* **F to Celsious**

class Calculation {

public double convert(double F) {

return (F - 32) / 1.8;

}

}

public class FtoC {

public static void main(String[] args) { // এখানে "Strring" -> "String" হবে

Calculation cal = new Calculation();

System.out.println("Temperature in celsius: " + cal.convert(45));

}

}

**Prime**   
class PrimeNumbers {

// Method to check prime

boolean isPrime(int n) {

if (n <= 1) return false; // 1 or less is not prime

for (int i = 2; i <= n / 2; i++) {

if (n % i == 0) return false; // divisible means not prime

}

return true;

}

public static void main(String[] args) {

PrimeNumbers obj = new PrimeNumbers(); // object create

int start = 10; // starting number

int end = 50; // ending number

System.out.println("Prime numbers between " + start + " and " + end + " are:");

for (int i = start; i <= end; i++) {

if (obj.isPrime(i)) {

System.out.print(i + " ");

}

}

}

}

Or,

import java.util.Scanner;

public class PrimeInterval {

// Method to check if a number is prime

public static boolean isPrime(int n) {

if (n <= 1)

return false;

if (n == 2)

return true;

if (n % 2 == 0)

return false;

for (int i = 3; i \* i <= n; i += 2) {

if (n % i == 0)

return false;

}

return true;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the starting interval: ");

int start = sc.nextInt();

System.out.print("Enter the ending interval: ");

int end = sc.nextInt();

System.out.println("Prime numbers between " + start + " and " + end + " are:");

for (int i = start; i <= end; i++) {

if (isPrime(i)) {

System.out.print(i + " ");

}

}

System.out.println();

sc.close();

}

}

\*\*\***CAr**

class Car {

private String brand;

private String model;

private int year;

private double price;

// Getter and Setter methods

public String getBrand() { return brand; }

public void setBrand(String brand) { this.brand = brand; }

public String getModel() { return model; }

public void setModel(String model) { this.model = model; }

public int getYear() { return year; }

public void setYear(int year) { this.year = year; }

public double getPrice() { return price; }

public void setPrice(double price) { this.price = price; }

}

public class CarTest {

public static void main(String[] args) {

Car myCar = new Car();

// Set values

myCar.setBrand("Toyota");

myCar.setModel("Corolla");

myCar.setYear(2022);

myCar.setPrice(20000.50);

// Display values

System.out.println("Car Brand: " + myCar.getBrand());

System.out.println("Car Model: " + myCar.getModel());

System.out.println("Car Year: " + myCar.getYear());

System.out.println("Car Price: $" + myCar.getPrice());

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**3 abstract vehicle \*\*\***

**abstract class Vehicle {**

**public abstract void startEngine();**

**public void stopEngine() {**

**System.out.println("Engine stopped.");**

**}**

**}**

**class Car extends Vehicle {**

**public void startEngine() {**

**System.out.println("Car engine started.");**

**}**

**}**

**class Motorcycle extends Vehicle {**

**public void startEngine() {**

**System.out.println("Motorcycle engine started.");**

**}**

**}**

**public class VehicleTest {**

**public static void main(String[] args) {**

**Vehicle myCar = new Car();**

**Vehicle myBike = new Motorcycle();**

**myCar.startEngine();**

**myCar.stopEngine();**

**myBike.startEngine();**

**myBike.stopEngine();**

**}**

**}**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**\*\*\* jog biog gun vag**

**import java.util.Scanner;**

**class Calculator {**

**// Methods for each operation**

**double add(double a, double b) {**

**return a + b;**

**}**

**double subtract(double a, double b) {**

**return a - b;**

**}**

**double multiply(double a, double b) {**

**return a \* b;**

**}**

**double divide(double a, double b) {**

**if (b == 0) {**

**System.out.println("Division by zero not allowed!");**

**return 0;**

**}**

**return a / b;**

**}**

**}**

**public class Main {**

**public static void main(String[] args) {**

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

**Calculator calc = new Calculator();**

**// Take inputs**

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

**double num1 = sc.nextDouble();**

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

**double num2 = sc.nextDouble();**

**// Menu**

**System.out.println("Choose Operation:");**

**System.out.println("1 - Addition");**

**System.out.println("2 - Subtraction");**

**System.out.println("3 - Multiplication");**

**System.out.println("4 - Division");**

**int choice = sc.nextInt();**

**// Perform operation**

**double result = 0;**

**switch (choice) {**

**case 1: result = calc.add(num1, num2); break;**

**case 2: result = calc.subtract(num1, num2); break;**

**case 3: result = calc.multiply(num1, num2); break;**

**case 4: result = calc.divide(num1, num2); break;**

**default: System.out.println("Invalid choice!");**

**}**

**// Show result**

**if (choice >= 1 && choice <= 4) {**

**System.out.println("Result: " + result);**

**}**

**sc.close();**

**}**

**}**

**Bank account**

class BankAccount{

    String name;

    int accNo;

    String accountType;

    double balance;

    public BankAccount(String name, int accNo, String accountType, double balance) {

        this.name = name;

        this.accNo = accNo;

        this.accountType = accountType;

        this.balance = balance;

    }

    public void deposit(double amount){

        if(amount > 0){

            balance+=amount;

            System.out.println("Succesfully deposite " + amount);

        }else{

            System.out.println("Invalid");

        }

    }

    public void withdraw(double amount){

        if(amount>0)

        if(balance >= amount){

            balance -=amount;

            System.out.println("Successfully withdrew " + amount);

        }

        else{

            System.out.println("Invalid");

        }

    }

    public void displayAccountInfo() {

        System.out.println("Account Holder Name: " + name);

        //System.out.println("Account Number: " + accNo);

        //System.out.println("Account Type: " + accountType);

        System.out.println("Account Balance: " + balance);

    }

}

public class Bacc

{

    public static void main( String[] args){

        BankAccount acc=new BankAccount("John Doe", 12345, "Savings", 1000.0);

        acc.displayAccountInfo();

        acc.deposit(500);

        acc.withdraw(200);

        acc.displayAccountInfo();

    }

}

**Using abstract**

**abstract class BankAccount {**

**String depositorName;**

**String accountNumber;**

**String accountType;**

**double balance;**

**public BankAccount(String depositorName, String accountNumber, String accountType, double balance) {**

**this.depositorName = depositorName;**

**this.accountNumber = accountNumber;**

**this.accountType = accountType;**

**this.balance = balance;**

**}**

**abstract void deposit(double amount);**

**abstract void withdraw(double amount);**

**abstract void display();**

**}**

**class MyBankAccount extends BankAccount {**

**public MyBankAccount(String depositorName, String accountNumber, String accountType, double balance) {**

**super(depositorName, accountNumber, accountType, balance);**

**}**

**@Override**

**void deposit(double amount) {**

**balance += amount;**

**System.out.println(amount + " deposited successfully.");**

**}**

**@Override**

**void withdraw(double amount) {**

**if (amount <= balance) {**

**balance -= amount;**

**System.out.println(amount + " withdrawn successfully.");**

**} else {**

**System.out.println("Insufficient balance.");**

**}**

**}**

**@Override**

**void display() {**

**System.out.println("Depositor Name: " + depositorName);**

**System.out.println("Account Number: " + accountNumber);**

**System.out.println("Account Type: " + accountType);**

**System.out.println("Balance: " + balance);**

**}**

**}**

**class BankAccountTest {**

**public static void main(String[] args) {**

**MyBankAccount acc1 = new MyBankAccount("Rahim", "12345", "Savings", 5000);**

**acc1.display();**

**acc1.deposit(2000);**

**acc1.withdraw(3000);**

**acc1.display();**

**}**

**}**

**\*\*\*evaporetot**

**class Door {**

**boolean close = true;**

**void close() {**

**close = true;**

**System.out.println("Door closed.");**

**}**

**void open() {**

**close = false;**

**System.out.println("Door opened.");**

**}**

**}**

**class ElevatorController {**

**int floorId;**

**int position;**

**boolean direction;**

**Door door;**

**public ElevatorController() {**

**door = new Door();**

**}**

**}**

**class Elevator {**

**boolean direction;**

**int currentFloor;**

**ElevatorController controller;**

**public Elevator() {**

**controller = new ElevatorController();**

**}**

**void move() {**

**System.out.println("Elevator moving.");**

**}**

**void stop() {**

**System.out.println("Elevator stopped.");**

**}**

**void status() {**

**System.out.println("Current Floor: " + currentFloor + ", Direction: " + direction);**

**}**

**}**

**class Button {**

**boolean illuminate = false;**

**void illuminate() {**

**illuminate = true;**

**System.out.println("Button illuminated.");**

**}**

**void cancelIlluminate() {**

**illuminate = false;**

**System.out.println("Button light off.");**

**}**

**void status() {**

**System.out.println("Button light status: " + (illuminate ? "On" : "Off"));**

**}**

**}**

**class ElevatorButton extends Button {**

**int floorNum;**

**}**

**class FloorButton extends Button {**

**int floorNum;**

**boolean direction;**

**}**

**class ElevatorTest {**

**public static void main(String[] args) {**

**Elevator elevator = new Elevator();**

**elevator.move();**

**elevator.status();**

**elevator.stop();**

**elevator.controller.door.open();**

**elevator.controller.door.close();**

**ElevatorButton eb = new ElevatorButton();**

**eb.floorNum = 5;**

**eb.illuminate();**

**eb.cancelIlluminate();**

**}**

**}**