Quiz_day:2

Q1)//months

```
import java.util.Scanner;
class months {
    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        int  A=sc.nextInt();
       String arr[]=
{"jan","feb","march","april","may","june","july","august","september","octaber","november
","december"};
       for(int col=0;col<A;col++){
           System.out.println(arr[col]+" ");
       }
    }
}
```

Q2)//Leap year

```
import java.util.Scanner;
public class LeapYear {
   public static void main(String[] args){
      int year;
      System.out.println("Enter an Year :: ");
      Scanner sc = new Scanner(System.in);
      year = sc.nextInt();

      if (((year % 4 == 0) && (year % 100!= 0)) || (year%400 == 0))
         System.out.println("Specified year is a leap year");
      else
         System.out.println("Specified year is not a leap year");
   }
}
```

Q3)//Maximum of 2 numbers

```
import java.util.Scanner;

public class LargestofTwo {
	private static Scanner sc;
	public static void main(String[] args)
	{
		int number1, number2;
		sc = new Scanner(System.in);
		System.out.print(" Please Enter the First Number : ");
		number1 = sc.nextInt();
		System.out.print(" Please Enter the Second Number : ");
		number2 = sc.nextInt();
		if(number1 > number2)
	   {
			System.out.println("\n The Largest Number = " + number1);
	   }
	   else if (number2 > number1)
	   {
		System.out.println("\n The Largest Number = " + number2);
	   }
	   else
	   {
		System.out.println("\n Both are Equal");
	   }
	}
}
Q3)//Maximum of 3 numbers
import java.util.Scanner;
public class LargestNumberExample1
{
public static void main(String[] args)
{
int a, b, c, largest, temp;
//object of the Scanner class
Scanner sc = new Scanner(System.in);
//reading input from the user
```

```
System.out.println("Enter the first number:");
a = sc.nextInt();
System.out.println("Enter the second number:");
b = sc.nextInt();
System.out.println("Enter the third number:");
c = sc.nextInt();
//comparing a and b and storing the largest number in a temp variable
temp=a>b?a:b;
//comparing the temp variable with c and storing the result in the variable
largest=c>temp?c:temp;
//prints the largest number
System.out.println("The largest number is: "+largest);
}
}
```

Q4)//Even or Odd

```
import java.util.Scanner;

public class EvenOdd {

    public static void main(String[] args) {

        Scanner reader = new Scanner(System.in);

        System.out.print("Enter a number: ");
        int num = reader.nextInt();

        if(num % 2 == 0)
            System.out.println(num + " is  1");
        else
            System.out.println(num + " is  0");
    }
}
```

Q5)//Find minimum of two Numbers

```
public static void main(String[] args)
	{
		int number1, number2;
		sc = new Scanner(System.in);
		System.out.print(" Please Enter the First Number : ");
		number1 = sc.nextInt();
		System.out.print(" Please Enter the Second Number : ");
		number2 = sc.nextInt();
		if(number1 < number2)
	{
			System.out.println("\n The smallest Number = " + number1);
	}
	else if (number2 < number1)
	{
		System.out.println("\n The smallest Number = " + number2);
	}
	else
	{
		System.out.println("\n Both are Equal");
	}
	}
}

Q6)//minimum of 3 numbers
import java.util.Scanner;
public class LargestNumberExample1
{
public static void main(String[] args)
{
int a, b, c, smallest, temp;
//object of the Scanner class
Scanner sc = new Scanner(System.in);
//reading input from the user
System.out.println("Enter the first number:");
a = sc.nextInt();
System.out.println("Enter the second number:");
```

```
b = sc.nextInt();
System.out.println("Enter the third number:");
c = sc.nextInt();
//comparing a and b and storing the smallest number in a temp variable
temp=a<b?a:b;
//comparing the temp variable with c and storing the result in the variable
smallest=c<temp?c:temp;
//prints the smallest number
System.out.println("The smallest number is: "+smallest);
}
}
```

Q7)//You are given 3 integer angles(in degrees) **A, B** and **C** of a triangle. You have to tell whether the triangle is valid or not.

```
import java.util.Scanner;
class GFG {

	// Function to check if sum of the
	// three angles is 180 or not
	public static int Valid(int a, int b, int c)
	{
		// check condition
		if (a + b + c == 180 && a != 0 && b != 0 && c != 0)
			return 1;
		else
			return 0;
	}

	// Driver Code
	public static void main(String args[])
	{

	Scanner sc=new Scanner(System.in);
	System.out.println("enter a value of a,b,c");
	int a=sc.nextInt();
	int b=sc.nextInt();
	int c=sc.nextInt();
```

```
        // function calling and print output
        if ((Valid(a, b, c)) == 1)
            System.out.print("1");
        else
            System.out.print("0");
    }
}
```

Q7)//Write a program to input from user three numbers(**A, B & C**) representing side lengths of a triangle.
You have to print if the traingle is "equilateral", "scalene" or "isosceles".

```
import java.util.Scanner;
class GFG{
// Function to check if the triangle
// is equilateral or isosceles or scalene
static void checkTriangle(int x, int y, int z)
{

    // Check for equilateral triangle
    if (x == y && y == z )
        System.out.println("Equilateral Triangle");

    // Check for isosceles triangle
    else if (x == y || y == z || z == x )
        System.out.println("Isosceles Triangle");

    // Otherwise scalene triangle
    else
        System.out.println("Scalene Triangle");
}

// Driver Code
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
```

```
int x=sc.nextInt();
int y=sc.nextInt();
int z=sc.nextInt();
        // Function call
        checkTriangle(x, y, z);
}
}
```

Q8)//Take an integer **A** as input. You have to tell whether **A** is divible by **both 5 and 11** or not.

```
import java.util.Scanner;

public class Divisibleby5and11 {

        private static Scanner sc;
        public static void main(String[] args)
        {
                int number;
                sc = new Scanner(System.in);
                System.out.print(" Please Enter any Number to Check whether it is Divisible by 5 and 11 : ");
                number = sc.nextInt();
                if((number % 5 == 0) && (number % 11 == 0))
                {
                        System.out.println("\n Given number " + number + " is Divisible by 5 and 11");
                }
                else
                {
                        System.out.println("\n Given number " + number + " is Not Divisible by 5 and 11");
                }
        }
}
```

Q9)//You are given a Bank account having N amount and you are asked to perform **ADD**(credit) or **SUBTRACT**(debit) operation of an amount X.]
After the operation **print the amount left** in the Bank account. If the debit amount is greater than current balance print **"Insufficient Funds"**(without quotes) and the operation is skipped.

```
class Bank {

	// Initial balance $100
	int total = 100;

	// Money withdrawal method. Withdraw only if
	// total money greater than or equal to the money
	// requested for withdrawal

	// Method
	// To withdraw money
	void withdrawn(String name, int withdrawal)
	{
		if (total >= withdrawal) {
			System.out.println(name + " withdrawn "
							+ withdrawal);

			total = total - withdrawal;
			System.out.println("Balance after withdrawal: "
							+ total);
			// Making the thread sleep for 1 second after
			// each withdrawal

			// Try block to check for exceptions
			try {

				// Making thread t osleep for 1 second
				Thread.sleep(1000);
			}
```

```
            // Catch block to handle the exceptions
            catch (InterruptedException e) {

                // Display the exception along with line
                // number
                // using printStacktrace() method
                e.printStackTrace();
            }
        }

        // If the money requested for withdrawal is greater
        // than the balance then deny transaction*/
        else {

            // Print statements
            System.out.println(name
                               + " you can not withdraw "
                               + withdrawal);

            System.out.println("your balance is: " + total);

            // Making the thread sleep for 1 second after
            // each transaction failure

            // Try block to check for exceptions
            try {
                Thread.sleep(1000);
            }

            catch (InterruptedException e) {

                e.printStackTrace();
            }
        }
    }
```

```
        // Method - to deposit money
        // Accept money whenever deposited
        void deposit(String name, int deposit)
        {
                System.out.println(name + " deposited " + deposit);
                total = total + deposit;
                System.out.println("Balance after deposit: "
                                        + total);
                // Making the thread sleep for 1 second after
                // each deposit
                try {
                        Thread.sleep(1000);
                }
                catch (InterruptedException e) {
                        e.printStackTrace();
                }
        }
}
// Class 2
// main class
class GFG {

        // Main driver method
        public static void main(String[] args)
        {
                // Declaring an object of Bank class and calling the
                // withdarwn and deposit methods with suitable
                // parameters

                // Creating object of class 1 inside main()
                Bank obj = new Bank();

                // Custom input - Transactions
                obj.withdrawn("Arnab", 20);
                obj.withdrawn("Monodwip", 40);
                obj.deposit("Mukta", 35);
```

```
            obj.withdrawn("Rinkel", 80);
            obj.withdrawn("Shubham", 40);
        }
}
```