Java-assignment 3.

1.Explain single, multilevel, and hierarchical inheritance.

Single Inheritance:

In this type, a class inherits from only one base class. It is the simplest form of inheritance, creating a direct parent-child relationship. Multilevel Inheritance: This involves a chain of inheritance. A class inherits from a base class, which in turn inherits from another base class, and so on. For example, class C inherits from B, and B inherits from A. Hierarchical Inheritance: In this structure, multiple classes inherit from a single base class. This allows several specialized classes to share common functionalities from one general class.

2..What is polymorphism?

Polymorphism in Java is one of the core concepts in OOP that allows objects to behave differently based on their specific class type. The word polymorphism means having many forms, and it comes from the Greek words poly (many) and morph (forms). This means one entity can take many forms.

3.Difference between compile-time and runtime polymorphism.

Compile Time Polymorphism: The object's behavior is decided at compile time, this is known as the compile-time polymorphism. At compile-time, java knows which method to call by checking the method signatures. So this is called compile-time polymorphism. Method Overloading says you can have more than one function with the same name in one class having a different prototype.

Run-Time Polymorphism: Whenever an object is bound with the functionality at run time, this is known as runtime polymorphism. The runtime polymorphism can be achieved by method overriding. Java virtual machine determines the proper method to call at the runtime. It means if the child class provides the specific implementation of the method that has been provided by one of its parent classes then it is known as method overriding.

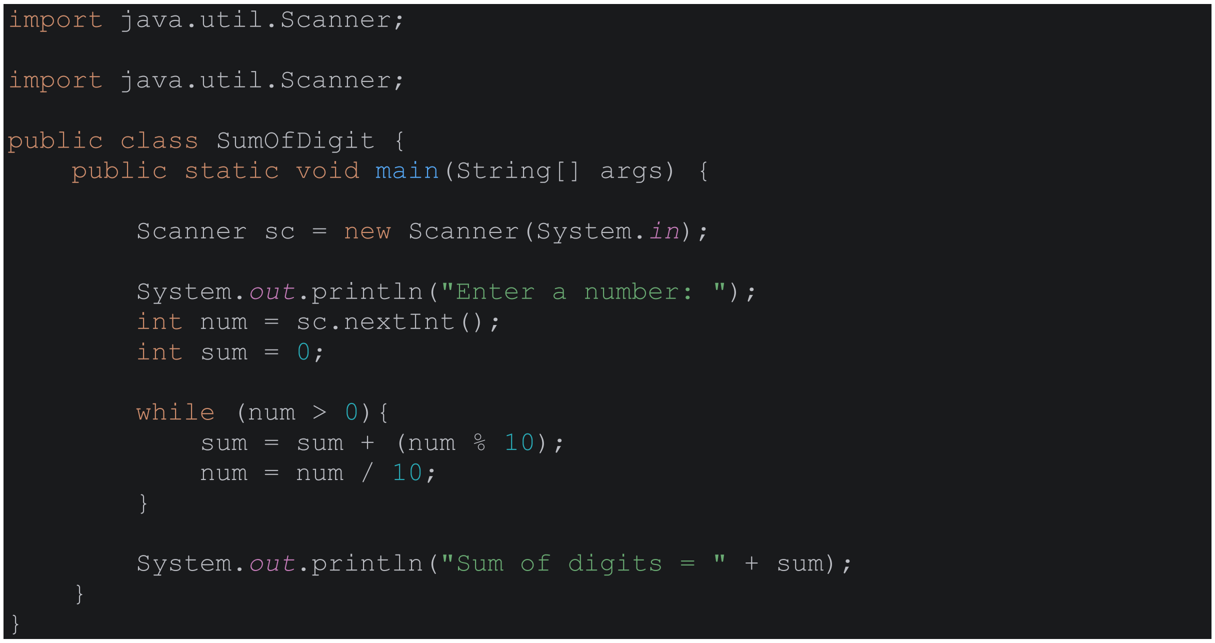
4.What is method overloading?

Method overloading means defining multiple methods with the same name but different parameter lists. The method call is resolved at compile time based on the arguments passed.

5.What is method overriding?

Method overriding occurs when a subclass provides its own implementation of a method already defined in its superclass. The method must have the same name, parameters, and return type.

1. .Write a program to find the sum of digits of a number.



```
import java.util.Scanner;

import java.util.Scanner;

public class SumOfDigit {
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter a number: ");
        int num = sc.nextInt();
        int sum = 0;

        while (num > 0){
            sum = sum + (num % 10);
            num = num / 10;
        }

        System.out.println("Sum of digits = " + sum);
    }
}
```

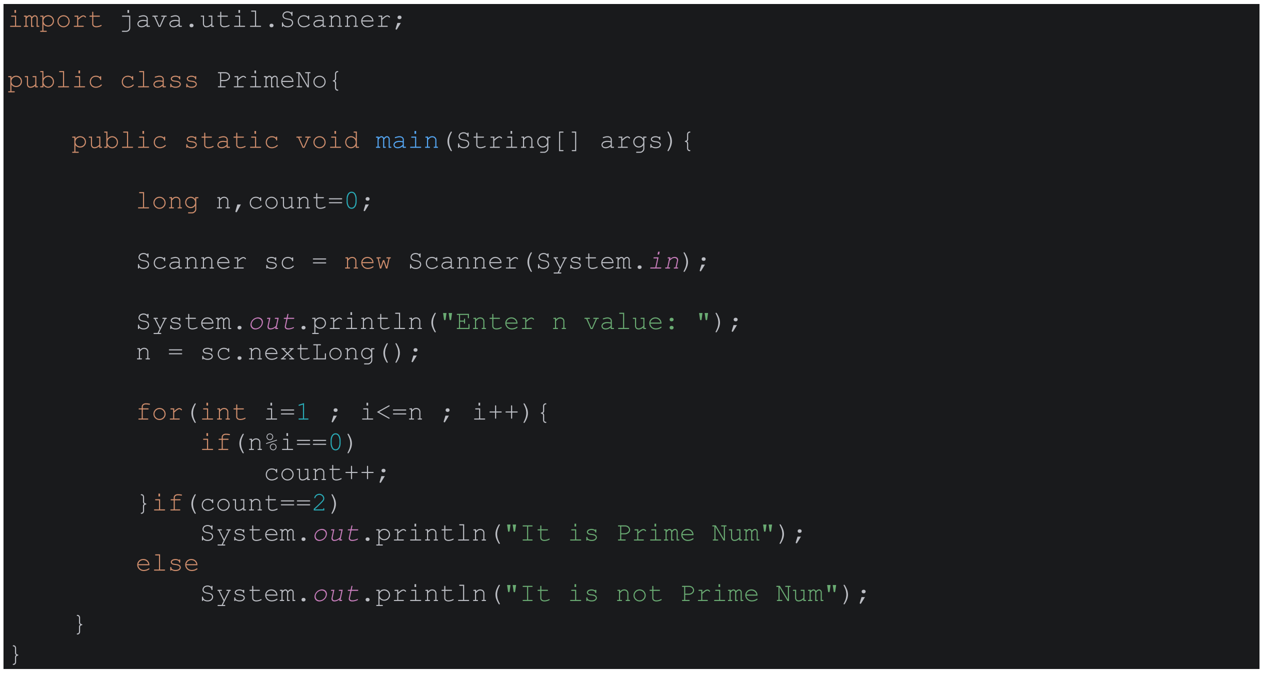
output:-

Enter a number:

12345

Sum of digits = 15

2.Write a program to check if a number is prime.



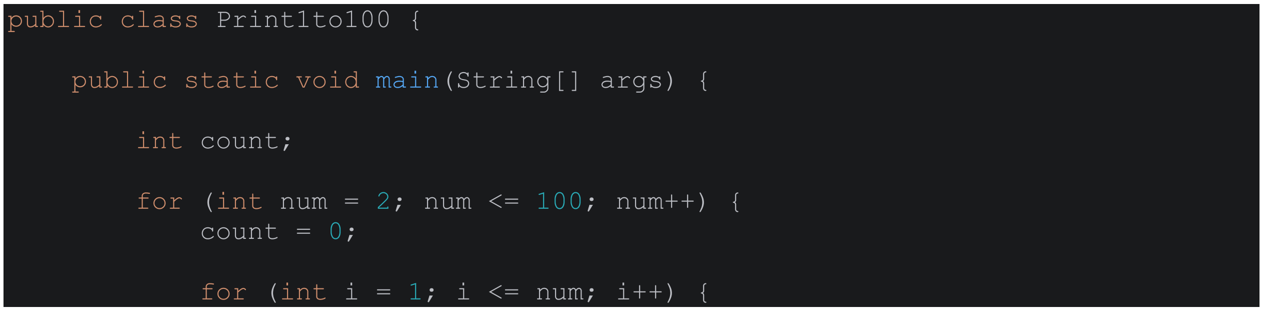
```
import java.util.Scanner;

public class PrimeNo{

    public static void main(String[] args){

        long n,count=0;

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter n value: ");
        n = sc.nextLong();

        for(int i=1 ; i<=n ; i++){
            if(n%i==0)
                count++;
        }if(count==2)
            System.out.println("It is Prime Num");
        else
            System.out.println("It is not Prime Num");
    }
}
```

Output:- Enter n value:

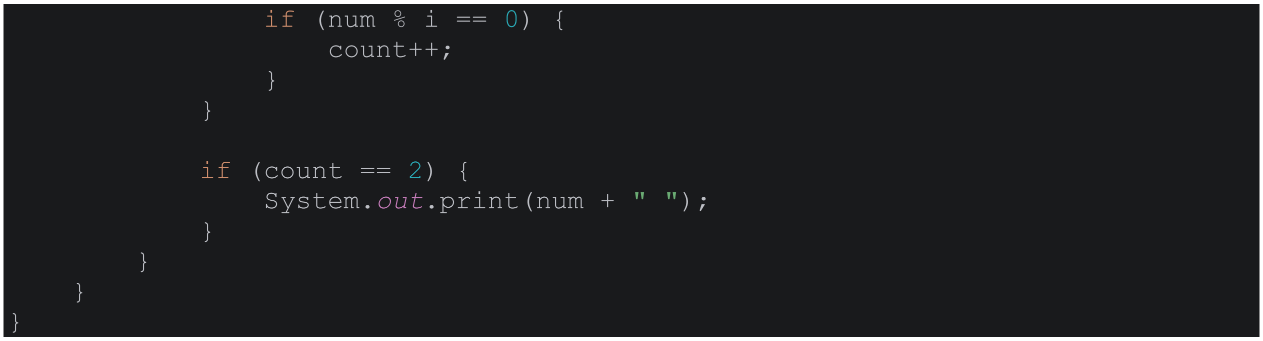
13

It is Prime Num

3.Write a program to print all prime numbers between 1 and 100.



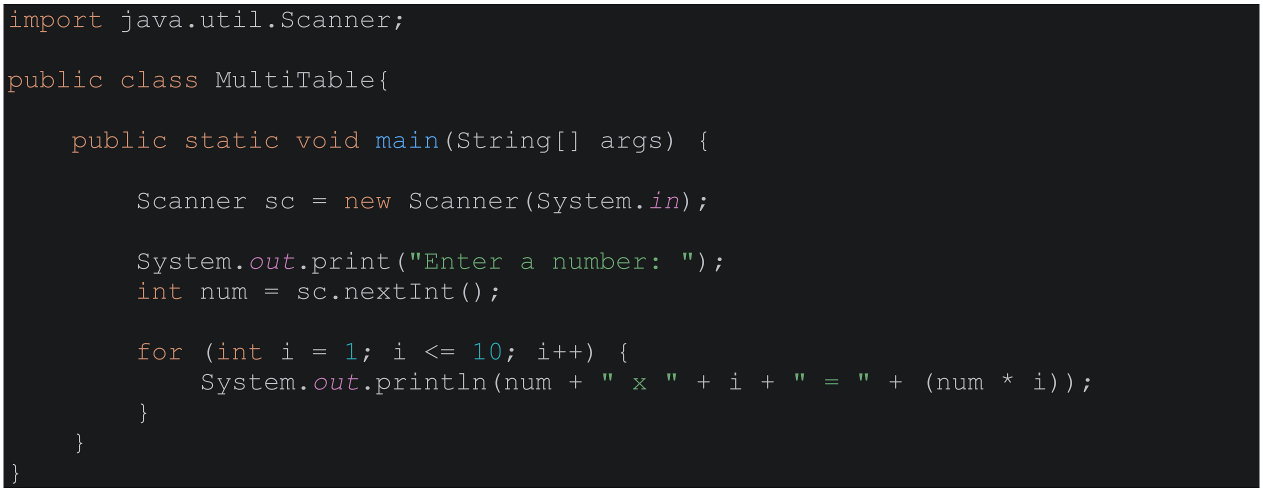
```
public class Print1to100 {

    public static void main(String[] args) {

        int count;

        for (int num = 2; num <= 100; num++) {
            count = 0;

            for (int i = 1; i <= num; i++) {
```



```
                if (num % i == 0) {
                    count++;
                }
            }

            if (count == 2) {
                System.out.print(num + " ");
            }
        }
    }
}
```

Output:- 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

4 : Write a program to print multiplication table of a number.



```
import java.util.Scanner;

public class MultiTable{

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

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

        for (int i = 1; i <= 10; i++) {
            System.out.println(num + " x " + i + " = " + (num * i));
        }
    }
}
```

Output:- Enter a number: 4

4 x 1 = 4

4 x 2 = 8

4 x 3 = 12

4 x 4 = 16

4 x 5 = 20

4 x 6 = 24

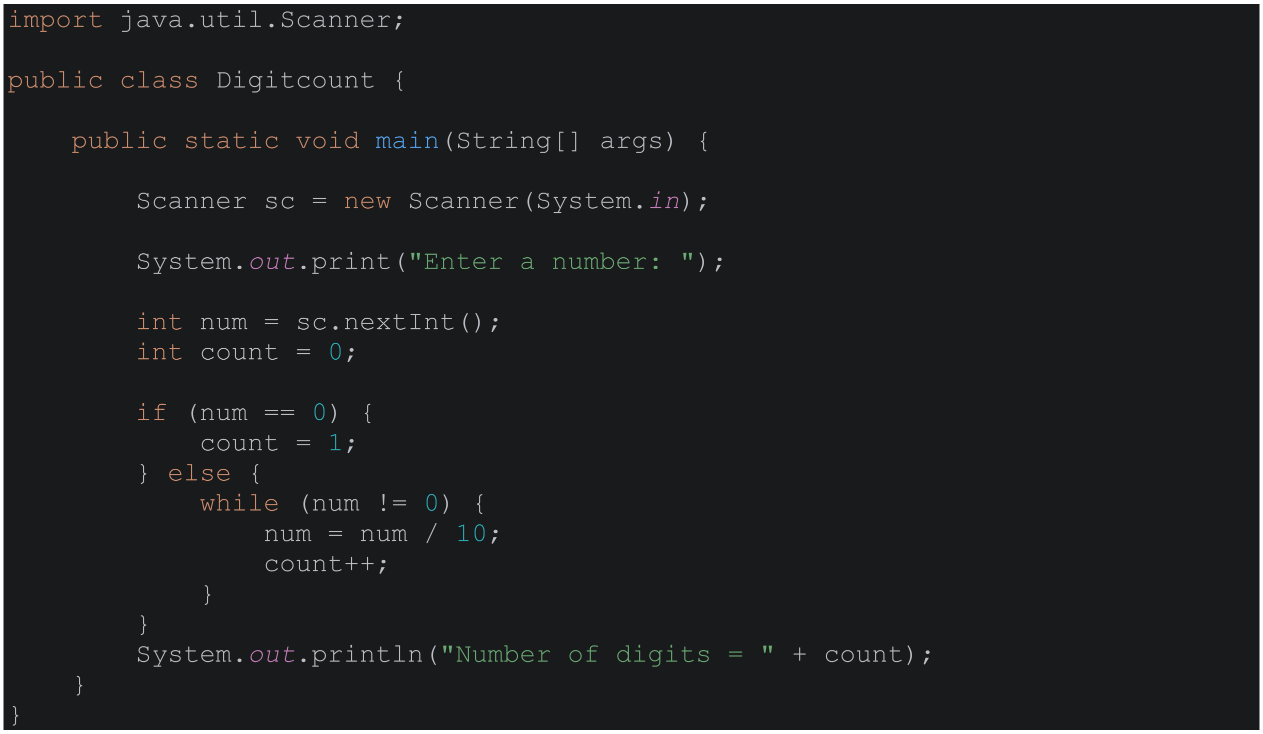
4 x 7 = 28

4 x 8 = 32

4 x 9 = 36

4 x 10 = 40

5. Write a program to count the number of digits in a number.



```java
import java.util.Scanner;

public class Digitcount {

    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) {
                num = num / 10;
                count++;
            }
        }
        System.out.println("Number of digits = " + count);
    }
}
```

Output:-

Enter a number: 1234

Number of digits = 4