

# Sri Krishna College of Engineering and Technology

Name :ASHWIN SRIRAM R  
Roll no:727721EUEC012  
Branch :SKCET  
Batch :2021-25

Email :727721euec012@skcet.ac.in  
Phone :9865674474  
Department :ECE  
Degree :BE-ECE

2021\_25\_IV\_JAVA Programming (ECE/EEE)\_IRC

SKG\_JAVA\_D15\_Inheritance&Encapsulation\_CE\_COD

Attempt : 1  
Total Mark : 50  
Marks Obtained : 50

## Section 1 : COD

1. Given a Book class and the main class, write a MyBook class that does the following:

Inherits from the book and has a parameterized constructor taking these 3 parameters:

string title, string author, int price. Implements the Book class' abstract display() method so it prints the title, author, and price.

**Answer**

```
import java.util.*;

abstract class Book {
    protected String title;
    protected String author;

    public Book(String title, String author) {
        this.title = title;
        this.author = author;
    }

    public abstract void display();
}
```

```

}

class MyBook extends Book {
    private int price;

    public MyBook(String title, String author, int price) {
        super(title, author);
        this.price = price;
    }

    public void display() {
        System.out.println("Title: " + title);
        System.out.println("Author: " + author);
        System.out.println("Price: " + price);
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        String title = scanner.nextLine();
        String author = scanner.nextLine();
        int price = scanner.nextInt();

        Book book = new MyBook(title, author, price);
        book.display();
    }
}

```

**Status :** Correct

**Marks :** 10/10

2. Write a java program to rotate a matrix n times in clockwise and anti-clockwise direction using inheritance.

Create a base class Clock.

From this extends a child class Clockwise and Anticlockwise.

Both these child classes should contain the method rotate().

**Answer**

```

// You are using Java
import java.util.*;
class Main
{
    public static void main(String[] args) {
        Scanner s=new Scanner(System.in);
        int row=s.nextInt();
        int col=s.nextInt();
        int [][]a=new int[row][col];
        for(int i=0;i<row;i++)
        {
            for(int j=0;j<col;j++)
            {
                a[i][j]=s.nextInt();
            }
        }
        int n=s.nextInt();
        Clock cl=new Clock();
        Clock cc=new Clockwise();
        Clock ac=new Anticlockwise();
        int [][]b=new int[row][col];
        for(int i=0;i<row;i++)
        {
            for(int j=0;j<col;j++)
            {
                b[i][j]=a[i][j];
            }
        }
        Anticlockwise acw=new Anticlockwise();
        for(int k=0;k<n;k++)
        cc.rotate(row,col,a,n);
        System.out.println("Clockwise ");
        for (int i = 0; i < row; i++)
        {
            for (int j = 0; j < col; j++)
                System.out.print( a[i][j] + " ");
            System.out.print("\n");
        }
        System.out.println("Anti clockwise ");
        int r=row;
        int c=col;
        int k=n;
    }
}

```

```

int f,K;
int l = 0;
int m = 0;
int Row = r-1;
int Col = c-1;
while(l < Row && m < Col)
{
    int rot = 2*(Row-l)+2*(Col-m);
    f = n%rot;
    for(int i=1;i<=f;i++)
    {
        ac.rotate(row,col,b,n);

        acw.rotation(l,m,Row,Col,b);
    }
    l++;
    m++;
    Row--;
    Col--;
}
for (int i = 0; i < row; i++)
{
    for (int j = 0; j < col; j++)
        System.out.print( b[i][j] + " ");
    System.out.print("\n");
}

```

```

}
}
class Clock
{
    public Clock()
    {

    }
    void rotate(int row,int col,int a[],int num)
    {

    }
}
}

```

```

class Clockwise extends Clock
{
    static int R;
    static int C;

    void rotate(int m,int n,int mat[],int num)
    {
        R = m;
        C = n;

        int row = 0, col = 0;
        int prev, curr;
        while (row < m && col < n)
        {
            if (row + 1 == m || col + 1 == n)
                break;
            prev = mat[row + 1][col];
            for (int i = col; i < n; i++)
            {
                curr = mat[row][i];
                mat[row][i] = prev;
                prev = curr;
            }
            row++;
            for (int i = row; i < m; i++)
            {
                curr = mat[i][n-1];
                mat[i][n-1] = prev;
                prev = curr;
            }
            n--;
            if (row < m)
            {
                for (int i = n-1; i >= col; i--)
                {
                    curr = mat[m-1][i];
                    mat[m-1][i] = prev;
                    prev = curr;
                }
            }
            m--;
            if (col < n)

```

```

    {
        for (int i = m-1; i >= row; i--)
        {
            curr = mat[i][col];
            mat[i][col] = prev;
            prev = curr;
        }
    }
    col++;
}

```

```

}
}
class Anticlockwise extends Clock
{
    void rotate(int m,int n,int mat[],int num)
    {}
    void rotation(int l, int m, int Row, int Col,int mat[])
    {
        int si,sj,i,j,t,f;
        si = l;
        sj = m;
        t = mat[l][m];

        for(i=l+1;i<=Row;i++)
        {
            f = mat[i][m];
            mat[i][m] = t;
            t = f;
        }
        l++;
        for(i=m+1;i<=Col;i++)
        {
            f = mat[Row][i];
            mat[Row][i] = t;
            t = f;
        }
        m++;
        if(l-1 < Row)
        {
            for(i=Row-1;i>=l-1;i--)

```

```

        {
            f = mat[i][Col];
            mat[i][Col] = t;
            t = f;
        }
    }
    Col--;
    if(m-1 < Col)
    {
        for(i=Col;i>=m;i--)
        {
            f = mat[l-1][i];
            mat[l-1][i] = t;
            t = f;
        }
    }
    Row--;
    mat[s1][s2] = t;
    return;
}

}

```

**Status :** Correct

**Marks :** 10/10

### 3. Employee Details:

Create a class called "Employee" in which the employee's name, ID, and salary are declared as private attributes and the company name is a static variable. The data types of the members are as follows:

String eName;

int eId;

int eSalary;

String companyName ; // static variable

Note: Use the public setter and getter methods to set and read the value of the attributes.

Initial companyName = "ABC Corp"

### ***Answer***

```
import java.util.Scanner;

class Employee {
    private String eName;
    private int eId;
    private int eSalary;
    private static String companyName = "ABC Corp";

    public void setEName(String eName) {
        this.eName = eName;
    }

    public void setEId(int eId) {
        this.eId = eId;
    }

    public void setESalary(int eSalary) {
        this.eSalary = eSalary;
    }

    public static void setCompanyName(String companyName) {
        Employee.companyName = companyName;
    }

    public String getEName() {
        return eName;
    }

    public int getEId() {
        return eId;
    }

    public int getESalary() {
        return eSalary;
    }

    public static String getCompanyName() {
        return companyName;
    }
}
```



```

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int n = scanner.nextInt();
        Employee[] employees = new Employee[n];
        for (int i = 0; i < n; i++) {
            Employee employee = new Employee();
            employee.setENAME(scanner.next());
            employee.setEId(scanner.nextInt());
            employee.setESalary(scanner.nextInt());
            employees[i] = employee;
        }
        String companyName = scanner.next();
        Employee.setCompanyName(companyName);
        for (Employee employee : employees) {
            System.out.println("Employee Name: " + employee.getENAME());
            System.out.println("Employee ID: " + employee.getEId());
            System.out.println("Employee Salary: " + employee.getESalary());
            System.out.println("Employee Company Name: ABC Corp" );
        }
        System.out.println("Updated Details");
        for (Employee employee : employees) {
            System.out.println("Employee Name: " + employee.getENAME());
            System.out.println("Employee ID: " + employee.getEId());
            System.out.println("Employee Salary: " + employee.getESalary());
            System.out.println("Employee Company Name: " +
Employee.getCompanyName());
        }
    }
}

```

**Status :** Correct

**Marks :** 10/10

#### 4. Account Details:

Create a class 'Account' in which all the fields are declared private.

Attributes are as follows:

```
int account_number;
```

```
int account_balance;
```

Get n user details and store them in the private variables. The account number to which the deposit must be made is then read. If an account number already exists, print the account balance; otherwise, display "Account Number Does Not Exist."

Note:

Use public setter and getter methods to set and read the value of the attributes.

The initial value of account\_balance is zero.

**Answer**

```
import java.util.*;

class Account {
    private int account_number;
    private int account_balance;

    public void setAccountNumber(int account_number) {
        this.account_number = account_number;
    }

    public int getAccountNumber() {
        return account_number;
    }

    public void setAccountBalance(int account_balance) {
        this.account_balance = account_balance;
    }

    public int getAccountBalance() {
        return account_balance;
    }
}

public class Main {
    public static void main(String args[]) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
```

```

Account[] acc = new Account[n];

for (int i = 0; i < n; i++) {
    int acc_num = sc.nextInt();
    int acc_bal = sc.nextInt();
    acc[i] = new Account();
    acc[i].setAccountNumber(acc_num);
    acc[i].setAccountBalance(acc_bal);
}

int acc_search = sc.nextInt();
boolean found = false;
for (int i = 0; i < n; i++) {
    if (acc[i].getAccountNumber() == acc_search) {
        System.out.println(acc[i].getAccountBalance());
        found = true;
        break;
    }
}

if (!found) {
    System.out.println("Account Number does not exist");
}
}
}

```

**Status :** Correct

**Marks :** 10/10

5. A company maintains a database that has the details of all the employees. There are two levels of employees where level 1 is the top management having salary more than 100 dollars and level 2 is the staffs who are getting a salary less than 100 dollars. Create a class named Employee with empId and salary as attributes. Create another class empLevel that extends employee and categorizes the employee into various levels.

**Answer**

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

```
class Employee {
```

```

private int empld;
private float salary;

public Employee(int empld, float salary) {
    this.empld = empld;
    this.salary = salary;
}

public int getEmpld() {
    return empld;
}

public float getSalary() {
    return salary;
}
}

class EmpLevel extends Employee {
    private int level;

    public EmpLevel(int empld, float salary) {
        super(empld, salary);
        if (salary > 100.0) {
            this.level = 1;
        } else {
            this.level = 2;
        }
    }

    public int getLevel() {
        return level;
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        int empld = scanner.nextInt();
        float salary = scanner.nextFloat();

        EmpLevel emp = new EmpLevel(empld, salary);
    }
}

```

```
        System.out.println(emp.getEmpId());
        System.out.println(emp.getSalary());
        System.out.println(emp.getLevel());

        scanner.close();
    }
}
```

**Status :** Correct

**Marks :** 10/10