JAVA PROGRAMMING ASSIGNMENT – 4

PART A – THEORY QUESTIONS

1)What is abstraction in java.

Ans:Abstraction in means showing only what is necessary and hiding the internal details.

In simple words we know what but don't know how.

It is nothing but Data Hiding.

Using Abstract class and Interface we can achieve Abstraction

In Abstraction we are hide a logic between method

Abstraction hides the implementation and shows only the functionality.

2)What is abstract class

Ans:A class contain one or more abstract method that class called abstract class.

abstract keyword is used for abstract class.

A class can have constructor,concrete methode,static data member and Abstract methods.

Using Inheritance we can create abstract class object.

without Inheritance directly we cannot create object.

Abstract method can not be static.

In abstract method defination of method is not present,Only declaration is present.

In abstract class have abstract and non abstract method.

```
ex. abstract class ClassName {
    abstract void method1();   // abstract method
    void method2() {           // concrete method
        // code
    }
}
```

3)What is an interface in Java.

AnsAn interface is a blueprint of a class.

It contains only method declarations (no body) and constants.

It tells what to do, not how to do.

Using interface keyword we can create interface.

Interface is a contract between class and methods.

Interface can contain static,default and abstract methods.By default any method is public abstract.By default any data member is public static final.

We can extend Interface with other Interface using 'extends' keyword.

Using interface we can achieve multiple Inheritance.

Implement keyword is ued for override methods from interface.

Nested interface also allowed.

Inheritance types: 1]Functional 2]Mutable 3]Simple

How to implement Interface

1]using sub class.

2]Anonymous Inner class.

3]Using lambda function.

4)Difference between abstract class and interface.

Ans: Abstract class Interface

1)In abstract class contain abstract and non abstract methods.	1)In Interface contains abstract,default,static methods.
2)It have instance variable	2) In Interface Variable are public static final by default.
3)Abstract can have Constructor	3)Interface cannot have Constructor
4)In this extend keyword is used	4)In this implement keyword is used
5)Methods can be public ,protected public.	5)Methods are public by default.

5)What is Constructor.

Ans:Constructor is ued to initialization of object.

while creating a object constructor is called automatically.

Constructor is special function without return type.

Class name and Constructor name must be same.

Constructor can be public,private and protected.

this() is also known as Constructor call.

Constructor call for each object.

Default/No parameter constructor is already given by JVM

syntax:

```
        public className() {

       }
```

Need of constructor is intilazation of object first time while creation.

It is faster than setter.

PART B – PROGRAMMING QUESTIONS

1)Write program to count the number of digit in a number.

Ans:
```
import java.util.Scanner;

 public class CountDigits {

public static void main(String[] args) {

              Scanner sc = new Scanner(System.in);

              System.out.print("Enter a number: ");

              int num = sc.nextInt();

             int count = 0;

              if (num == 0) {

                        count = 1;

                  } else {

                     while (num != 0) {

                        count++;

                        num = num / 10;

                     }

                  }

                  System.out.println("Number of digit:"+count);

              }

              }
```

2)Write program to find the greatest commom divisor (gcd)   .

Ans:
```
import java.util.Scanner;


class GCD {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first number: ");
        int a = sc.nextInt();
        System.out.print("Enter second number: ");
        int b = sc.nextInt();
        int gcd = 1;
        for (int i = 1; i <= a && i <= b; i++) {
            if (a % i == 0 && b % i == 0) {
                gcd = i;
            }
        }
        System.out.println("GCD = " + gcd);
    }
}
```

3)Write program to claculate LCM of two number.

Ans:import java.util.Scanner;

```
public class LCMExample {
    public static void main(String[] args) {
        int a, b, lcm;
        int max;
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter first number: ");
        a = sc.nextInt();

        System.out.print("Enter second number: ");

        b = sc.nextInt();

        max = (a > b) ? a : b;

               while (true) {

            if (max % a == 0 && max % b == 0) {

                lcm = max;

                break;

            }

            max++;

        }

        System.out.println("LCM of " + a + " and " + b + " is: " + lcm);

    }

}
```

4)Write a program to check if a year is leap year.

```
Ans:import java.util.Scanner;


public class LeapYear {

    public static void main(String[] args) {

        int year;

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter a year: ");

        year = sc.nextInt();

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

5)Write program to print all even numbers between 1 and 50.

```
Ans:public class EvenNumbers {
    public static void main(String[] args) {
        System.out.println("Even numbers between 1 and 50:");
        for (int i = 1; i <= 50; i++) {
            if (i % 2 == 0) {
                System.out.print(i + " ");
            }
        }
    }
}
```

6)Write a program to calculate power of a number.

```
Ans: import java.util.Scanner;

public class PowerOfNumber {
    public static void main(String[] args) {
        int base, exponent;
        long result = 1;
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter base: ");
        base = sc.nextInt();

        System.out.print("Enter exponent: ");
        exponent = sc.nextInt();

        for (int i = 1; i <= exponent; i++) {
            result = result * base;
        }
        System.out.println(base + " power " + exponent + " = " + result);
    }
}
```