TASK\_2

SOLUTION\_1

public class Solution\_1 {  
 // Properties of the class  
 private String name;  
 private int age;  
  
 // Default constructor with default age value  
 public Solution\_1() {  
 this.name = "Sedhu";  
 this.age = 18; // Default age  
 }  
  
 // Constructor to initialize name and age  
 public Solution\_1(String name, int age) {  
 this.name = name;  
 this.age = age;  
 }  
  
 // Method to display name and age of the person  
 public void displayInfo() {  
 System.*out*.println("Name: " + name);  
 System.*out*.println("Age: " + age);  
 }  
  
 // Main method to test the class  
 public static void main(String[] args) {  
 // Creating a Person object with default values  
 Solution\_1 person1 = new Solution\_1();  
 person1.displayInfo();  
  
 // Creating a Person object with custom values  
 Solution\_1 person2 = new Solution\_1("Prasanna", 15);  
 person2.displayInfo();  
 }  
}

OUTPUT

Name: Sedhu

Age: 18

Name: Prasanna

Age: 15

SOLUTION\_2

public class Solution\_2 {  
 private int pid; // Product ID  
 private double price; // Price of the product  
 private int quantity; // Quantity of the product  
  
 // Parameterized constructor to initialize product details  
 public Solution\_2(int pid, double price, int quantity) {  
 this.pid = pid;  
 this.price = price;  
 this.quantity = quantity;  
 }  
  
 // Getters for Product fields  
 public int getPid() {  
 return pid;  
 }  
  
 public double getPrice() {  
 return price;  
 }  
  
 public int getQuantity() {  
 return quantity;  
 }  
}

import java.util.Scanner;  
  
public class Solution\_2\_1 {  
  
 // Method to calculate total amount spent on all products  
 public static double calculateTotalAmount(Solution\_2[] products) {  
 double totalAmount = 0;  
  
 // Loop through each product to calculate amount  
 for (Solution\_2 product : products) {  
 totalAmount += product.getPrice() \* product.getQuantity();  
 }  
  
 return totalAmount;  
 }  
  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 Solution\_2[] products = new Solution\_2[5]; // Array to store 5 products  
  
 // Accept product information from the user  
 for (int i = 0; i < 5; i++) {  
 System.*out*.println("Enter details for Product " + (i + 1));  
  
 System.*out*.print("Enter Product ID: ");  
 int pid = scanner.nextInt();  
  
 System.*out*.print("Enter Price: ");  
 double price = scanner.nextDouble();  
  
 System.*out*.print("Enter Quantity: ");  
 int quantity = scanner.nextInt();  
  
 // Create Product object and store it in the array  
 products[i] = new Solution\_2(pid, price, quantity);  
 }  
  
 // Find the Product with the highest price  
 Solution\_2 highestPriceProduct = products[0];  
 for (int i = 1; i < 5; i++) {  
 if (products[i].getPrice() > highestPriceProduct.getPrice()) {  
 highestPriceProduct = products[i];  
 }  
 }  
  
 // Display the PID of the product with the highest price  
 System.*out*.println("Product with the highest price:");  
 System.*out*.println("Product ID: " + highestPriceProduct.getPid());  
 System.*out*.println("Price: " + highestPriceProduct.getPrice());  
  
 // Calculate and display the total amount spent on all products  
 double totalAmount = *calculateTotalAmount*(products);  
 System.*out*.println("Total amount spent on all products: " + totalAmount);  
  
 // Close the scanner to avoid resource leak  
 scanner.close();  
 }  
}

OUTPUT

Enter details for Product 1

Enter Product ID: 01

Enter Price: 100

Enter Quantity: 10

Enter details for Product 2

Enter Product ID: 02

Enter Price: 100

Enter Quantity: 10

Enter details for Product 3

Enter Product ID: 03

Enter Price: 100

Enter Quantity: 10

Enter details for Product 4

Enter Product ID: 04

Enter Price: 100

Enter Quantity: 10

Enter details for Product 5

Enter Product ID: 05

Enter Price: 100

Enter Quantity: 10

Product with the highest price:

Product ID: 1

Price: 100.0

Total amount spent on all products: 5000.0

SOLUTION\_3

public class Solution\_3 {  
 private double balance; // Data member to store the balance  
  
 // No-argument constructor  
 public Solution\_3() {  
 balance = 0.0; // Initialize balance to 0  
 }  
  
 // Constructor with two arguments: initial balance  
 public Solution\_3(double initialBalance) {  
 balance = initialBalance; // Initialize balance with provided value  
 }  
  
 // Method to deposit amount to the account  
 public void deposit(double amount) {  
 if (amount > 0) {  
 balance += amount; // Add the deposit amount to balance  
 System.*out*.println("Deposited: " + amount);  
 } else {  
 System.*out*.println("Deposit amount must be positive.");  
 }  
 }  
  
 // Method to withdraw amount from the account  
 public void withdraw(double amount) {  
 if (amount > 0 && amount <= balance) {  
 balance -= amount; // Subtract the withdrawal amount from balance  
 System.*out*.println("Withdrawn: " + amount);  
 } else if (amount > balance) {  
 System.*out*.println("Insufficient balance.");  
 } else {  
 System.*out*.println("Withdrawal amount must be positive.");  
 }  
 }  
  
 // Method to display the current balance  
 public void displayBalance() {  
 System.*out*.println("Current Balance: " + balance);  
 }  
}

import java.util.Scanner;  
  
public class Solution\_3\_1 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
  
 // Create an account using the no-argument constructor  
 Solution\_3 account = new Solution\_3();  
  
 // Alternatively, you could create an account with a specific balance like this:  
 // Account account = new Account(1000.0);  
  
 // Test deposit method  
 System.*out*.print("Enter amount to deposit: ");  
 double depositAmount = scanner.nextDouble();  
 account.deposit(depositAmount);  
  
 // Test withdraw method  
 System.*out*.print("Enter amount to withdraw: ");  
 double withdrawAmount = scanner.nextDouble();  
 account.withdraw(withdrawAmount);  
  
 // Display the balance  
 account.displayBalance();  
  
 // Close the scanner to avoid resource leak  
 scanner.close();  
 }  
}

OUTPUT

Enter amount to deposit: 50000

Deposited: 50000.0

Enter amount to withdraw: 1000

Withdrawn: 1000.0

Current Balance: 49000.0

SOLUTION\_4

// Base class  
public class Solution\_4 {  
 // Attributes of Person class  
 protected String name;  
 protected int age;  
  
 // Constructor to initialize Person's attributes  
 public Solution\_4(String name, int age) {  
 this.name = name;  
 this.age = age;  
 }  
  
 // Method to display person's details  
 public void displayPersonDetails() {  
 System.*out*.println("Name: " + name);  
 System.*out*.println("Age: " + age);  
 }  
}

// Subclass that inherits from Person  
public class Solution\_4\_1 extends Solution\_4 {  
 // Additional attributes for Employee class  
 private String employeeID;  
 private double salary;  
  
 // Constructor that uses 'super' to initialize Person's attributes and initializes Employee's attributes  
 public Solution\_4\_1(String name, int age, String employeeID, double salary) {  
 // Calling the constructor of the Person class to initialize name and age  
 super(name, age);  
 this.employeeID = employeeID;  
 this.salary = salary;  
 }  
  
 // Method to display employee details  
 public void displayEmployeeDetails() {  
 // Calling the displayPersonDetails() from the Person class to display person's details  
 displayPersonDetails();  
 System.*out*.println("Employee ID: " + employeeID);  
 System.*out*.println("Salary: " + salary);  
 }  
}

public class Solution\_4\_2 {  
 public static void main(String[] args) {  
 // Creating an instance of Employee  
 Solution\_4\_1 employee = new Solution\_4\_1("Sedhu prasanna", 19, "2K05", 50000.0);  
  
 // Displaying the details of the employee  
 employee.displayEmployeeDetails();  
 }  
}

OUTPUT

Name: Sedhu prasanna

Age: 19

Employee ID: 2K05

Salary: 50000.0