# LAB RECORD
# OBJECT ORIENTED PROGRAMMING
# (23CSE111)

*Submitted by*

SIDDHARTH S

CH.SC.U4CSE24145

**BACHELOR OF TECHNOLOGY**

IN

COMPUTER SCIENCE AND ENGINEERING

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF COMPUTING

CHENNAI

March - 2025

**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.U4CSE24145 – SIDDHARTH S*** 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 DIAGRAM | |
| 1. | **LIBRARY MANAGEMENT** | |
| | 1.a) Use Case Diagram | 4 |
| | 1.b) Class Diagram | 5 |
| | 1.c) Sequence Diagram | 5 |
| | 1.d) Statechart Diagram | 6 |
| | 1.e) Activity Diagram | 6 |
| 2. | **ATM- TRANSACTION** | |
| | 2.a) Use Case Diagram | 7 |
| | 2.b) Class Diagram | 8 |
| | 2.c) Sequence Diagram | 8 |
| | 2.d) Statechart Diagram | 9 |
| | 2.e) Component Diagram | 9 |
| 3. | **BASIC JAVA PROGRAMS** | |
| | 3.a) Armstrong Number | 10 |
| | 3.b) Sum of Even, Odd Digits | 11 |
| | 3.c) Factorial | 12 |
| | 3.d) Fibonacci Series | 13 |
| | 3.e) LCM Calculator | 14 |
| | 3.f) Number Pattern | 15 |
| | 3.g) Palindrome Check | 16 |
| | 3.h) Prime Checker | 17 |
| | 3.i) Reverse Number | 18 |
| | 3.j) Sum of Digits | 19 |
| 4. | SINGLE INHERITANCE | |
| | 4a. single inheritance 1 | |
| | 4b. single inheritance 2 | |

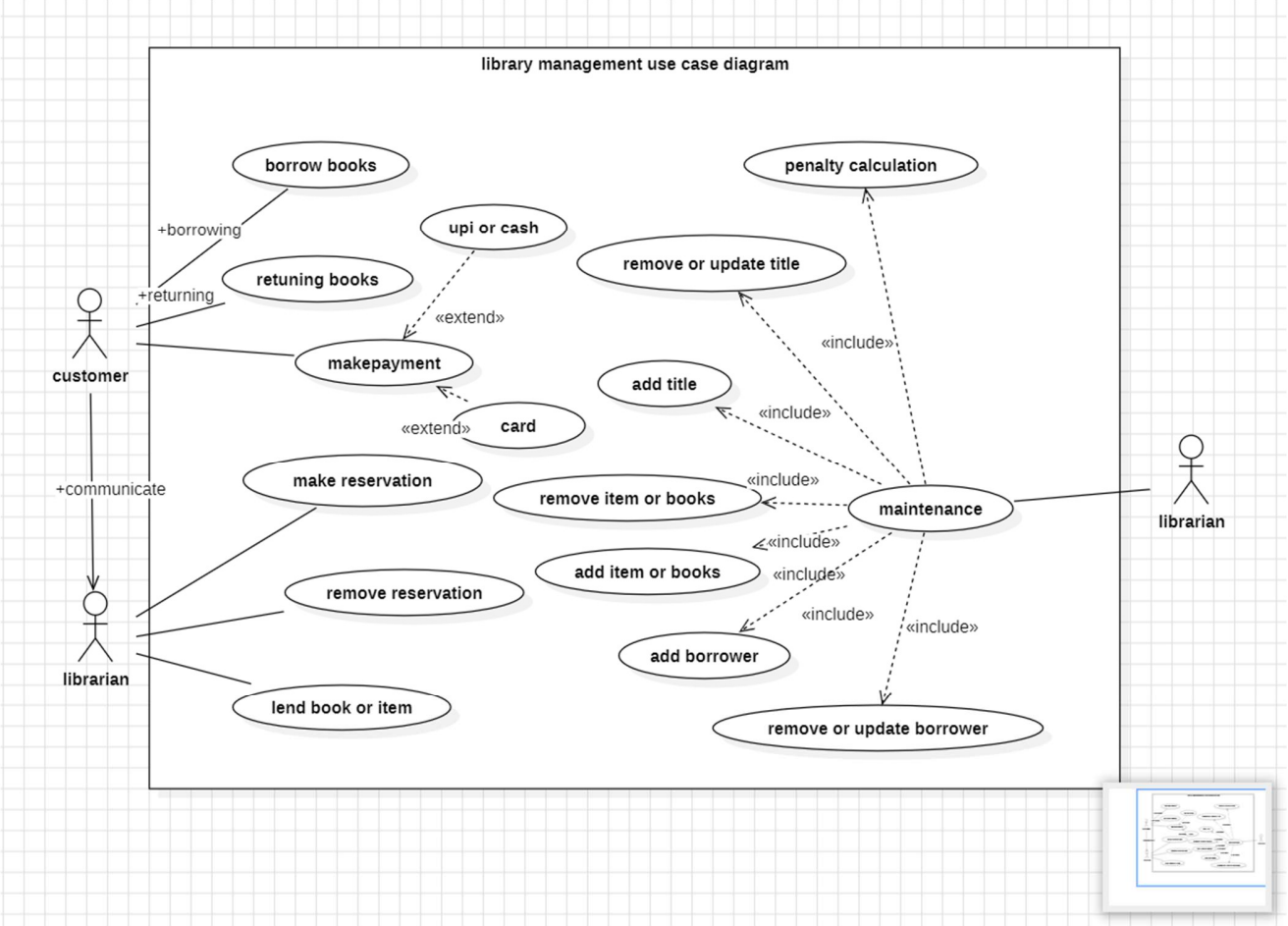
| | | |
|---|---|---|
| 5. | **MULTILEVEL INHERITANCE PROGRAMS** | |
| | 5.a) Multilevel 1 | |
| | 5.b) Multilevel 2 | |
| 6. | **HIERARCHICAL INHERITANCE PROGRAMS** | |
| | 6.a) Hierarchical 1 | |
| | 6.b) Hierarchical 2 | |
| 7. | **HYBRID INHERITANCE PROGRAMS** | |
| | 7.a) Hybrid 1 | |
| | 7.b) Hybrid 2 | |
| | **POLYMORPHISM** | |
| 8. | **CONSTRUCTOR PROGRAMS** | |
| | 8.a) Constructor 1 | |
| | 8.b) Constructor 2 | |
| 9. | **CONSTRUCTOR OVERLOADING PROGRAMS** | |
| | 9.a) Constructor overloading 1 | |
| | 9.b) Constructor overloading 2 | |
| 10. | **METHOD OVERLOADING PROGRAMS** | |
| | 10.a) Overloading 1 | |
| | 10.b) Overloading 2 | |
| 11. | **METHOD OVERRIDING PROGRAMS** | |
| | 11.a) Overriding 1 | |
| | 11.b) Overriding 2 | |
| | **ABSTRACTION** | |
| 12. | **INTERFACE PROGRAMS** | |
| | 12.a) Interface 1 | |
| | 12.b) Interface 2 | |
| | 12.c) Interface 3 | |
| | 12.d) Interface 4 | |
| 13. | **ABSTRACT CLASS PROGRAMS** | |
| | 13.a) Abstract class 1 | |
| | 13.b) Abstract class 2 | |
| | 13.c) Abstract class 3 | |
| | 13.d) Abstract class 4 | |
| | **ENCAPSULATION** | |
| 14. | **ENCAPSULATION PROGRAMS** | |
| | 14.a) Encapsulation 1 | |
| | 14.b) Encapsulation 2 | |
| | 14.c) Encapsulation 3 | |
| | 14.d) Encapsulation 4 | |
| 15. | **PACKAGES PROGRAMS** | |
| | 15.a) User-Defined 1 | |
| | 15.b) User-Defined 2 | |

| | | |
|---|---|---|
| | 15.c) Built-in 1 | |
| | 15.d) Built-in 2 | |
| 16. | **EXCEPTION HANDLING PROGRAMS** | |
| | 16.a) Exception 1 | |
| | 16.b) Exception 2 | |
| | 16.c) Exception 3 | |
| | 16.d) Exception 4 | |
| 17. | **FILE HANDLING PROGRAMS** | |
| | 17.a) File 1 | |
| | 17.b) File 2 | |
| | 17.c) File 3 | |
| | 17.d) File 4 | |

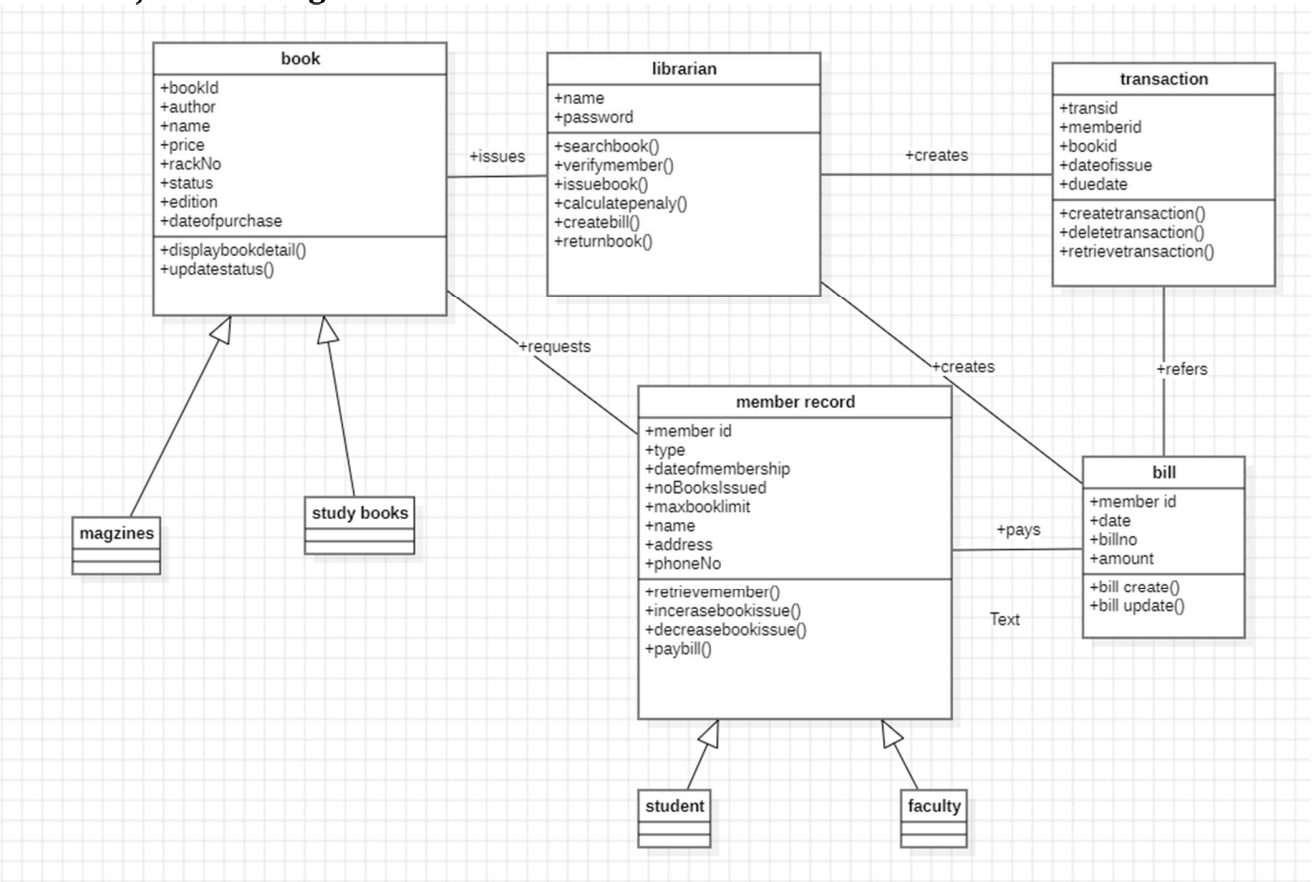
# UML DIAGRAMS

## 1. LIBRARY MANAGEMENT SYSTEM

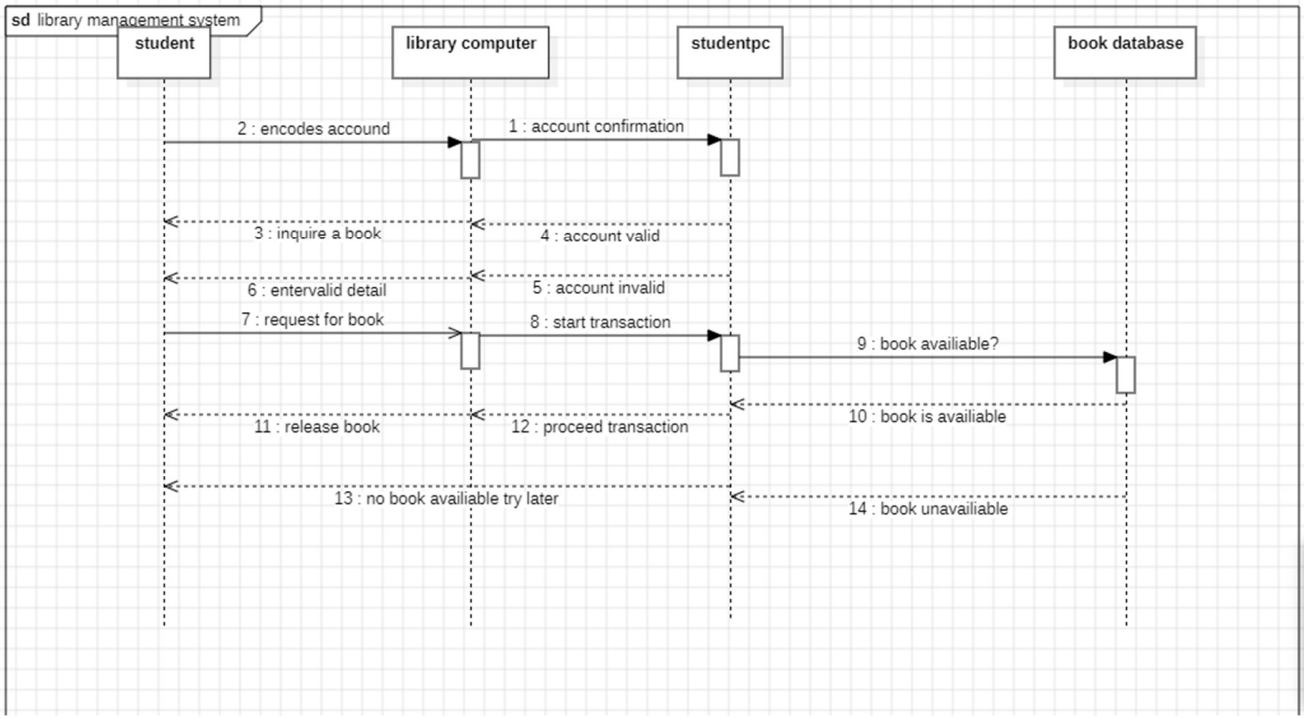
### 1.a) Use Case Diagram:



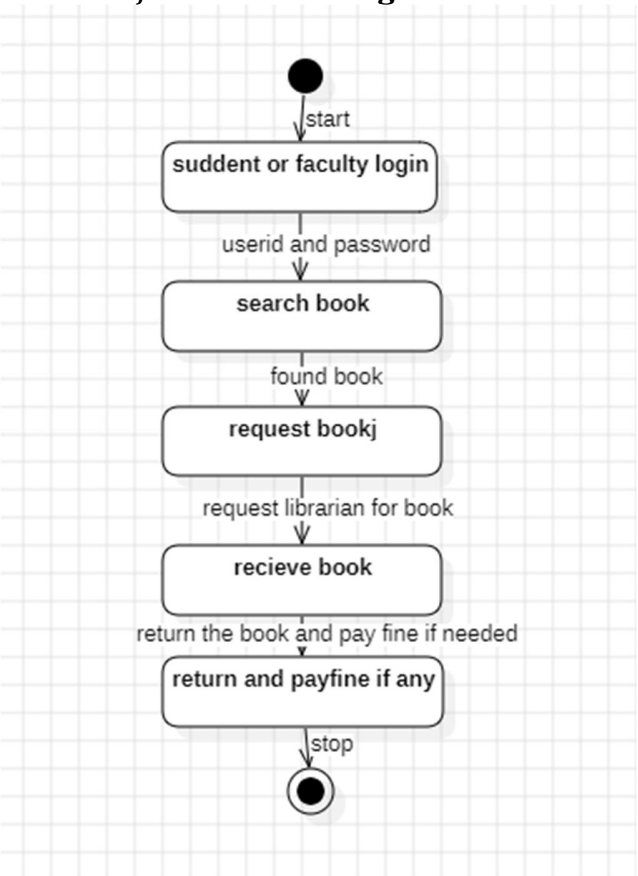
## 1.b) Class Diagram:



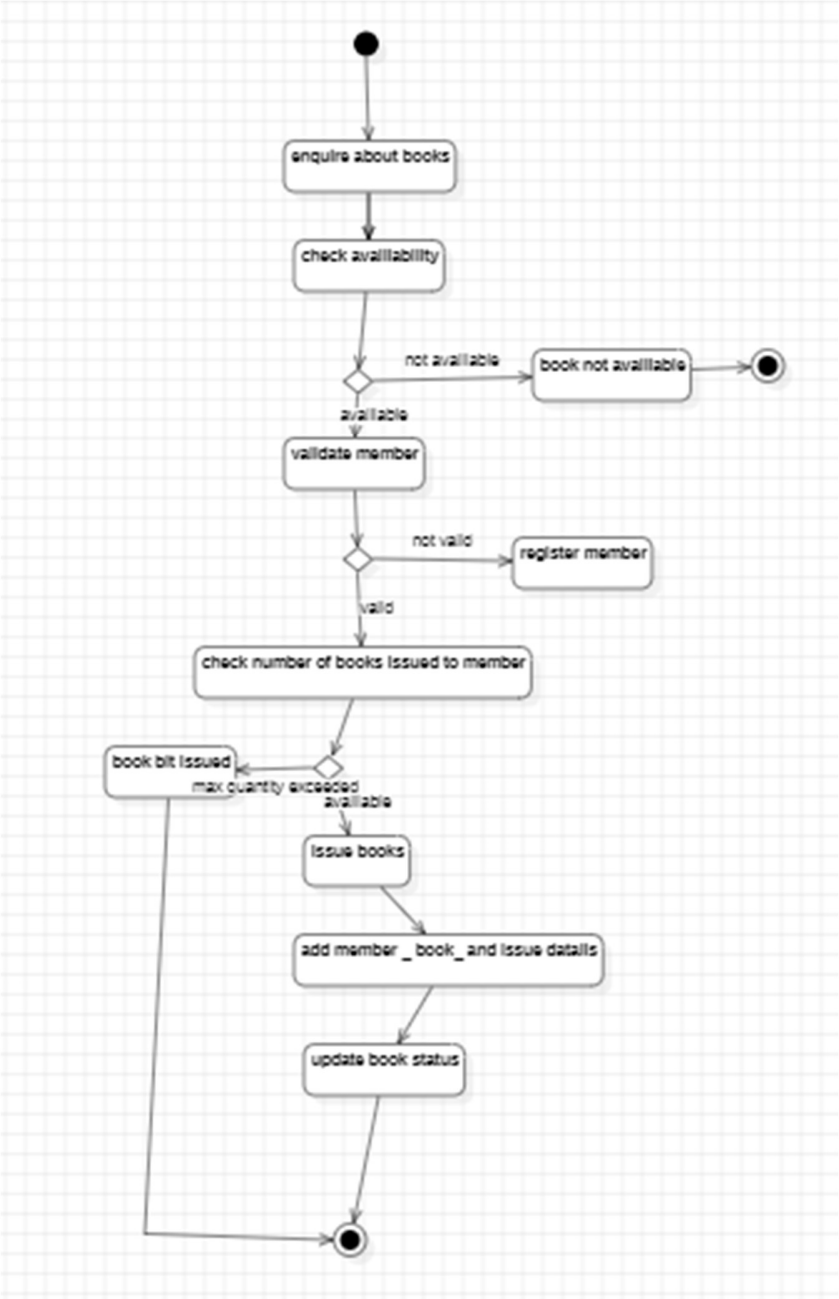

## 1.c) Sequence Diagram:



## 1.d) Statechart diagram:

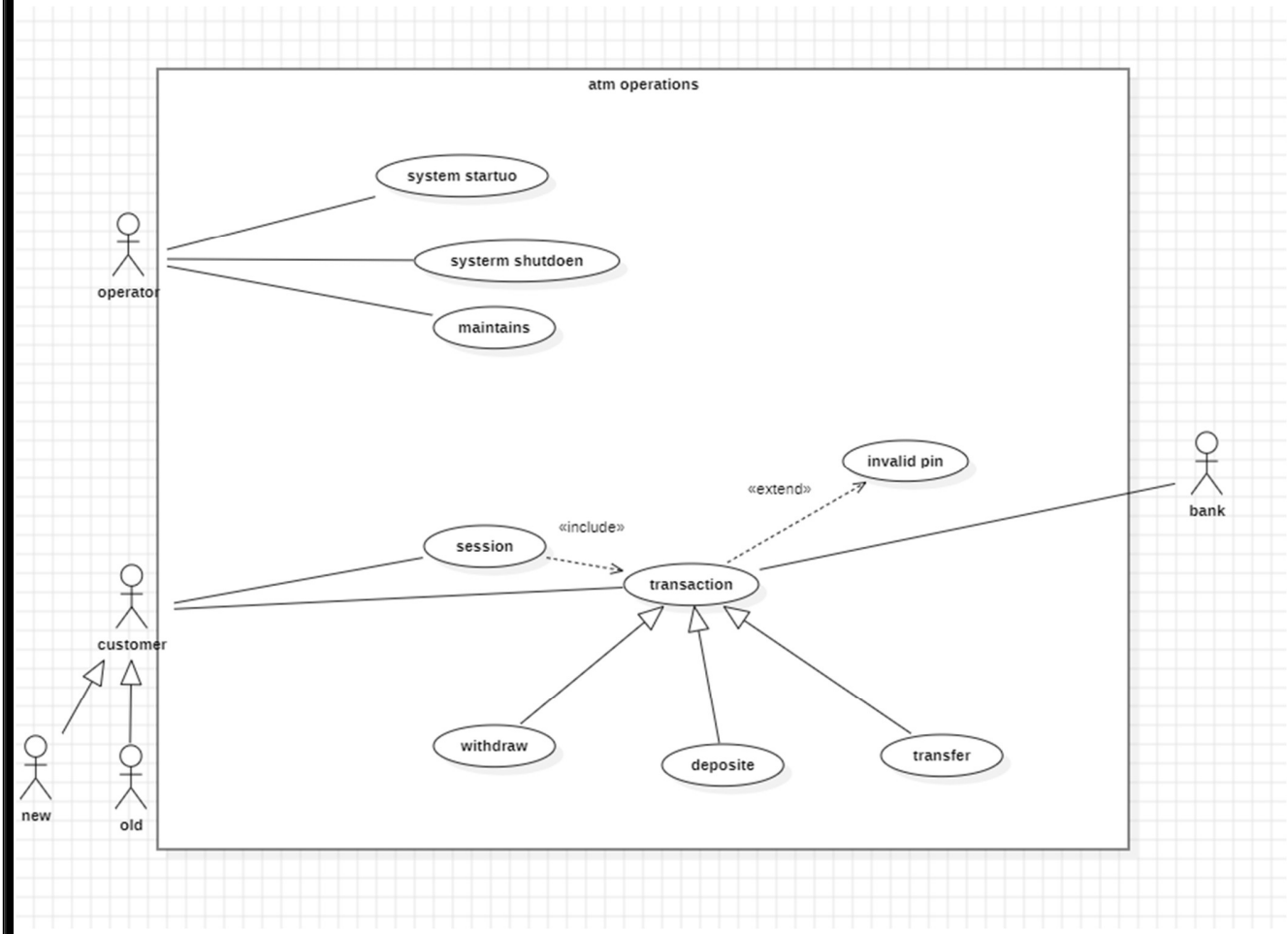


## 1.e) Activity Diagram:

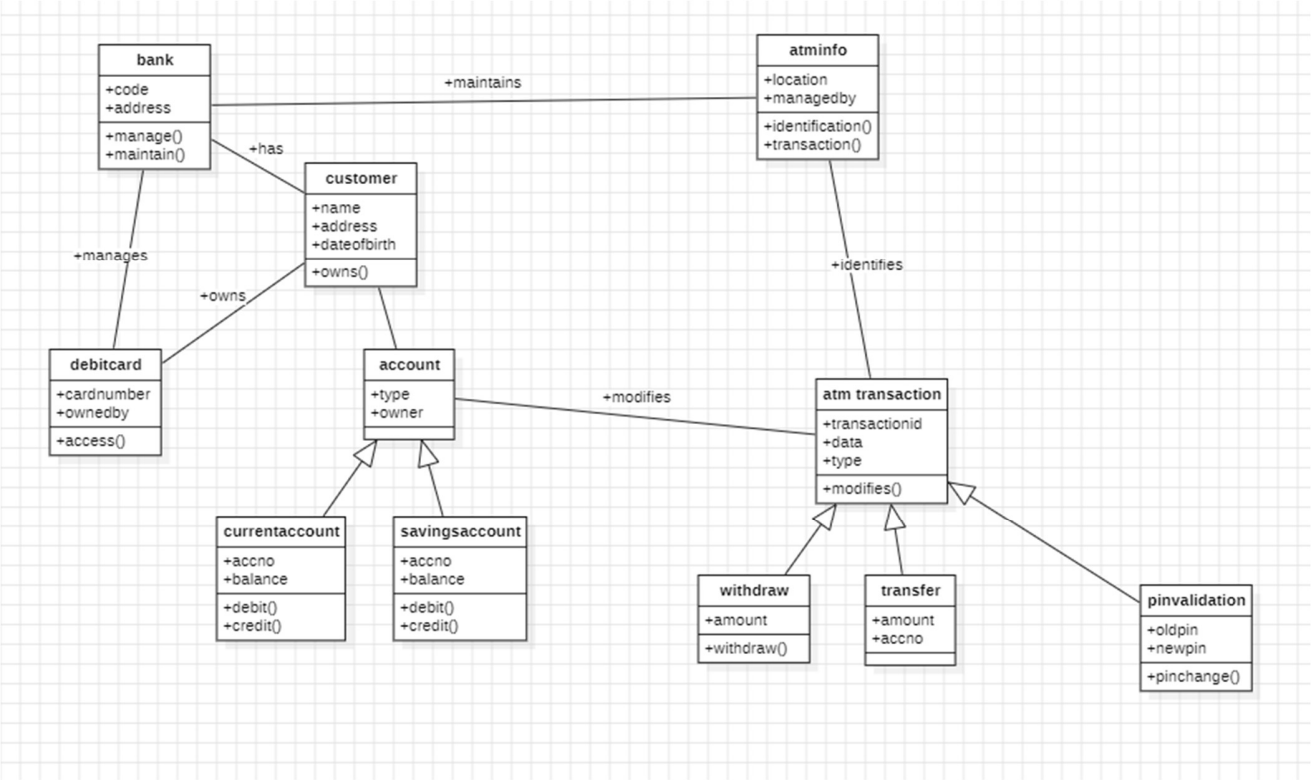


# 2. ATM TRANSACTION

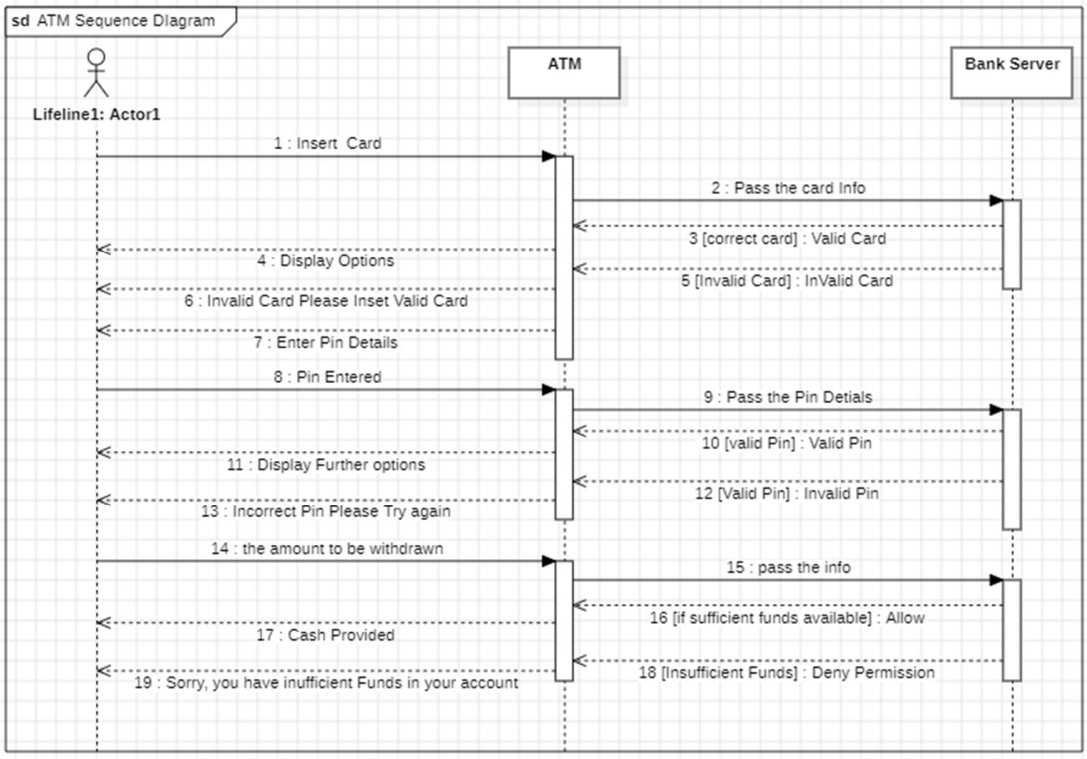
## 2.a) Use Case Diagram:



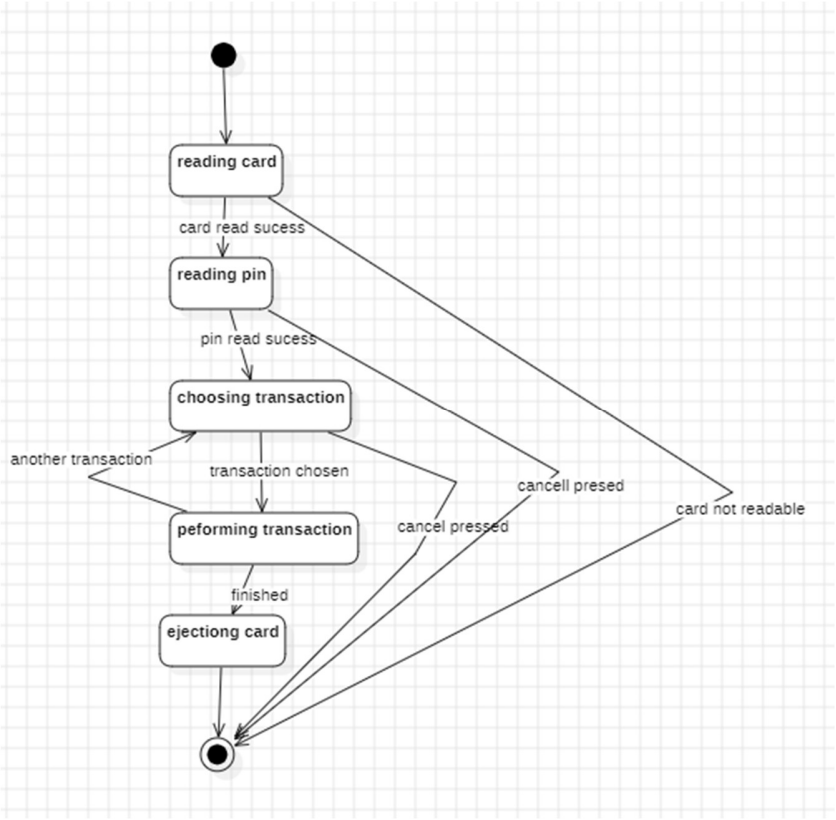
## 2.b) Class Diagram:



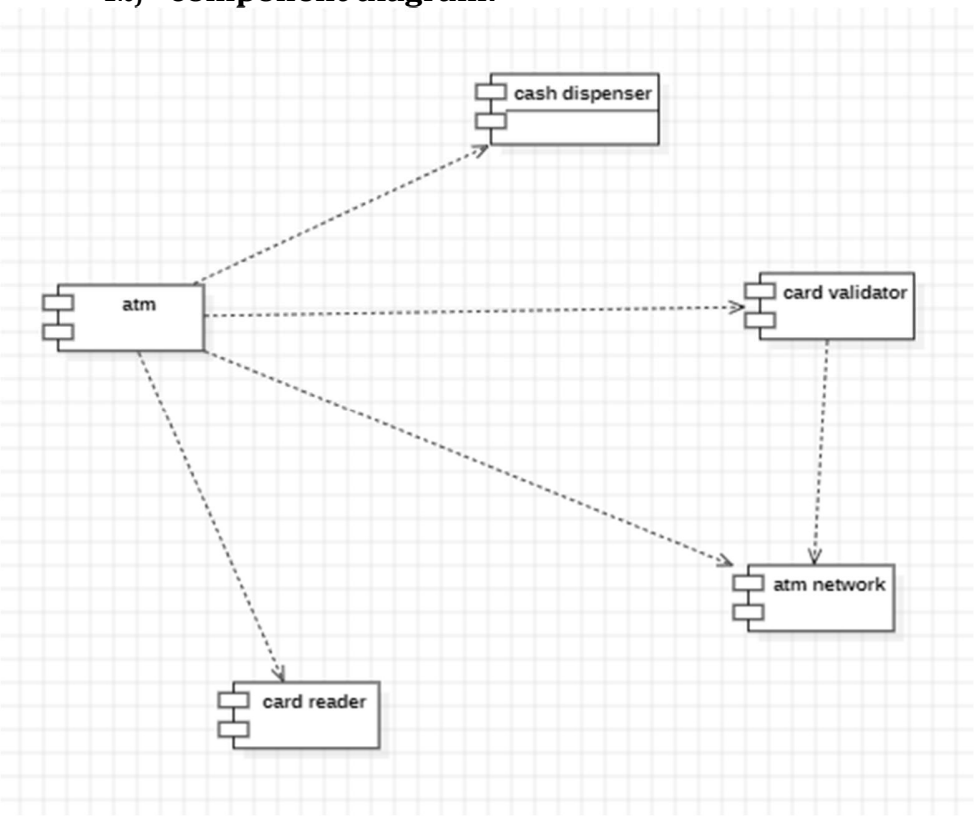

## 2.c) Sequence Diagram:



## 2.d) statechart:




## 2.e) component diagram:



# 3. Basic Java Programs

## 3.a) Armstrong Number:

**Code:**

```java
import java.util.Scanner;
public class armstrong_number
{
int num;
public armstrong_number(int num)
{
this.num = num;
}
public void operation()
{
int originalNum = num;
int sum = 0;
int digits = String.valueOf(num).length();
while (num != 0)
{
int digit = num % 10;
sum += Math.pow(digit, digits);
num /= 10;
}
if (sum == originalNum)
{
System.out.println(originalNum + " is an Armstrong number.");
}
else
{
System.out.println(originalNum + " is not an Armstrong number.");
}
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number: ");
int inputNum = inp.nextInt();
armstrong_number obj = new armstrong_number(inputNum);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> javac "C:\Users\Siddharth\Desktop\college\OOPS\armstr
ong_number.java"
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\armstro
ng_number.java"
Enter the number:
12
12 is not an Armstrong number.
PS C:\Users\Siddharth>
```

## 3.b) Sum of Even, Odd Digits:

**Code:**

```java
import java.util.Scanner;
public class even_odd_sum
{
int limit;
public even_odd_sum(int limit)
{
this.limit = limit;
}
public void operation()
{
int evenSum = 0, oddSum = 0;
for (int i = 1; i <= limit; i++)
{
if (i % 2 == 0)
{
evenSum += i;
}
else
{
oddSum += i;
}
}
System.out.println("Sum of even numbers up to " + limit + " is: " + evenSum);
System.out.println("Sum of odd numbers up to " + limit + " is: " + oddSum);
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the limit: ");
int inputLimit = inp.nextInt();
even_odd_sum obj = new even_odd_sum(inputLimit);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\even_od
d_sum[1].java"
Enter the limit:
25
Sum of even numbers up to 25 is: 156
Sum of odd numbers up to 25 is: 169
```

## 3.c) Factorial:

**Code:**

```java
import java.util.Scanner;
public class factorial_number
{
int num;
public factorial_number(int num)
{
this.num = num;
}
public void operation()
{
long factorial = 1;
for (int i = 1; i <= num; i++)
{
factorial *= i; // Multiply factorial by i
}
System.out.println("Factorial of " + num + " is: " + factorial);
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number: ");
int inputNum = inp.nextInt();
factorial_number obj = new factorial_number(inputNum);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\factor:
al_number.java"
Enter the number:
4
Factorial of 4 is: 24
PS C:\Users\Siddharth>
```

## 3.d) Fibonacci Series:

**Code:**

```java
import java.util.Scanner;
public class fibonacci_series
{
int limit;
public fibonacci_series(int limit)
{
this.limit = limit;
}
public void operation()
{
int num1 = 0, num2 = 1;
System.out.println("Fibonacci Series up to " + limit + " terms:");
for (int i = 1; i <= limit; i++)
{
System.out.print(num1 + " ");
int nextNum = num1 + num2;
num1 = num2;
num2 = nextNum;
}
System.out.println();
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number of terms for the Fibonacci series: ");
int inputLimit = inp.nextInt();
fibonacci_series obj = new fibonacci_series(inputLimit);
obj.operation();
}
}
```

**Output;**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\fibonac
ci_series.java"
Enter the number of terms for the Fibonacci series:
5
Fibonacci Series up to 5 terms:
0 1 1 2 3
```

## 3.e) LCM Calculator:

**Code:**

```java
import java.util.Scanner;
public class lcm_numbers
{
int num1, num2;
public lcm_numbers(int num1, int num2)
{
this.num1 = num1;
this.num2 = num2;
}
public void operation()
{
int lcm = (num1 > num2) ? num1 : num2;
while (true)
{
if (lcm % num1 == 0 && lcm % num2 == 0)
{
System.out.println("LCM of " + num1 + " and " + num2 + " is: " + lcm);
break;
}
lcm++;
}
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the first number: ");
int inputNum1 = inp.nextInt();
System.out.println("Enter the second number: ");
int inputNum2 = inp.nextInt();
lcm_numbers obj = new lcm_numbers(inputNum1, inputNum2);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\lcm_num
bers.java"
Enter the first number:
2
Enter the second number:
8
LCM of 2 and 8 is: 8
PS C:\Users\Siddharth>
```

## 3.f) Number Triangle:

**Code:**

```
import java.util.Scanner;
public class lower_triangle
{
int rows;
public lower_triangle(int rows)
{
this.rows = rows;
}
public void operation()
{
for (int i = 1; i <= rows; i++)
{
for (int j = 1; j <= i; j++)
{
System.out.print(j + " ");
}
System.out.println();
}
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number of rows: ");
int inputRows = inp.nextInt();
lower_triangle obj = new lower_triangle(inputRows);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\lower_triangle.java"
Enter the number of rows:
3
1
1 2
1 2 3
PS C:\Users\Siddharth>
```

## 3.g) Palindrome Check:

**Code:**

```java
import java.util.Scanner;
public class palindrome_number
{
int num;
int reversedNum = 0;
public palindrome_number(int num)
{
this.num = num;
}
public void operation()
{
int originalNum = num;
while (num != 0)
{
int digit = num % 10;
reversedNum = reversedNum * 10 + digit;
num = num / 10; // Remove the last digit
}
if (originalNum == reversedNum)
{
System.out.println(originalNum + " is a palindrome number.");
}
else
{
System.out.println(originalNum + " is not a palindrome number.");
}
}
public static void main(String[] args)
{
Scanner inp = new Scanner(System.in);
System.out.println("Enter the number: ");
int inputNum = inp.nextInt();
palindrome_number obj = new palindrome_number(inputNum);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\palindr
ome_number.java"
Enter the number:
12321
12321 is a palindrome number.
PS C:\Users\Siddharth>
```

## 3.h) Prime Checker:

**Code:**

```
import java.util.Scanner;
public class prime_number
{
int num;
Boolean isPrime=true;
public prime_number(int num)
{
this.num=num;
}
public void operation()
{
if (num<=1)
{
isPrime=false;
}
else
{
for (int i=2;i*i<=num;i++)
{
if (num % i == 0)
{
isPrime=false;
break;
}
}
}
if (isPrime)
{
System.out.println(num + " is a prime number");
}
else
{
System.out.println(num + " is not a prime number");
}
}
public static void main (String[] args)
{
Scanner inp= new Scanner(System.in);
System.out.println("Enter the number : ");
int inputNum=inp.nextInt();
prime_number obj=new prime_number(inputNum);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\prime_n
umber.java"
Enter the number :
29
29 is a prime number
PS C:\Users\Siddharth>
```

## 3.i) Reverse Number:

**Code:**

```
import java.util.Scanner;
public class reverse_number
{
int num,rev=0,digit;
public reverse_number(int num)
{
this.num=num;
}
public void operation()
{
while (num != 0)
{
digit=num%10;
rev=rev*10+digit;
num=num/10;
}
System.out.println("Reversed number : " + rev);
}
public static void main (String[] args)
{
Scanner inp= new Scanner(System.in);
System.out.println("Enter the number : ");
int inputNum=inp.nextInt();
reverse_number obj=new reverse_number(inputNum);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\reverse
_number.java"
Enter the number :
1234
Reversed number : 4321
PS C:\Users\Siddharth>
```

## 3.j) Sum of Digits:

**Code:**

```
import java.util.Scanner;
public class sum_of_digits
{
int num,sum=0;
public sum_of_digits(int num)
{
this.num=num;
}
public void operation()
{
while (num != 0)
{
sum=sum+num%10;
num=num/10;
}
System.out.println("Sum of digits : " + sum);
}
public static void main (String[] args)
{
Scanner inp= new Scanner(System.in);
System.out.println("Enter the number : ");
int inputNum=inp.nextInt();
sum_of_digits obj=new sum_of_digits(inputNum);
obj.operation();
}
}
```

**Output:**

```
PS C:\Users\Siddharth> java "C:\Users\Siddharth\Desktop\college\OOPS\sum_of_digits[1].java"
Enter the number :
123
Sum of digits : 6
PS C:\Users\Siddharth>
```

**INHERITANCE**

**4. SINGLE LEVEL INHERITANCE**

**4.a)**

```
class Employee {
String name;
int empId;

public Employee(String name, int empId) {
this.name = name;
this.empId = empId;
}

public void displayDetails() {
System.out.println("Employee Name: " + name);
System.out.println("Employee ID: " + empId);
}
}


class Manager extends Employee {
String department;

public Manager(String name, int empId, String department) {
super(name, empId);
this.department = department;
}

public void displayManagerDetails() {
displayDetails();
System.out.println("Department: " + department);
}
}
public class SingleInheritanceExample1 {
Run | Debug
public static void main(String[] args) {
Manager mgr = new Manager(name:"LEO DAS", empId:7, department:"Sales");
mgr.displayManagerDetails();
}
}
```

**Output:**

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp
' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8c087069\redhat.java\jdt_ws\jdt.ls-java-project\bin' 'Sin
gleInheritanceExample1'
Employee Name: LEO DAS
Employee ID: 7
Department: Sales
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**4b]**

```java
class Account {
    String holderName;
    double currentAmount;

    Account(String holderName, double currentAmount) {
        this.holderName = holderName;
        this.currentAmount = currentAmount;
    }

    void displayAmount() {
        System.out.println("Holder Name: " + holderName);
        System.out.println("Current Amount: $" + currentAmount);
    }
}

class FixedDepositAccount extends Account {
    double rateOfInterest = 5.0;

    FixedDepositAccount(String holderName, double currentAmount) {
        super(holderName, currentAmount);
    }

    void computeInterest() {
        double interest = (currentAmount * rateOfInterest) / 100;
        System.out.println("Interest Earned: $" + interest);
    }
}

public class singleinheritance2 {
    Run | Debug
    public static void main(String[] args) {
        FixedDepositAccount acc = new FixedDepositAccount(holderName:"Dada Hussain", currentAmount:10000);
        acc.computeInterest();
        void FixedDepositAccount.computeInterest()
        acc.computeInterest();
    }
}
```

**Output:**

```
C:\Users\Siddharth>javac "C:\Users\Siddharth\Desktop\college\OOPS\lab work\singleinheritance2.java"

C:\Users\Siddharth>java "C:\Users\Siddharth\Desktop\college\OOPS\lab work\singleinheritance2.java"
Holder Name: Dada Hussain
Current Amount: $10000.0
Interest Earned: $500.0
```

**5. MULTILEVEL INHERITANCE**

**5.a)**

```java
class A {
    String a;
    A(String a) {
        this.a = a;
    }
    void x() {
        System.out.println("Person Name: " + a);
    }
}

class B extends A {
    int b;
    double c;
    B(String a, int b, double c) {
        super(a);
        this.b = b;
        this.c = c;
    }
    void y() {
        System.out.println("Employee ID: " + b);
        System.out.println("Salary: $" + c);
    }
}

class C extends B {
    String d;
    C(String a, int b, double c, String d) {
        super(a, b, c);
        this.d = d;
    }
    void z() {
        System.out.println("Department: " + d);
        System.out.println("Role: Manager");
    }
}

public class multilevelinheritance1 {
    public static void main(String[] args) {
        C obj = new C("Bob", 1022, 92000, "Sales");
        obj.x();
        obj.y();
        obj.z();
    }
}
```

**Output:**

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>java multilevelinheritance1
Person Name: Bob
Employee ID: 1022
Salary: $92000.0
Department: Sales
Role: Manager
```

**5b]**

```java
class X {
    String a;
    X(String a) {
        this.a = a;
    }
    void m() {
        System.out.println("Vehicle Type: " + a);
    }
}

class Y extends X {
    String b;
    int c;
    Y(String a, String b, int c) {
        super(a);
        this.b = b;
        this.c = c;
    }
    void n() {
        System.out.println("Car Model: " + b);
        System.out.println("Year: " + c);
    }
}

class Z extends Y {
    int d;
    Z(String a, String b, int c, int d) {
        super(a, b, c);
        this.d = d;
    }
    void o() {
        System.out.println("Battery Capacity: " + d + " kWh");
        System.out.println("Type: Electric Car");
    }
}

public class multilevelinheritance2 {
    public static void main(String[] args) {
        Z obj = new Z("Four Wheeler", "BYD", 2023, 85);
        obj.m();
        obj.n();
        obj.o();
    }
}
```

**Output:**

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:51988' '-XX:+ShowCodeDetailsInEx
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8c087069\redhat.java\jdt_ws\lab work_df56f04e\bin' 'MultilevelInheritance3'
Vehicle Type: Four Wheeler
Car Model: BYD
Year: 2023
Battery Capacity: 85 kWh
Type: Electric Car
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

6.HIERARCHICAL INHERITANCE

6a]

```
class A {
    String a;
    A(String a) {
    this.a = a;
    }
    void x() {
    System.out.println("Animal: " + a);
    }
    }

    class B extends A {
    B(String a) {
    super(a);
    }
    void y() {
    System.out.println(x:"Sound: Bark");
    }
    }

    class C extends A {
    C(String a) {
    super(a);
    }
    void z() {
    System.out.println(x:"Sound: Meow");
    }
    }

    public class Hierarcalinheritance1 {
    Run | Debug
    public static void main(String[] args) {
    B d = new B(a:"Dog");
    C c = new C(a:"Cat");
    d.x();
    d.y();
    c.x();
    c.z();
    }
    }
```

**Output:**

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:j
dwp=transport=dt_socket,server=n,suspend=y,address=localhost:52275' '-XX:+ShowCodeDetailsInExceptionMessages' '-
cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8c087069\redhat.java\
jdt_ws\lab work_df56f04e\bin' 'Hierarcalinheritance1'
Animal: Dog
Sound: Bark
Animal: Cat
Sound: Meow
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**6b]**

```java
class M {
    String a;
    M(String a) {
    this.a = a;
    }
    void f() {
    System.out.println("Device: " + a);
    }
    }

    class N extends M {
    N(String a) {
    super(a);
    }
    void g() {
    System.out.println(x:"RAM: 16GB");
    }
    }

    class O extends M {
    O(String a) {
    super(a);
    }
    void h() {
    System.out.println(x:"Storage: 128GB");
    }
    }

    public class hierarcalinheritance2 {
    Run | Debug
    public static void main(String[] args) {
    N l = new N(a:"Laptop");
    O m = new O(a:"Mobile");
    l.f();
    l.g();
    m.f();
    m.h();
    }
    }
```

**Output:**

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>java "C:\Users\Sid
ance2.java"
Device: Laptop
RAM: 16GB
Device: Mobile
Storage: 128GB
```

# HYBRID INHERITANCE

7a]

```
class A {
    String a;
    A(String a) {
    this.a = a;
    }
    void p() {
    System.out.println("Name: " + a);
    }
    }

    class B extends A {
    int b;
    B(String a, int b) {
    super(a);
    this.b = b;
    }
    void q() {
    System.out.println("ID: " + b);
    }
    }

    class C {
    double c;
    C(double c) {
    this.c = c;
    }
    void r() {
    System.out.println("Score: " + c);
    }
    }

    class D extends B {
    C obj;
    D(String a, int b, double c) {
    super(a, b);
    obj = new C(c);
    }
    void s() {
    p();
    q();
    obj.r();
    System.out.println("Category: Final");
    }
    }

    public class hybridinheritance1 {
    public static void main(String[] args) {
    D x = new D("Siddharth", 505, 91.7);
    x.s();
    }
    }
```

Outpu:

```
C:\Users\Siddharth\Desktop\college\OOPS\lab w
k\hybridinheritance1.java"
Name: Siddharth
ID: 505
Score: 91.7
Category: Final
```

7b]

```java
class X {
    String a;
    X(String a) {
    this.a = a;
    }
    void m() {
    System.out.println("Appliance: " + a);
    }
    }

    class Y extends X {
    int b;
    Y(String a, int b) {
    super(a);
    this.b = b;
    }
    void n() {
    System.out.println("Capacity: " + b + " kg");
    }
    }

    class Z {
    String c;
    Z(String c) {
    this.c = c;
    }
    void o() {
    System.out.println("Feature: " + c);
    }
    }

    class W extends Y {
    Z obj;
    W(String a, int b, String c) {
    super(a, b);
    obj = new Z(c);
    }
    void p() {
    m();
    n();
    obj.o();
    System.out.println("Type: Smart Washing Machine");
    }
    }

    public class hybridinheritance2 {
    public static void main(String[] args) {
    W w = new W("WasherX", 7, "Wi-Fi Enabled");
    w.p();
    }
    }
```

OUTPUT:

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>JAVA "C:\
k\hybridinheritance2.java"
Appliance: WasherX
Capacity: 7 kg
Feature: Wi-Fi Enabled
Type: Smart Washing Machine
```

## POLYMORPHISM

### 8A constructor]

```
class constructor1 {
    String a;
    constructor1(String a) {
    this.a = a;
    System.out.println("Car brand: " + a);
    }
    public static void main(String[] args) {
    constructor1 obj = new constructor1("Honda");
    }
    }
```

### OUTPUT:

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>JAVA "C:
k\constructor1.java"
Car brand: Honda
```

### 8B constructor overloading]

```
class overloading {
    String x;
    int y;
    overloading(String x) {
    this.x = x;
    System.out.println("Name: " + x);
    }
    overloading(String x, int y) {
    this.x = x;
    this.y = y;
    System.out.println("Name: " + x);
    System.out.println("ID: " + y);
    }
    public static void main(String[] args) {
        overloading obj1 = new overloading("Dhoni");
        overloading obj2 = new overloading("Virat", 18);
    }
    }
```

**OUTPUT:**

```
C:\Users\Siddharth\Desktop\college\
k\overloading.java"
Name: Dhoni
Name: Virat
ID: 18
```

## 8C method overloading]

```java
class Employee {
    private String name;
    private int id;
    private double salary;
    void setDetails(String name, int id) {
        this.name = name;
        this.id = id;
    }
    void setDetails(String name, int id, double salary) {
        this.name = name;
        this.id = id;
        this.salary = salary;
    }
    void setDetails(String name) {
        this.name = name;
    }
    void displayDetails() {
        System.out.println("Name: " + name + ", ID: " + id + ", Salary: " + salary);
    }
}
public class methodoverloading1 {
    public static void main(String[] args) {
        Employee emp1 = new Employee();
        emp1.setDetails("siddharth", 101);
        emp1.displayDetails();
        Employee emp2 = new Employee();
        emp2.setDetails("srini", 102, 50000.0);
        emp2.displayDetails();
        Employee emp3 = new Employee();
        emp3.setDetails("raaghav");
        emp3.displayDetails();
    }
}
```

**OUTPUT:**

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>JAVA "C:\Use
k\methodoverloading1.java"
Name: siddharth, ID: 101, Salary: 0.0
Name: srini, ID: 102, Salary: 50000.0
Name: raaghav, ID: 0, Salary: 0.0
```

**8D** method overloading]

```
class Shape {
    void draw(double radius) {
        System.out.println("Drawing a circle with radius: " + radius);
    }
 void draw(double length, double width) {
        System.out.println("Drawing a rectangle with length: " + length + " and width: " + width);
    }
    void draw(double side1, double side2, double side3) {
        System.out.println("Drawing a triangle with sides: " + side1 + ", " + side2 + ", " + side3);
    }
}
public class  methodoverloading2{
    public static void main(String[] args) {
        Shape shape = new Shape();
        shape.draw(65.0);
        shape.draw(12.0, 12.0);
        shape.draw(3.0, 4.0, 5.0);     }
}
```

**OUTPUT:**

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>JAVA "C:\Users\Siddh
k\methodoverloading2.java"
Drawing a circle with radius: 65.0
Drawing a rectangle with length: 12.0 and width: 12.0
Drawing a triangle with sides: 3.0, 4.0, 5.0
```

**8E** method overridding]

```
class Vehicle {
    void speed() {
        System.out.println("Vehicle is moving");
    }
}

class Car extends Vehicle {
    void speed() {
        System.out.println("Car moves at 80 km/h");
    }
}

class Bike extends Vehicle {
    void speed() {
        System.out.println("Bike moves at 60 km/h");
    }
}

public class methodoverriding1 {
    public static void main(String[] args) {
        Vehicle v;
        v = new Car();
        v.speed();
        v = new Bike();
        v.speed();
    }
}
```

**OUTPUT:**

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>JAVA "C:
k\methodoverriding1.java"
Car moves at 80 km/h
Bike moves at 60 km/h
```

**8F** method overridding**]**

```java
class Bank {
    double getInterestRate() {
        return 0;
    }
}

class SBI extends Bank {
    double getInterestRate() {
        return 5.5;
    }
}

class ICICI extends Bank {
    double getInterestRate() {
        return 6.7;
    }
}

class HDFC extends Bank {
    double getInterestRate() {
        return 7.2;
    }
}

public class methodoverriding2 {
    public static void main(String[] args) {
        Bank b;

        b = new SBI();
        System.out.println("SBI Interest Rate: " + b.getInterestRate() + "%");

        b = new ICICI();
        System.out.println("ICICI Interest Rate: " + b.getInterestRate() + "%");

        b = new HDFC();
        System.out.println("HDFC Interest Rate: " + b.getInterestRate() + "%");
    }
}
```

**OUTPUT:**

```
k\methodoverriding2.java"
SBI Interest Rate: 5.5%
ICICI Interest Rate: 6.7%
HDFC Interest Rate: 7.2%
```

## ABSTRACTION(interface)

**1.**

```java
interface Engine {
    void start();
}

interface Wheels {
    void rotate();
}

class Bike implements Engine, Wheels {
    public void start() {
        System.out.println("Bike engine starting...");
    }

    public void rotate() {
        System.out.println("Bike wheels rotating...");
    }
}

public class interface1{
    public static void main(String[] args) {
        Bike myBike = new Bike();
        myBike.start();
        myBike.rotate();
    }
}
```

**OUTPUT:**

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,suspend=y,address=localhost:53490' '-XX:+ShowCodeDetailsInE
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8c087069\redhat.java\jdt_ws\lab work_df56f04e\bin' 'interface1'
Bike engine starting...
Bike wheels rotating...
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**2.**

```java
interface Shape{
    void getArea();
    }
    class Rectangle implements Shape{
        double lenght,breadth;

        Rectangle(double lenght, double breadth) {
            this.lenght = lenght;
            this.breadth = breadth;
        }
        public void getArea(){
           System.out.println("The area of rectangle "+(lenght*breadth));
        }
    }
    class Circle implements Shape{
        double radius;
        Circle(double radius){
            this.radius=radius;
        }
        public void getArea(){
            System.out.println("The are of circle is "+(3.14*(radius*radius)));
        }
    }
    public class interface2 {
        Run | Debug
        public static void main(String[] args) {
            Rectangle r1=new Rectangle(lenght:12, breadth:12);
            r1.getArea();
            Circle c1=new Circle(radius:4);
            c1.getArea();
        }

    }
```

**OUTPUT:**

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>JAVA "C
k\interface2.java"
The area of rectangle 144.0
The are of circle is 50.24
```

**3.**

interface3.java > interface3

```java
interface Bank {
    void interest();
    }

    class Savings implements Bank {
    double amount;
    Savings(double amount) {
    this.amount = amount;
    }
    public void interest() {
    System.out.println("Savings Interest: " + (amount * 0.05));
    }
    }

    class Current implements Bank {
    double amount;
    Current(double amount) {
    this.amount = amount;
    }
    public void interest() {
    System.out.println("Current Interest: " + (amount * 0.02));
    }
    }

    public class interface3 {
    Run | Debug
    public static void main(String[] args) {
    Savings s = new Savings(amount:10000);
    s.interest();
    Current c = new Current(amount:15000);
    c.interest();
    }
    }
```

PROBLEMS 57 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d692
Savings Interest: 500.0
Current Interest: 300.0
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**4.**

interface4.java > interface4

```java
interface Person {
    void details();
    }

    class Teacher implements Person {
    String name;
    int id;
    Teacher(String name, int id) {
    this.name = name;
    this.id = id;
    }
    public void details() {
    System.out.println("Teacher Name: " + name + ", ID: " + id);
    }
    }

    class Engineer implements Person {
    String name;
    String dept;
    Engineer(String name, String dept) {
    this.name = name;
    this.dept = dept;
    }
    public void details() {
    System.out.println("Engineer Name: " + name + ", Department: " + dept);
    }
    }

    public class interface4 {
    Run | Debug
    public static void main(String[] args) {
    Teacher t = new Teacher(name:"Ravi", id:101);
    t.details();
    Engineer e = new Engineer(name:"Maya", dept:"Software");
    e.details();
    }
    }
```

**OUTPUT:**

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>JAVA "C:\Users\Siddhar
k\interface4.java"
Teacher Name: Ravi, ID: 101
Engineer Name: Maya, Department: Software
```

**5.(abstract class)**

abstract1.java > ...

```java
abstract class M {
    abstract void p();
    void q() {
    System.out.println(x:"Animal is breathing...");
    }
    }

    class N extends M {
    void p() {
    System.out.println(x:"Dog is barking...");
    }
    }

    public class abstract1 {
    Run | Debug
    public static void main(String[] args) {
    M a = new N();
    a.p();
    a.q();
    }
    }
```

PROBLEMS 54 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.e
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106
Dog is barking...
Animal is breathing...
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**6.**

abstraction2.java

abstraction2.java > abstraction2

```java
abstract class Shape {
    String color;

    Shape(String color) {
        this.color = color;
    }

    abstract void draw();
}

class Circle extends Shape {
    Circle(String color) {
        super(color);
    }

    void draw() {
        System.out.println("Drawing a " + color + " circle");
    }
}

public class abstraction2{
    Run | Debug
```

PROBLEMS 54 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStora
Drawing a black circle
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

7.

abstractiion3.java > Lam > carSpeed(int, String)

```java
public class abstractiion3{
    Run | Debug
    public static void main(String[] args){
    Car c1=new Lam();
    c1.carName(name:"pagani");
    c1.carSpeed(speed:122,name:"ferrari");
    }
    }

    abstract class Car{
    abstract void carName(String name);

    abstract void carSpeed(int speed,String name);

    abstract void Mileage(int fuel,double mileage);

    }
    class Lam extends Car{
    void carName(String name){
    System.out.println("Your Car name is "+name);
    }
    void carSpeed(int speed,String name){
```

PROBLEMS 54 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\j
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d692
Your Car name is pagani
ferrari can travel at the speed of 122 km/h
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**8.**

```java
abstract class A {
    abstract void x();
    void y() {
    System.out.println(x:"Powering the device...");
    }
    }

    class B extends A {
    void x() {
    System.out.println(x:"Laptop is booting...");
    }
    }

    public class abstraction4 {
    Run | Debug
    public static void main(String[] args) {
    A obj = new B();
    obj.x();
    obj.y();
    }
    }
```

**Output:**

```
C:\Users\Siddharth\Desktop\college\OOPS\lab work>java "C:
k\abstraction4.java"
Laptop is booting...
Powering the device...
```

**ENCAPULATION:**

**1.**

encapsulation1.java > encapsulation1

```java
class Person {
    private int age;

    public int getAge() {
        return age;
    }

    public void setAge(int age) {
        if (age > 0) {
            this.age = age;
        } else {
            System.out.println(x:"Age must be positive!");
        }
    }
}

public class encapsulation1 {
    Run | Debug
    public static void main(String[] args) {
        Person person = new Person();
        person.setAge(age:43);
        System.out.println("Person's age: " + person.getAge());
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d69
Person's age: 43
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**2.**

encapsulation2.java > encapsulation2

```java
class Book {
    public void setTitle(String title) {
    }

    public String getAuthor() {
    return author;
    }

    public void setAuthor(String author) {
    this.author = author;
    }
    }

    public class encapsulation2 {
    Run | Debug
    public static void main(String[] args) {
    Book b = new Book();
    b.setTitle(title:"The Alchemist");
    b.setAuthor(author:"Paulo Coelho");
    System.out.println("Title: " + b.getTitle());
    System.out.println("Author: " + b.getAuthor());
    }
    }
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-age
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7
Title: The Alchemist
Author: Paulo Coelho
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**3.**

encapsulation3.java > ...

```java
class BankAccount {
    private double balance;

    public String getAccountNumber() {
    return accountNumber;
    }

    public void setAccountNumber(String accountNumber) {
    this.accountNumber = accountNumber;
    }

    public double getBalance() {
    return balance;
    }

    public void setBalance(double balance) {
    if (balance >= 0) {
    this.balance = balance;
    } else {
    System.out.println(x:"Balance cannot be negative!");
    }
    }
    }

    public class encapsulation3 {
    Run | Debug
    public static void main(String[] args) {
    BankAccount acc = new BankAccount();
    acc.setAccountNumber(accountNumber:"AC987654");
    acc.setBalance(balance:5000.75);
    System.out.println("Account Number: " + acc.getAccountNumber());
    System.out.println("Balance: RS:" + acc.getBalance());
    }
    }
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8
Employee Name: Nina
Employee ID: 1022
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**4.**

encapsulation4.java

encapsulation4.java > encapsulation4

```java
class Employee {
    private String empName;
    private int empID;

    public String getEmpName() {
    return empName;
    }

    public void setEmpName(String empName) {
    this.empName = empName;
    }

    public int getEmpID() {
    return empID;
    }

    public void setEmpID(int empID) {
    if (empID > 0) {
    this.empID = empID;
    } else {
    System.out.println(x:"Invalid employee ID!");
    }
    }
    }

    public class encapsulation4 {
    Run | Debug
    public static void main(String[] args) {
    Employee e = new Employee();
    e.setEmpName(empName:"Nina");
    e.setEmpID(empID:1022);
    System.out.println("Employee Name: " + e.getEmpName());
    System.out.println("Employee ID: " + e.getEmpID());
    }
    }
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8
Employee Name: Nina
Employee ID: 1022
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

# Built in packages:

1.

builtinpackage1.java > builtinpackage1 > main(String[])

```java
import java.util.Scanner;
import java.io.FileWriter;
import java.io.IOException;

public class builtinpackage1 {
    Run | Debug
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.println(x:"Enter your name:");
        String name = input.nextLine();
        System.out.println(x:"Enter your age:");
        int age = input.nextInt();

        try {
            FileWriter writer = new FileWriter(fileName:"user_info.txt");
            writer.write("Name: " + name + "\n");
            writer.write("Age: " + age);
            writer.close();
            System.out.println(x:"User info saved to file successfully!");
        } catch (IOException e) {
            System.out.println(x:"An error occurred while saving the file.");
        }

        input.close();
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:jdwp=transp
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8c087069\red
Enter your name:
siddharth
Enter your age:
17
User info saved to file successfully!
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**2.**

builtinpackage2.java > builtinpackage2

```java
import java.util.Scanner;
import java.net.URL;
import java.io.BufferedReader;
import java.io.InputStreamReader;

public class builtinpackage2 {
    Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println(x:"Enter a website URL (e.g., https://www.example.com):");
        String urlString = scanner.nextLine();
        System.out.println(x:"Enter a keyword to search:");
        String keyword = scanner.nextLine();

        try {
            URL url = new URL(urlString);
            BufferedReader reader = new BufferedReader(new InputStreamReader(url.openStream()));

            String line;
            int count = 0;
            while ((line = reader.readLine()) != null) {
                int index = 0;
                while ((index = line.indexOf(keyword, index)) != -1) {
                    count++;
                    index += keyword.length();
                }
            }
            reader.close();
            System.out.println("The keyword \"" + keyword + "\" appeared " + count + " times.");
        } catch (Exception e) {
            System.out.println(x:"Unable to fetch or search the URL content.");
        }

        scanner.close();
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:jdwp=transport=d
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8c087069\redhat.j
Enter a website URL (e.g., https://www.example.com):
https://www.youtube.com
Enter a keyword to search:
music
The keyword "music" appeared 73 times.
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**User defined packages:**

**1.**

```java
package mypackage;

public class packageone {
    public int add(int a, int b) {
        return a + b;
    }

    public int multiply(int a, int b) {
        return a * b;
    }
}

```

```java
import mypackage.packageone;

public class MainApp {
    Run | Debug
    public static void main(String[] args) {
        packageone calc = new packageone();
        int result1 = calc.add(a:10, b:5);
        int result2 = calc.multiply(a:6, b:7);

        System.out.println("Addition: " + result1);
        System.out.println("Multiplication: " + result2);
    }
}

```

**Output:**

```
C:\Users\Siddharth\Desktop\college\OOPS\vishnu>javac -d . packageone.java
```

```
C:\Users\Siddharth\Desktop\college\OOPS\vishnu>javac MainApp.java

C:\Users\Siddharth\Desktop\college\OOPS\vishnu>java MainApp
Addition: 15
Multiplication: 42
```

2.

C: > Users > Siddharth > Desktop > college > OOPS > vishnu > StringHelper.java > {} utils

```java
package utils;

public class StringHelper {
    public String reverse(String input) {
        return new StringBuilder(input).reverse().toString();
    }

    public String toUpper(String input) {
        return input.toUpperCase();
    }
}
```

```java
import utils.StringHelper;

public class App {
    Run | Debug
    public static void main(String[] args) {
        StringHelper helper = new StringHelper();

        String original = "hello world";
        String reversed = helper.reverse(original);
        String uppercased = helper.toUpper(original);

        System.out.println("Original: " + original);
        System.out.println("Reversed: " + reversed);
        System.out.println("Uppercased: " + uppercased);
    }
}
```

**Output:**

```
Microsoft Windows [Version 10.0.26100.3624]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Siddharth\Desktop\college\OOPS\vishnu>javac -d . StringHelper.java

C:\Users\Siddharth\Desktop\college\OOPS\vishnu>javac App.java

C:\Users\Siddharth\Desktop\college\OOPS\vishnu>java App
Original: hello world
Reversed: dlrow olleh
Uppercased: HELLO WORLD

C:\Users\Siddharth\Desktop\college\OOPS\vishnu>
```

**Exception handling:**

**1.**

exception1.java > ...

```java
public class exception1 {
    Run | Debug
    public static void main(String[] args) {
        try {
            int a = 10, b = 0;
            int result = a / b;
            System.out.println("Result: " + result);
        } catch (ArithmeticException e) {
            System.out.println(x:"Cannot divide by zero!");
        }
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff1
Cannot divide by zero!
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**2.**

exception2.java > exception2 > main(String[])

```java
public class exception2 {
    Run | Debug
    public static void main(String[] args) {
        try {
            int[] arr = {1, 2, 3};
            System.out.println(arr[5]);
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println(x:"Index is out of bounds!");
        }
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8
Cannot divide by zero!
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**3.**

exception3.java > ...

```java
public class exception3{
    Run | Debug
    public static void main(String[] args) {
        try {
            String str = null;
            System.out.println(str.length());
        } catch (NullPointerException e) {
            System.out.println(x:"Null reference accessed!");
        }
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\jav
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff
Null reference accessed!
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**4.**

exception4.java > ...

```java
public class exception4 {
    Run | Debug
    public static void main(String[] args) {
        try {
            String s = "abc";
            int num = Integer.parseInt(s);
            System.out.println("Number: " + num);
        } catch (NumberFormatException e) {
            System.out.println(x:"Invalid number format!");
        }
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:jd
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8c6
Invalid number format!
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**File Handling:**

**1.**

FileAppend.java > ...

```java
import java.io.FileWriter;
import java.io.IOException;

public class FileAppend {
    Run | Debug
    public static void main(String[] args) {
        try {
            FileWriter writer = new FileWriter(fileName:"data.txt", append:true);
            writer.write(str:"\nThis line is appended.");
            writer.close();
            System.out.println(x:"Data appended successfully.");
        } catch (IOException e) {
            System.out.println(x:"Error appending to file.");
        }
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe'
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c
Data appended successfully.
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**2.**

FileRead.java > ...

```java
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

public class FileRead {
    Run | Debug
    public static void main(String[] args) {
        try {
            File file = new File(pathname:"data.txt");
            Scanner sc = new Scanner(file);
            while (sc.hasNextLine()) {
                String line = sc.nextLine();
                System.out.println(line);
            }
            sc.close();
        } catch (FileNotFoundException e) {
            System.out.println(x:"File not found.");
        }
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8
Word replaced successfully.
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**3.**

FileAppend.java | FileReplace.java

FileReplace.java > ...

```java
import java.io.*;
import java.nio.file.*;

public class FileReplace {
    Run | Debug
    public static void main(String[] args) {
        String filePath = "data.txt";
        try {
            String content = new String(Files.readAllBytes(Paths.get(filePath)));
            content = content.replace(target:"first", replacement:"updated");
            Files.write(Paths.get(filePath), content.getBytes());
            System.out.println(x:"Word replaced successfully.");
        } catch (IOException e) {
            System.out.println(x:"Error processing file.");
        }
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe' '-agentlib:jdwp=
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c17adc7857ea8c0870
Word replaced successfully.
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work> ^C
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  c:; cd 'c:\Users\Siddharth\Desktop\college\OOPS\lab work'; &
uspend=y,address=localhost:54722' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roamin
_df56f04e\bin' 'FileReplace'
Word replaced successfully.
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

**4.**

FileAppend.java | FileWrite.java

FileWrite.java > ...

```java
import java.io.FileWriter;
import java.io.IOException;

public class FileWrite {
    Run | Debug
    public static void main(String[] args) {
        try {
            FileWriter writer = new FileWriter(fileName:"data.txt");
            writer.write(str:"This is the first line.");
            writer.close();
            System.out.println(x:"Data written successfully.");
        } catch (IOException e) {
            System.out.println(x:"Error writing file.");
        }
    }
}
```

PROBLEMS 50 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>  & 'C:\Program Files\Java\jdk-23\bin\java.exe'
ceptionMessages' '-cp' 'C:\Users\Siddharth\AppData\Roaming\Code\User\workspaceStorage\d6920ff106e8c1
Data written successfully.
PS C:\Users\Siddharth\Desktop\college\OOPS\lab work>
```

File Edit View

```
This is the first line.
```