



M Cynthia Shree

CH.SC.U4CSE24110

OBJECT ORIENTED PROGRAMMING (23CSE111)

LAB RECORD



**AMRITA VISHWA VIDYAPEETHAM AMRITA
SCHOOL OF COMPUTING, CHENNAI**

BONAFIDE CERTIFICATE

This is to certify that the Lab Record work for 23CSE111-
Object Oriented Programming Subject submitted by
CH.SC.U4CSE24112 – M Cynthia Shree in “**Computer
Science and Engineering**” is a Bonafide record of the
work carried out under my guidance and supervision at
Amrita School of Computing, Chennai.

This Lab examination held on

Internal Examiner 1

Internal Examiner 2

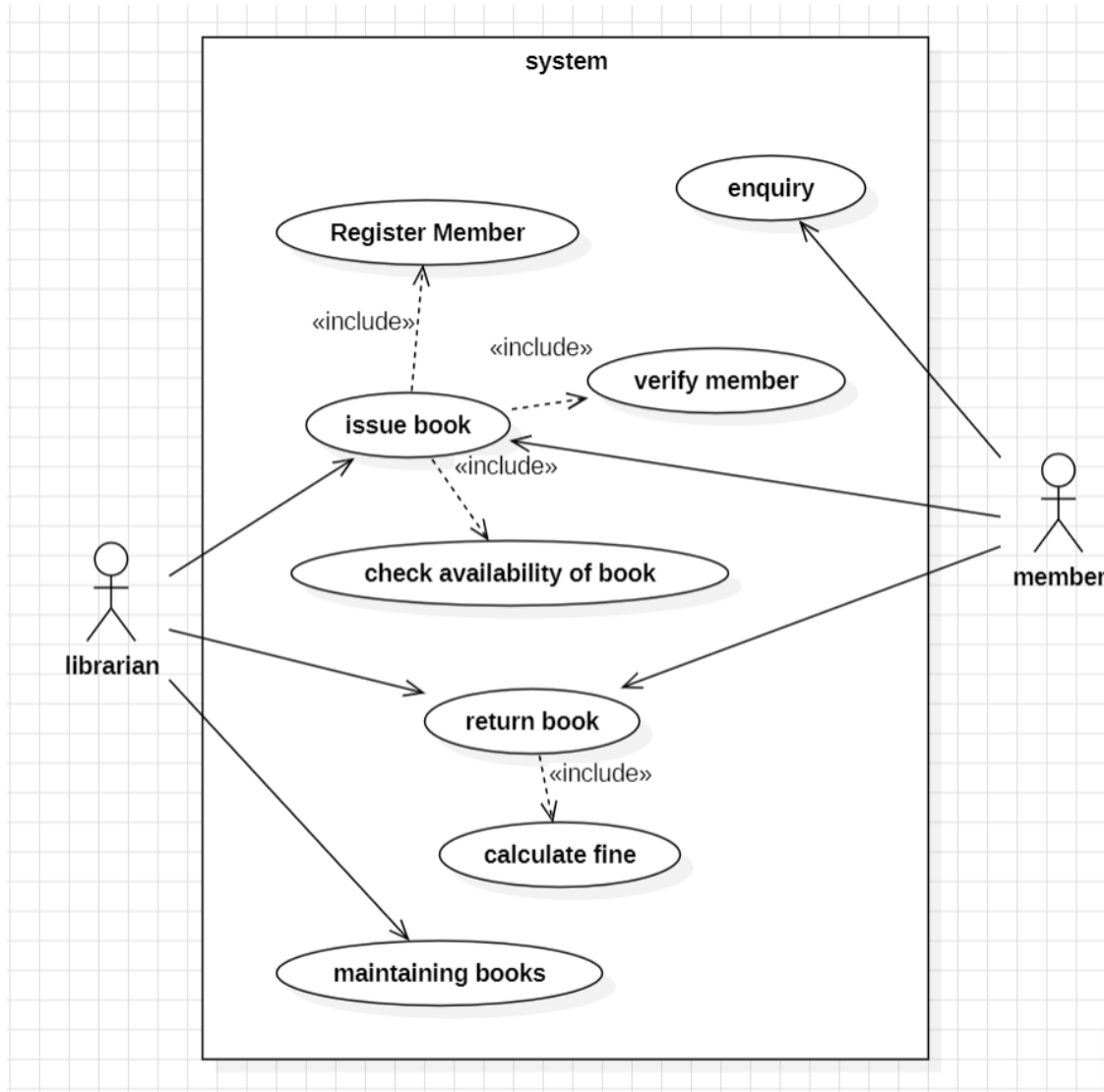
INDEX

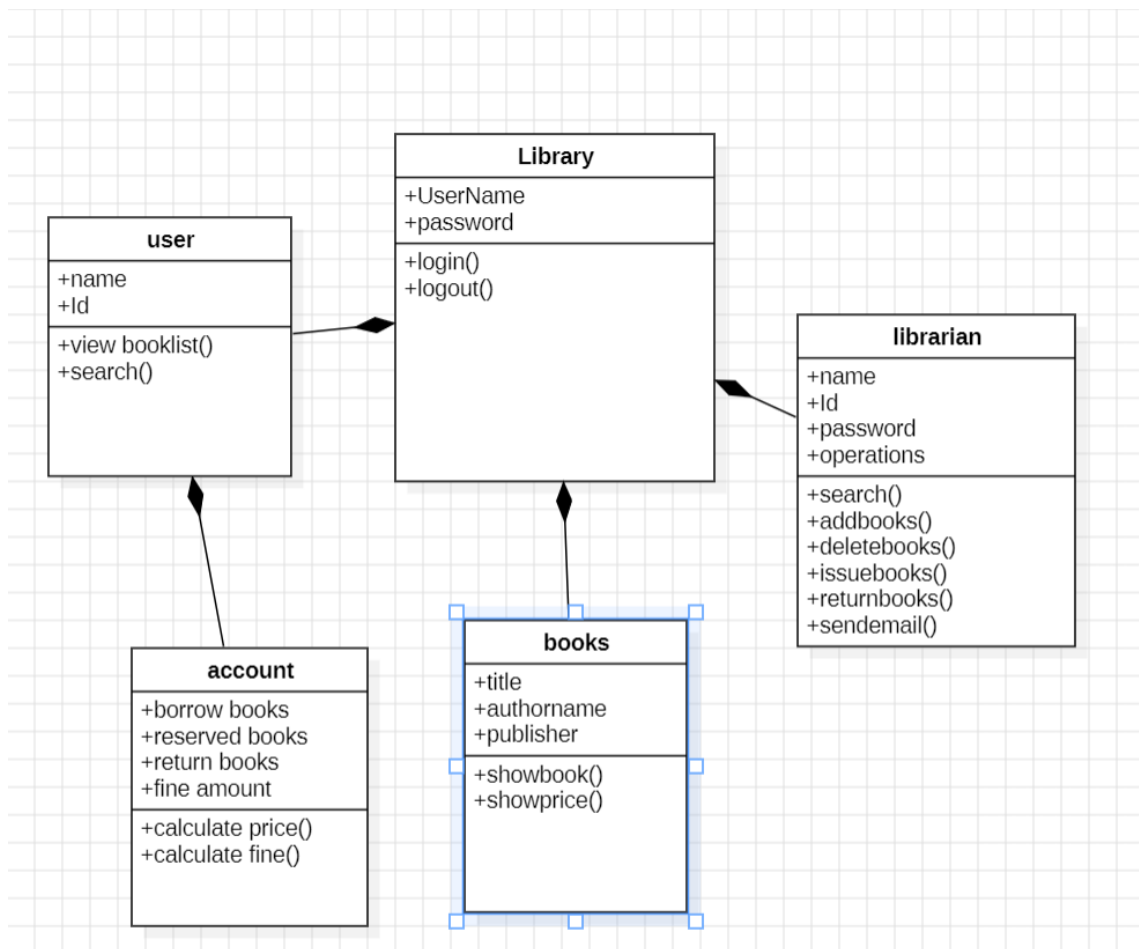
S.NO	TITLE	PAGE NO.
UML DIAGRAMS		
1.	LIBRARY MANAGEMENT SYSTEM	
	A) USE CASE DIAGRAM	4
	B) CLASS DIAGRAM	5
	C) SEQUENCE DIAGRAM	6
	D) STATE ACTIVITY DIAGRAM	7
	E) OBJECT DIAGRAM	8
2.	ONLINE FOOD ORDERING SYSTEM	
	A) USE CASE DIAGRAM	9
	B) CLASS DIAGRAM	10
	C) SEQUENCE DIAGRAM	11
	D) STATE ACTIVITY DIAGRAM	12
	E) OBJECT DIAGRAM	13
3.	BASIC JAVA PROGRAMS	
	A) BASIC CALCULATOR	14
	B) PRIME NUMBER CHECK	15
	C) FIBONACCI SERIES	16
	D) FACTORIAL OF A NUMBER	17
	E) PALINDROME	18
	F) EVEN OR ODD NUMBER	19
	G) REVERSE OF A NUMBER	20
	H) SUM OF DIGITS	21
	I) PRINT MULTIPLICATION TABLE	22
	J) MAXIMUM OF TWO NUMBERS	23

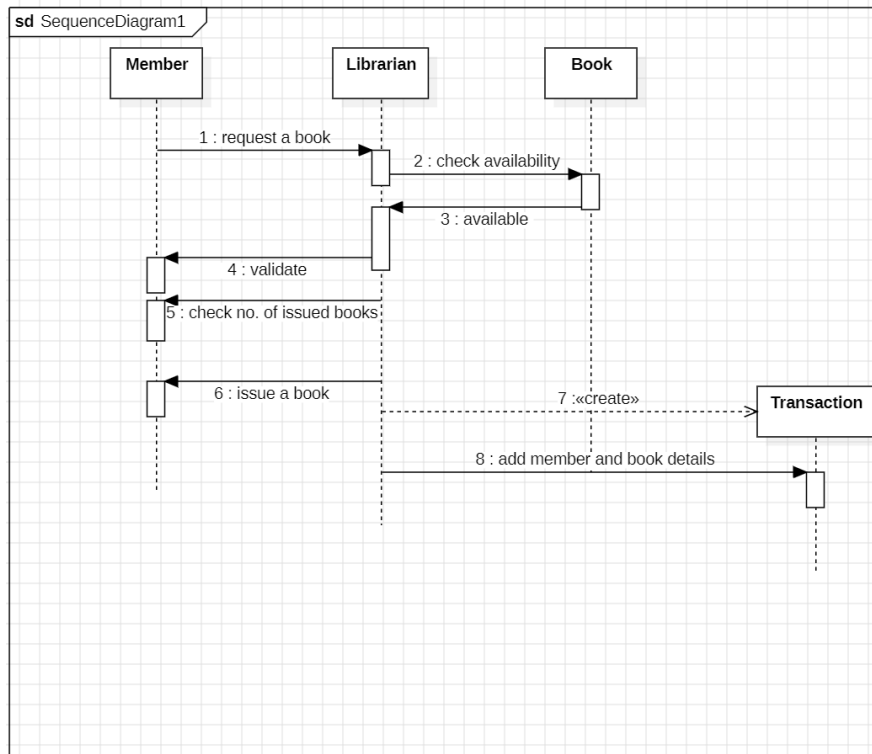
UML DIAGRAMS

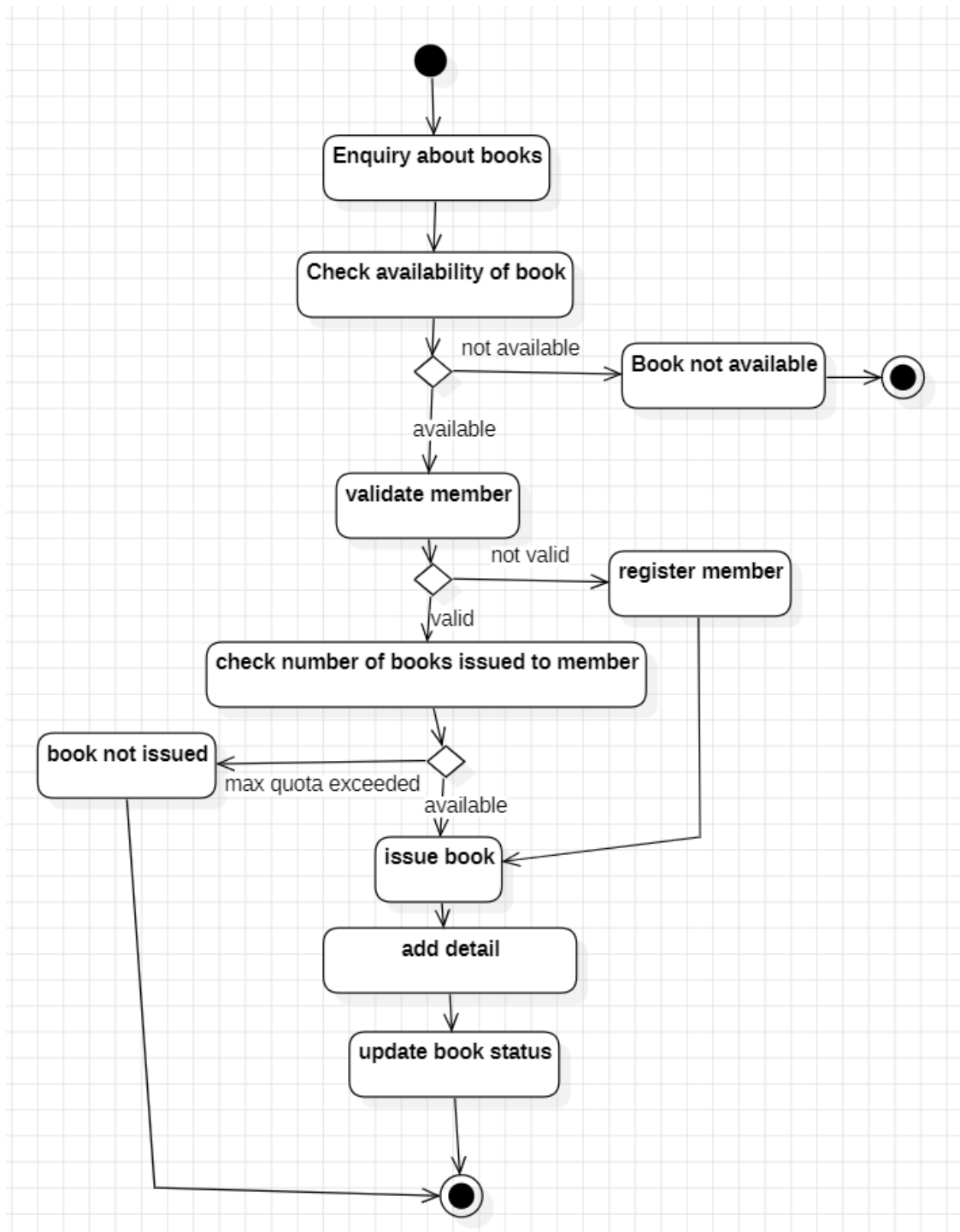
1. LIBRARY MANAGEMENT SYSTEM

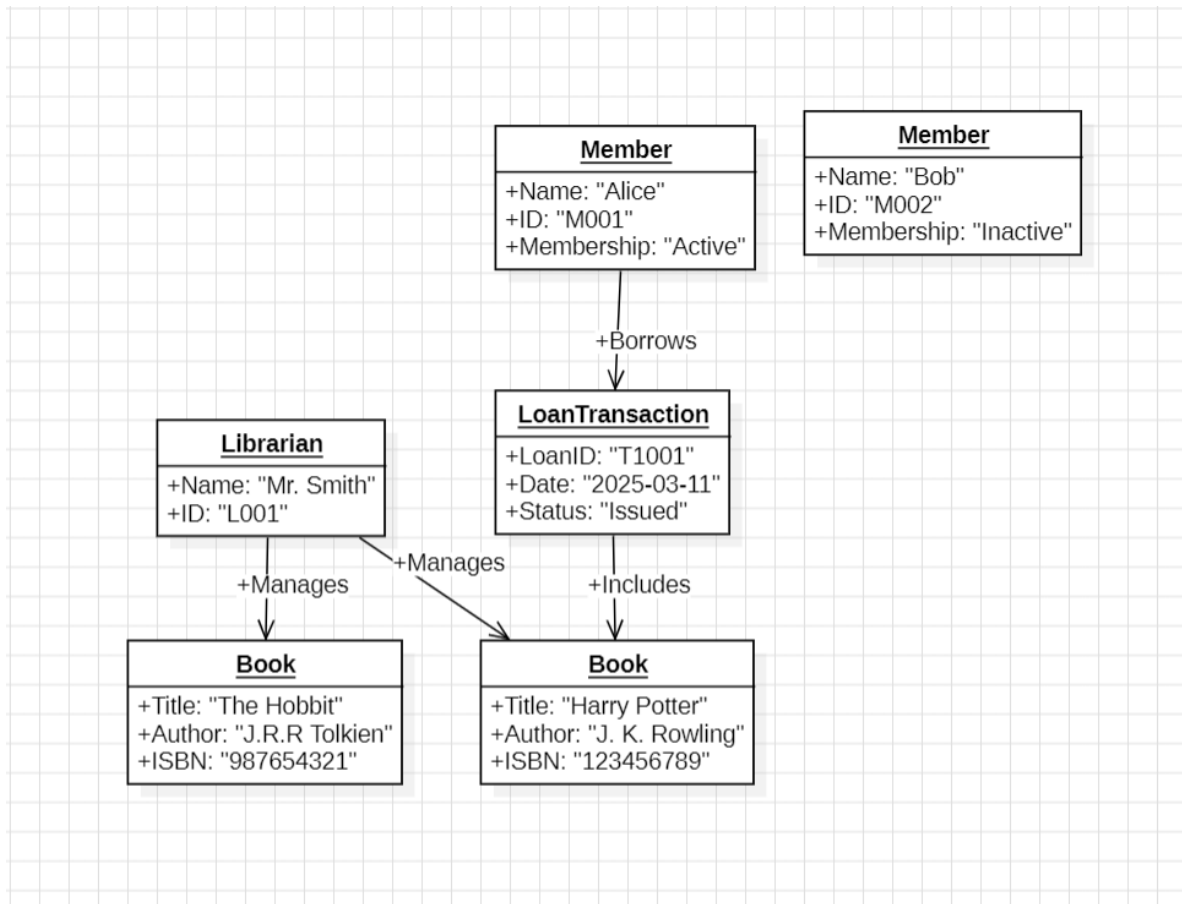
A) Use Case Diagram:



B) Class Diagram:

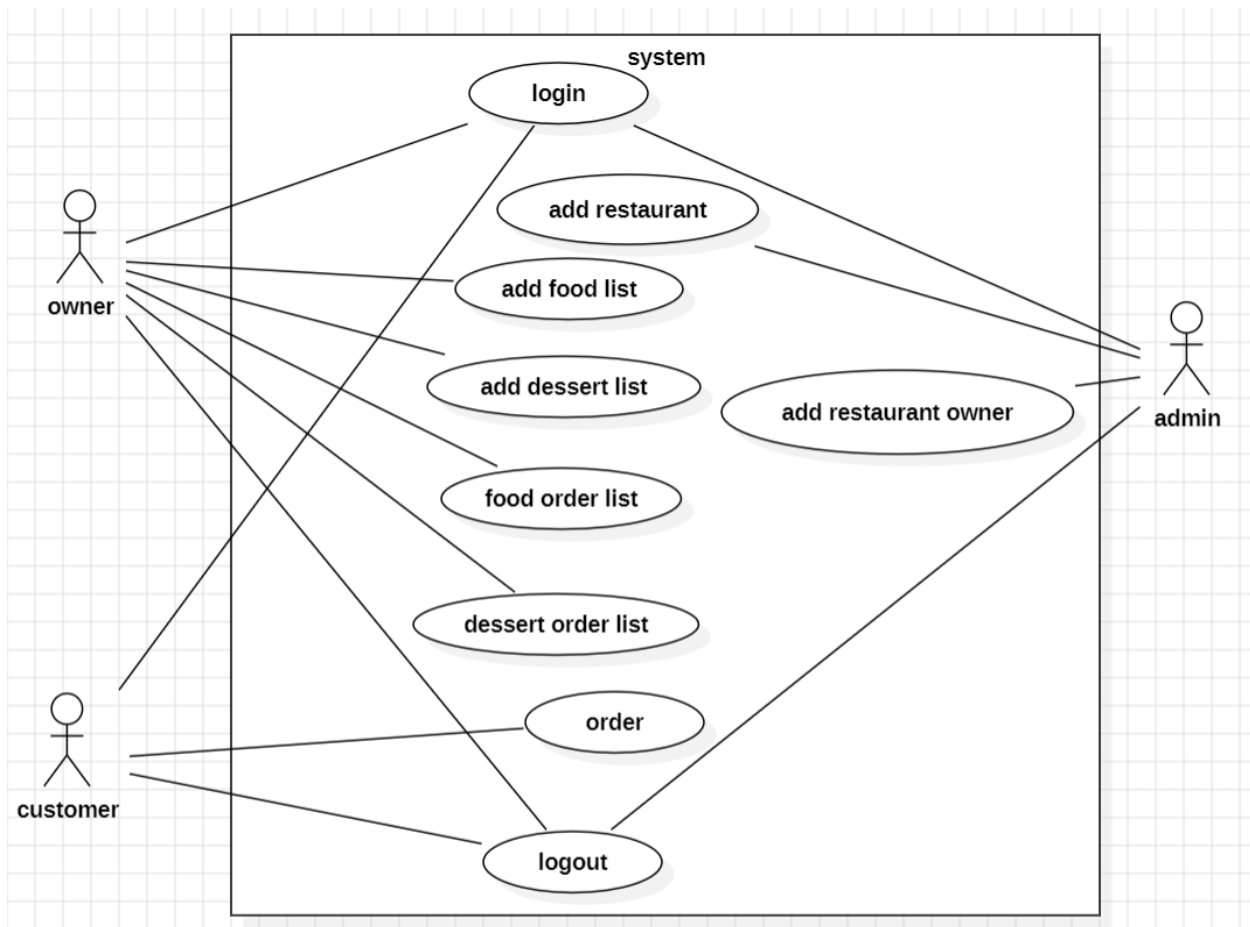
C) Sequence Diagram:

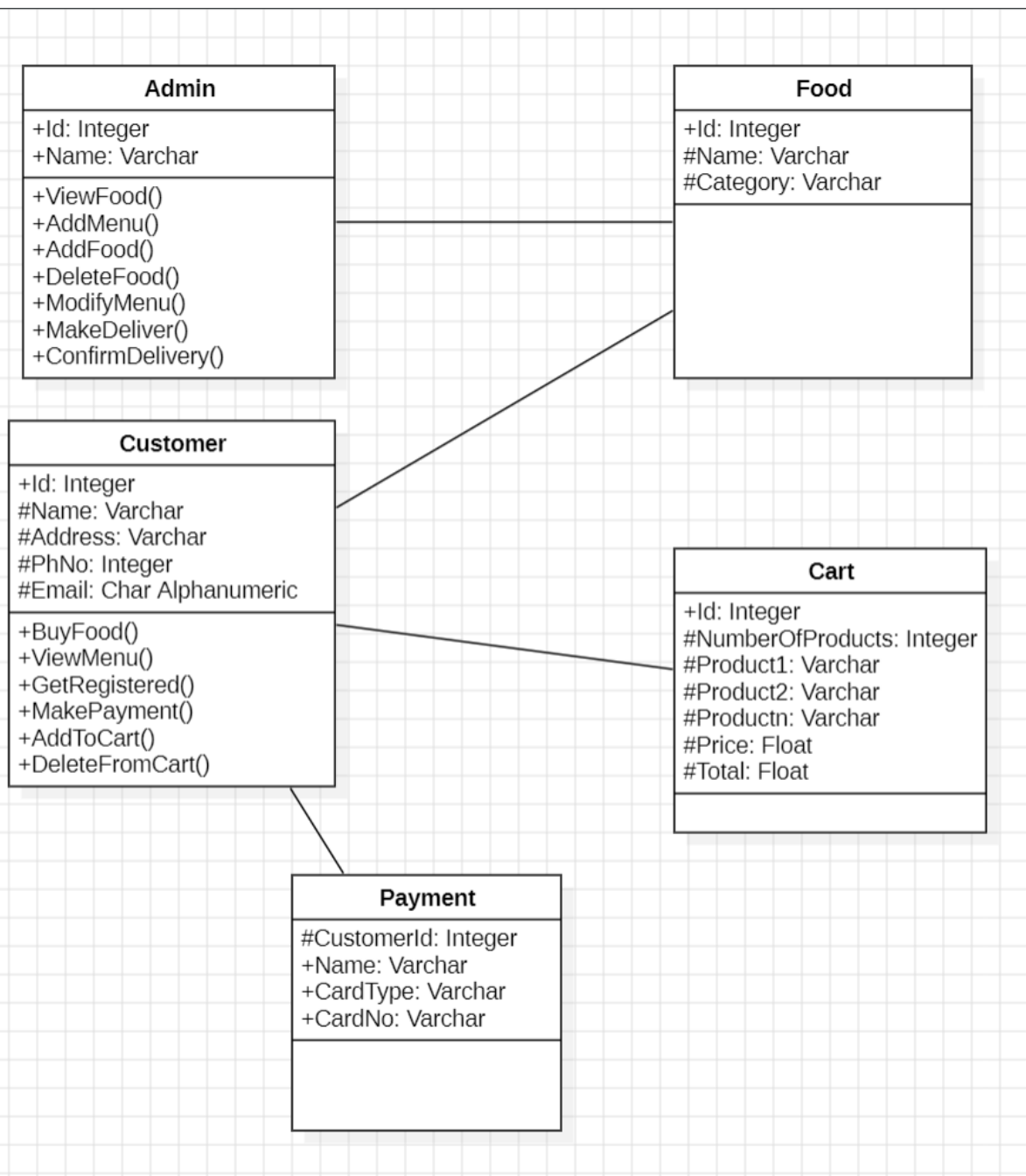
D) State Activity Diagram:

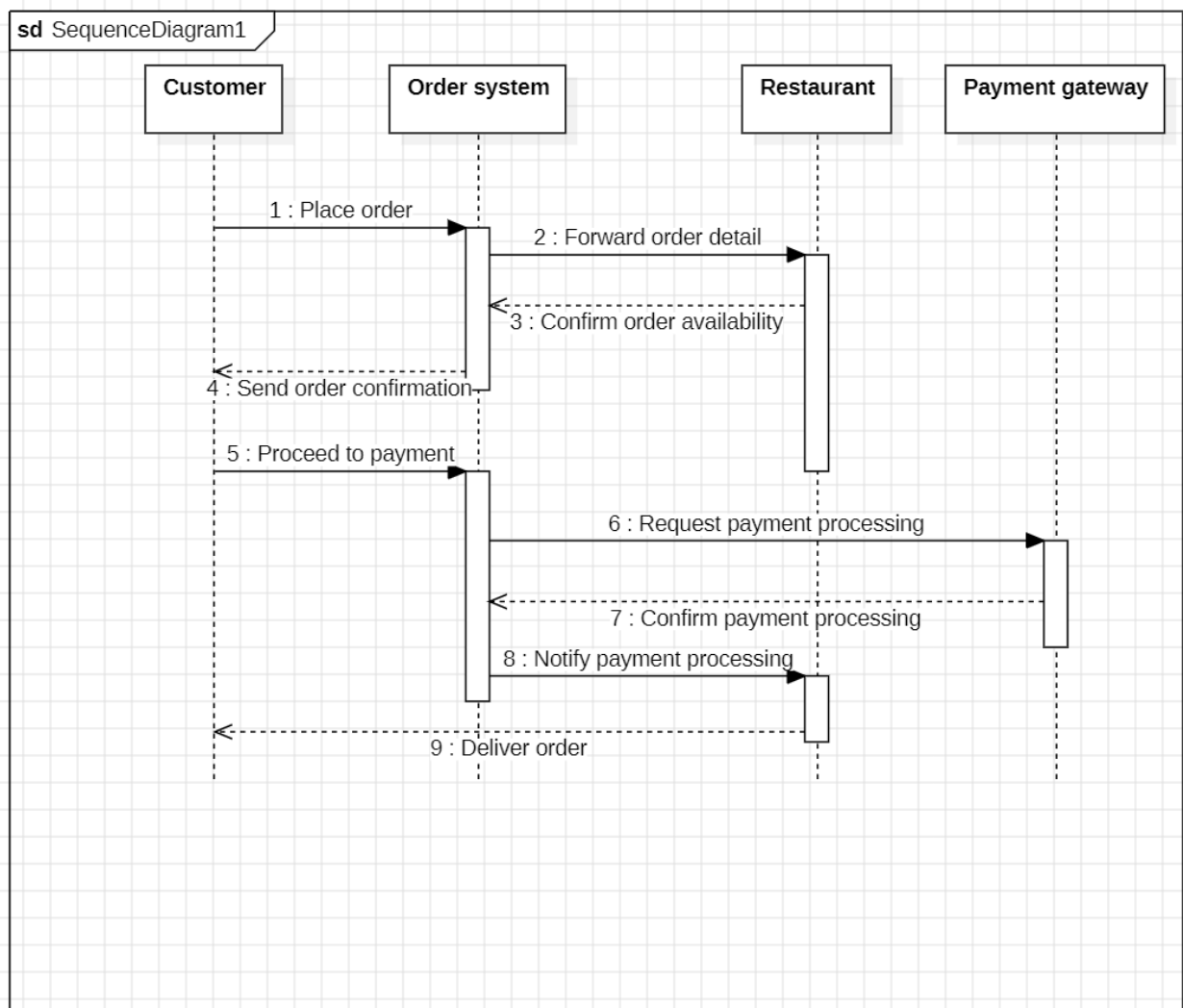
E) Object Diagram:

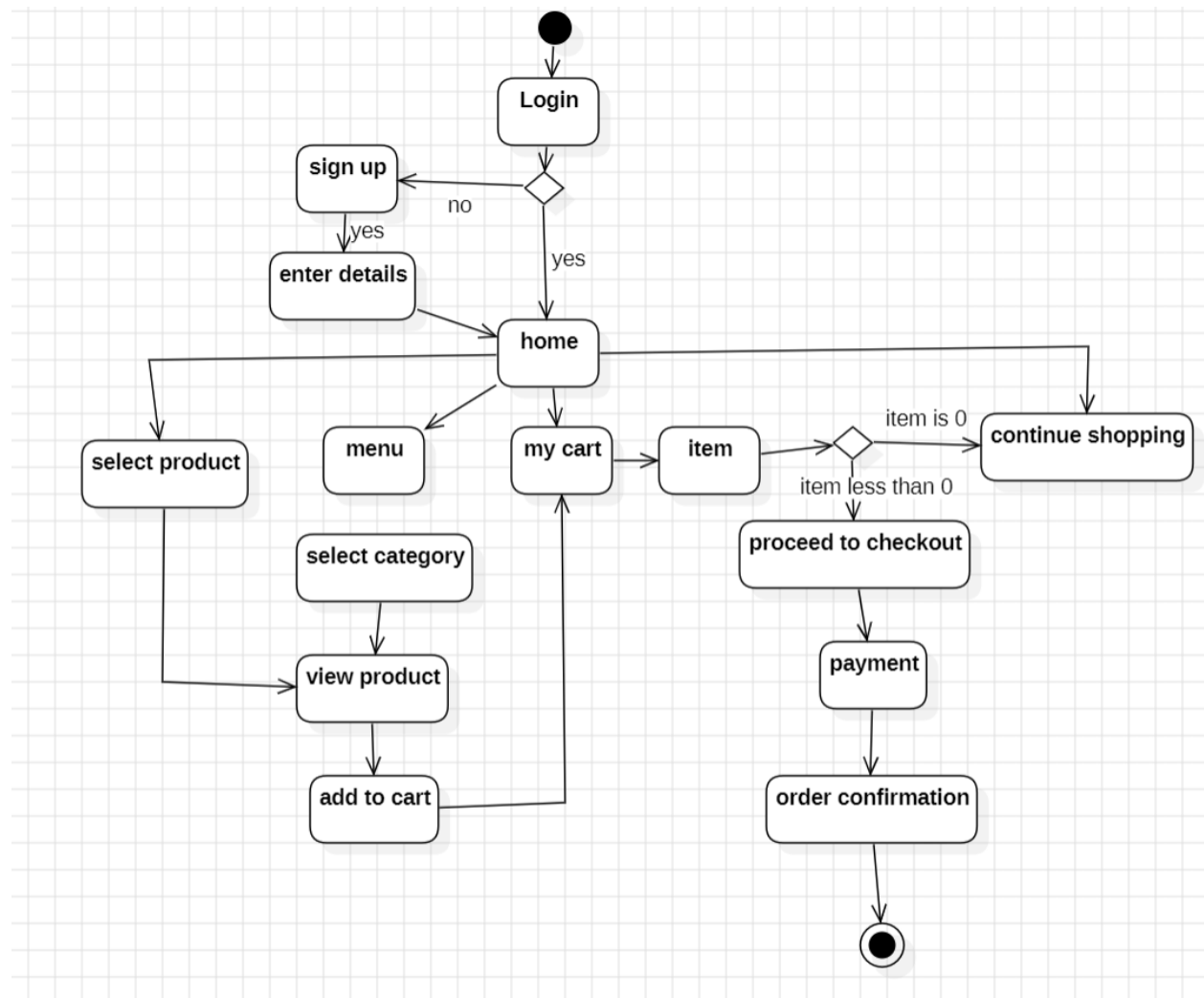
ONLINE FOOD ORDERING SYSTEM

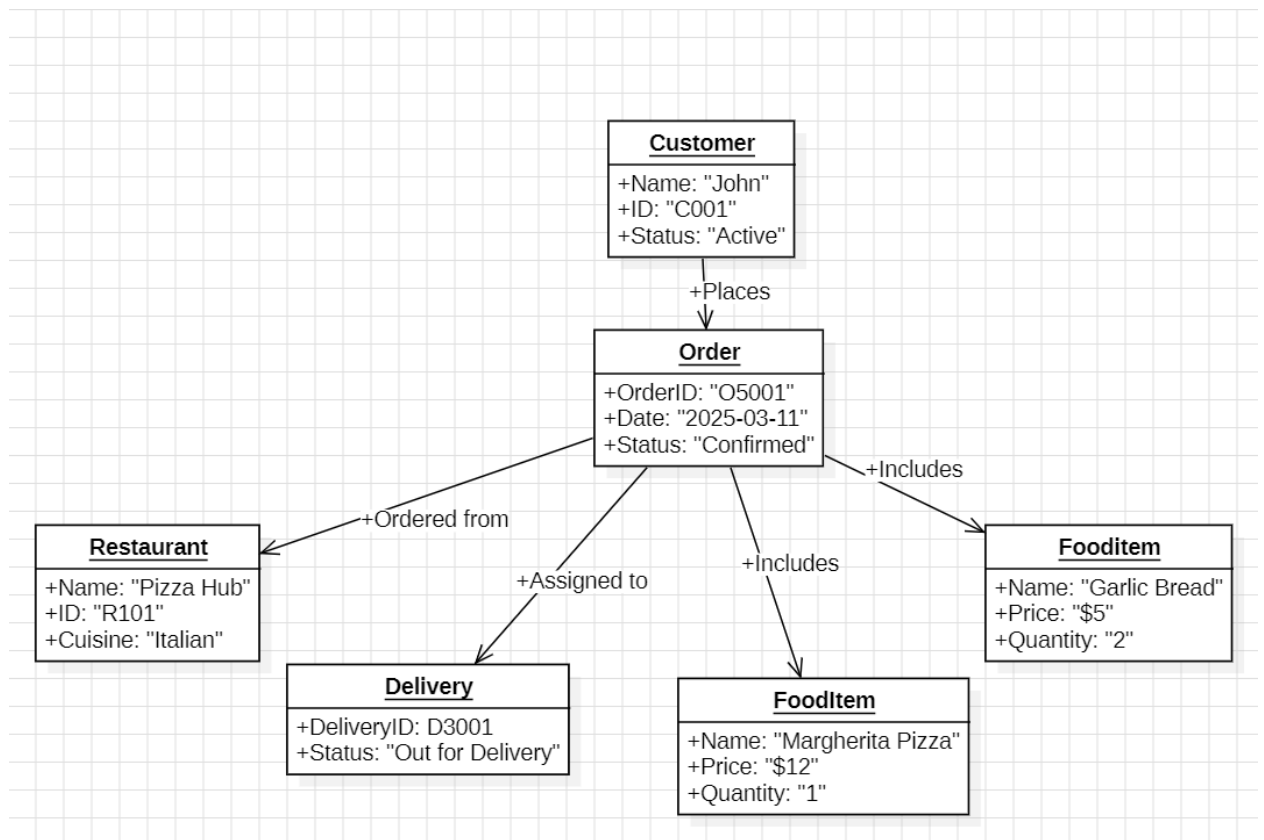
A) Use Case Diagram:



B) Class Diagram:

C) Sequence Diagram:

D) State Activity Diagram:

E) Object Diagram:

Basic Java Programs

A) Basic Calculator

CODE:

```
import java.util.Scanner;

public class Calculator {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

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

        double num1 = sc.nextDouble();

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

        double num2 = sc.nextDouble();

        System.out.println("Choose an operation (+, -, *, /): ");

        char operator = sc.next().charAt(0);

        double result = 0;

        switch (operator) {

            case '+':

                result = num1 + num2;

                break;

            case '-':

                result = num1 - num2;

                break;

            case '*':

                result = num1 * num2;

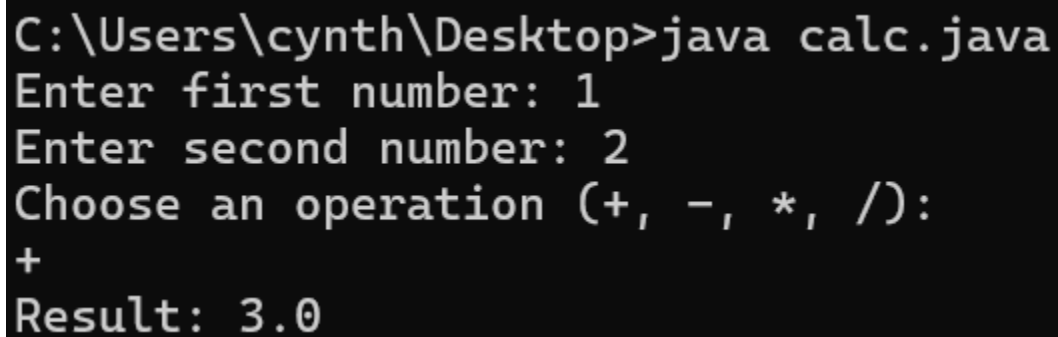
                break;

            case '/':

                result = num1 / num2;
```

```
        break;
    default:
        System.out.println("Invalid operator!");
        return;
    }
    System.out.println("Result: " + result);
}
}
```

OUTPUT



```
C:\Users\cynth\Desktop>java calc.java
Enter first number: 1
Enter second number: 2
Choose an operation (+, -, *, /):
+
Result: 3.0
```

B) Prime Number Check

CODE

```
import java.util.Scanner;

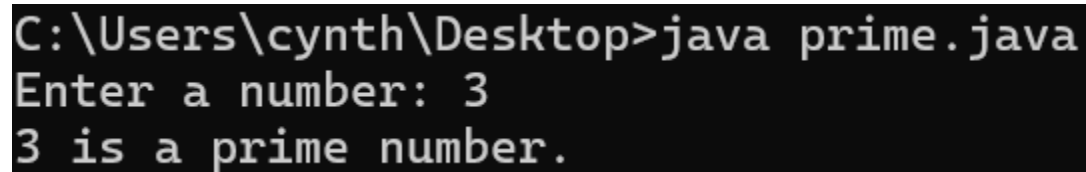
public class PrimeNumber {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        boolean isPrime = true;

        for (int i = 2; i <= num / 2; ++i) {
            if (num % i == 0) {
```

```
        isPrime = false;
        break;
    }
}

if (isPrime && num > 1) {
    System.out.println(num + " is a prime number.");
} else {
    System.out.println(num + " is not a prime number.");
}
}
```

OUTPUT



```
C:\Users\cynth\Desktop>java prime.java
Enter a number: 3
3 is a prime number.
```

C) Fibonacci Series

CODE

```
import java.util.Scanner;

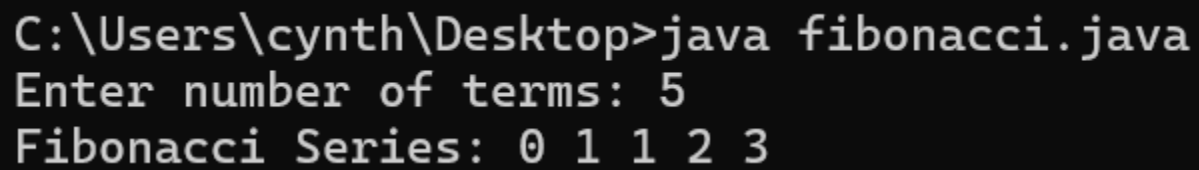
public class Fibonacci {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter number of terms: ");
        int n = sc.nextInt();
        int a = 0, b = 1, c;

        System.out.print("Fibonacci Series: " + a + " " + b);
```



```
        for (int i = 2; i < n; i++) {  
            c = a + b;  
            System.out.print(" " + c);  
            a = b;  
            b = c;  
        }  
    }  
}
```

OUTPUT



```
C:\Users\cynth\Desktop>java fibonacci.java  
Enter number of terms: 5  
Fibonacci Series: 0 1 1 2 3
```

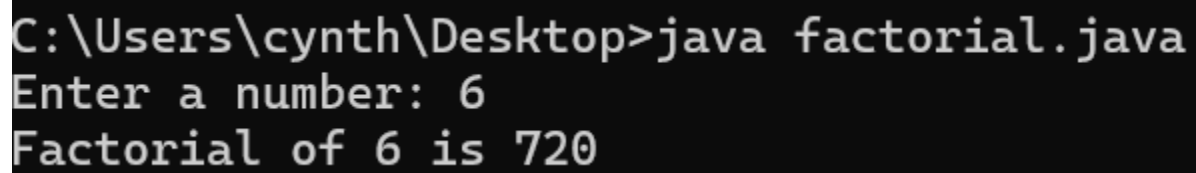
D) Factorial of a Number

CODE

```
import java.util.Scanner;  
  
public class Factorial {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.print("Enter a number: ");  
        int num = sc.nextInt();  
        long factorial = 1;  
  
        for (int i = 1; i <= num; i++) {  
            factorial *= i;  
        }  
    }  
}
```

```
        System.out.println("Factorial of " + num + " is " + factorial);
    }
}
```

OUTPUT



```
C:\Users\cynth\Desktop>java factorial.java
Enter a number: 6
Factorial of 6 is 720
```

E) PALINDROME

CODE

```
import java.util.Scanner;

public class Palindrome {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = sc.nextLine();
        String reversed = new StringBuilder(str).reverse().toString();

        if (str.equals(reversed)) {
            System.out.println(str + " is a palindrome.");
        } else {
            System.out.println(str + " is not a palindrome.");
        }
    }
}
```

OUTPUT

```
C:\Users\cynth\Desktop>java palindrome.java
Enter a string: mom
mom is a palindrome.
```

F) EVEN OR ODD NUMBER

CODE

```
import java.util.Scanner;

public class EvenOdd {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();

        if (num % 2 == 0) {
            System.out.println(num + " is an even number.");
        } else {
            System.out.println(num + " is an odd number.");
        }
    }
}
```

OUTPUT

```
C:\Users\cynth\Desktop>java evenodd.java
Enter a number: 5
5 is an odd number.
```

G) REVERSE OF A NUMBER

CODE

```
import java.util.Scanner;

public class ReverseNumber {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

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

        int num = sc.nextInt();

        int reversed = 0;

        while (num != 0) {

            int digit = num % 10;

            reversed = reversed * 10 + digit;

            num /= 10;

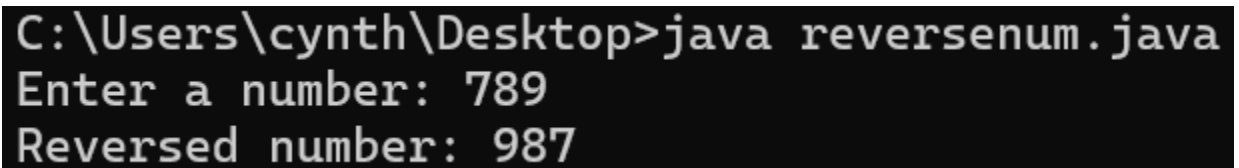
        }

        System.out.println("Reversed number: " + reversed);

    }

}
```

OUTPUT

A screenshot of a terminal window with a black background and white text. The prompt 'C:\Users\cynth\Desktop>' is followed by the command 'java reversenum.java'. The program then prompts 'Enter a number: ' and the user has entered '789'. The program outputs 'Reversed number: 987'.

```
C:\Users\cynth\Desktop>java reversenum.java
Enter a number: 789
Reversed number: 987
```

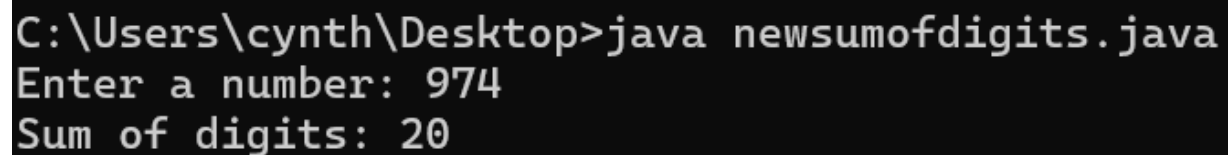
H) SUM OF DIGITS

CODE

```
import java.util.Scanner;
```

```
public class SumOfDigits {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.print("Enter a number: ");  
        int num = sc.nextInt();  
        int sum = 0;  
  
        while (num != 0) {  
            sum += num % 10;  
            num /= 10;  
        }  
  
        System.out.println("Sum of digits: " + sum);  
    }  
}
```

OUTPUT



```
C:\Users\cynth\Desktop>java newsumofdigits.java  
Enter a number: 974  
Sum of digits: 20
```

I) PRINT MULTIPLICATION TABLE

CODE

```
import java.util.Scanner;  
  
public class MultiplicationTable {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);
```

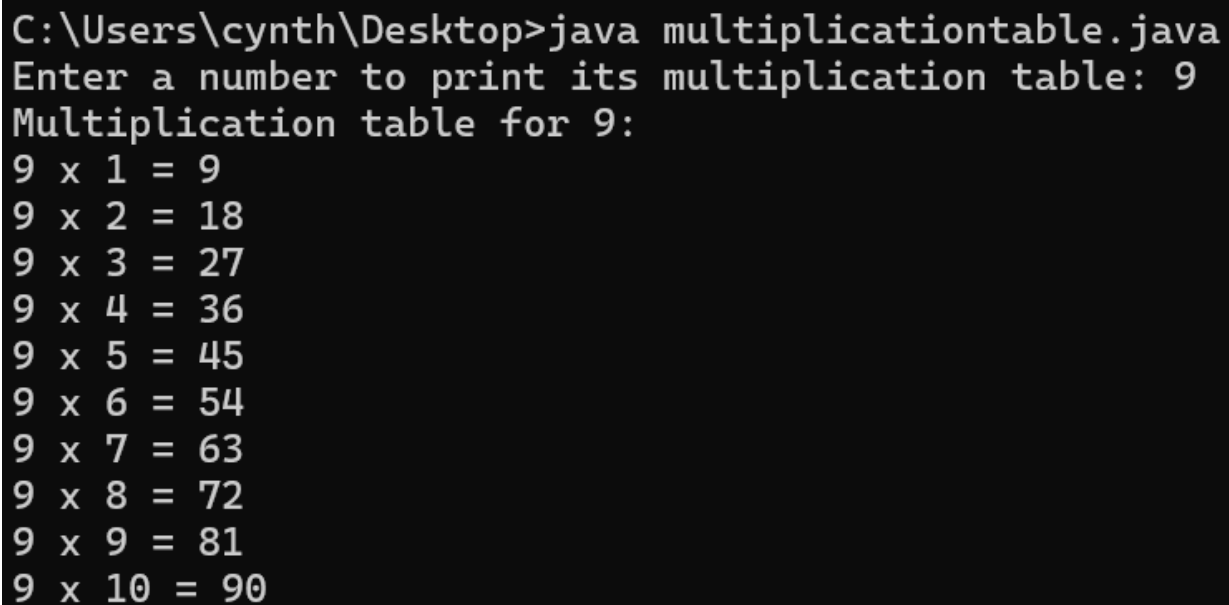
```
System.out.print("Enter a number to print its multiplication table: ");

int num = sc.nextInt();

System.out.println("Multiplication table for " + num + ":");

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

OUTPUT



```
C:\Users\cynth\Desktop>java multiplicationtable.java
Enter a number to print its multiplication table: 9
Multiplication table for 9:
9 x 1 = 9
9 x 2 = 18
9 x 3 = 27
9 x 4 = 36
9 x 5 = 45
9 x 6 = 54
9 x 7 = 63
9 x 8 = 72
9 x 9 = 81
9 x 10 = 90
```

J) MAXIMUM OF TWO NUMBERS

CODE

```
import java.util.Scanner;

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

```
Scanner sc = new Scanner(System.in);

System.out.print("Enter the first number: ");
int num1 = sc.nextInt();

System.out.print("Enter the second number: ");
int num2 = sc.nextInt();

if (num1 > num2) {
    System.out.println(num1 + " is the maximum.");
} else if (num2 > num1) {
    System.out.println(num2 + " is the maximum.");
} else {
    System.out.println("Both numbers are equal.");
}
}
```

OUTPUT

```
C:\Users\cynth\Desktop>java maxoftwo.java
Enter the first number: 45
Enter the second number: 78
78 is the maximum.
```