***Note:***

***=>'goto' statement is not available in Java.***

***===========================================================***

***Ex\_Program:***

***wap to read three integer values and perform comparision based***

***on User choice.***

***1.Greater***

***2.Smaller***

***Layout:***

***Greater.java***

***package p2;***

***public class Greater {***

***public int compare(int x,int y,int z)***

***{***

***if(x>y && x>z)***

***{***

***return x;***

***}***

***else if(y>x && y>z)***

***{***

***return y;***

***}***

***else***

***{***

***return z;***

***}***

***}***

***}***

***Smaller.java***

***package p2;***

***public class Smaller {***

***public int compare(int x,int y,int z)***

***{***

***if(x<y && x<z)***

***{***

***return x;***

***}***

***else if(y<x && y<z)***

***{***

***return y;***

***}***

***else***

***{***

***return z;***

***}***

***}***

***}***

***DemoComparision.java(MainClass)***

***package p1;***

***import java.util.Scanner;***

***import p2.Greater;***

***import p2.Smaller;***

***public class DemeComparision {***

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

***Scanner s = new Scanner(System.in);***

***System.out.println("Enter the Value1:");***

***int v1 = s.nextInt();***

***System.out.println("Enter the Value2:");***

***int v2 = s.nextInt();***

***System.out.println("Enter the Value3:");***

***int v3 = s.nextInt();***

***System.out.println("====Choice====");***

***System.out.println("1.Greater\n2.Smaller");***

***System.out.println("Enter the Choice:");***

***int choice = s.nextInt();***

***switch(choice)***

***{***

***case 1:***

***Greater gt = new Greater();***

***int r1 = gt.compare(v1,v2,v3);***

***System.out.println("Greater Value:"+r1);***

***break;***

***case 2:***

***Smaller sm = new Smaller();***

***int r2 = sm.compare(v1,v2,v3);***

***System.out.println("Smaller Value:"+r2);***

***break;***

***default:***

***System.out.println("Invalid Choice...");***

***}//end of switch***

***s.close();***

***}***

***}***

***========================================================***

***faq:***

***define switch-case statement?***

***=>swich-case statement is used to select one from multiple***

***available options or cases.***

***syntax:***

***switch(value)***

***{***

***case 1 : statements;***

***break;***

***case 2 : statements;***

***break;***

***.***

***.***

***case n : statements;***

***break;***

***default : statements;***

***}***

***behaviour:***

***=>The switch-value is compared with available options,if the***

***value is matched with any option then the statements under the***

***option are executed.***

***=>After executing the statements the switch-case execution is***

***stopped using 'break'.***

***=>If the switch-value is not matched with any available options***

***then default is executed.***

***=========================================================***

***Assignment:***

***wap to read two int values and perform arithmetic operation based***

***on User choice:***

***1.add***

***2.sub***

***3.mul***

***4.div***

***5.modDiv***

***=========================================================***