# Java Programs Collection

## Question-1.java

import java.util.\*;  
  
class Bank {  
 String customer\_name;  
 int account\_no;  
 double balance;  
  
 Bank() {  
 System.out.println("Default Account Created : ");  
 printInfo();  
 }  
  
 Bank(String name, int acc, double balance) {  
 System.out.println("Perimeterized Account Created : ");  
 this.customer\_name = name;  
 this.account\_no = acc;  
 this.balance = balance;  
 }  
  
 public void printInfo() {  
 System.out.println("Customer name = " + customer\_name);  
 System.out.println("Account name = " + account\_no);  
 System.out.println("Balance = " + balance);  
 }  
  
 @Override  
 protected void finalize() {  
 System.out.println("The object is garbage collected ");  
 }  
}  
  
public class Q1 {  
 public static void main(String[] args) {  
  
 Bank b1 = new Bank();  
 Bank b2 = new Bank("Syed Arham", 9397, 98123);  
 b2.printInfo();  
  
 b1 = null;  
 b2 = null;  
  
 System.gc();  
  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
  
}

## Question-2.java

import java.util.\*;  
  
class Students {  
 String name;  
 int ID;  
 double gpa;  
  
 Students() {  
 System.out.println("Default Student profile Created");  
 name = "Default Student";  
 System.out.println("Name = " + name);  
 System.out.println("ID = " + ID);  
 System.out.println("GPA = " + gpa);  
 }  
  
 Students(String name, int id, double gpa) {  
 this.name = name;  
 this.ID = id;  
 this.gpa = gpa;  
 System.out.println("New Student Profile Created");  
 System.out.println("Name = " + name);  
 System.out.println("ID = " + id);  
 System.out.println("GPA = " + gpa);  
  
 }  
  
 @Override  
 protected void finalize() {  
 if (this.ID == 0 && this.name.equals("Default Student")) {  
 System.out.println("Default Student Profile Deleted : " + ID + " " + name);  
 } else {  
 System.out.println("Student Record Deleted: " + ID + " " + name);  
 }  
 }  
  
}  
  
public class Q2 {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
 Students s1 = new Students();  
 s1 = new Students("Arham", 2551, 3.49);  
 s1 = null;  
  
 System.gc();  
  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
  
 }  
}

## Question-3.java

import java.util.\*;  
  
class Product {  
  
 String name;  
 double price;  
 int quantity;  
 boolean check = false;  
  
 Product() {  
 System.out.println("Default Product Added : ");  
 name = "Generic";  
 System.out.println("Prodcut : " + name);  
 System.out.println("price : " + price);  
 System.out.println("quantity : " + quantity);  
 }  
  
 Product(String name, int quantity, double price) {  
 this.name = name;  
 this.quantity = quantity;  
 this.price = price;  
 System.out.println("New product Added :");  
 System.out.println("Prodcut : " + name);  
 System.out.println("price : " + price);  
 System.out.println("quantity : " + quantity);  
 }  
  
 @Override  
 protected void finalize() {  
 if (this.quantity == 0 && this.name.equals("Generic")) {  
 System.out.println("Product Removed from cart : " + name);  
 } else  
 System.out.println("Product Removed from cart : " + name);  
 }  
}  
  
public class Q3 {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
 Product p1 = new Product();  
 System.out.print("Enter Product : ");  
 String name = sc.nextLine();  
 System.out.print("Enter quantity : ");  
 int quantity = sc.nextInt();  
 System.out.print("Enter price : ");  
 double price = sc.nextDouble();  
  
 p1 = new Product(name, quantity, price);  
  
 p1 = null;  
  
 System.gc();  
  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
  
 }  
}

## Question-4.java

import java.util.Scanner;  
  
class Employee {  
 String name;  
 double salary;  
 int ID;  
 double tax;  
 Double bonus;  
 double netSalary;  
  
 Employee() {  
 System.out.println("Default Employee Added: ");  
 name = "Unknown";  
 ID = 0;  
 salary = 0.0;  
 tax = 0.0;  
 bonus = new Double(0.0);  
 netSalary = 0.0;  
  
 displayDetails();  
 }  
  
 Employee(String name, int id, double salary, double tax, double bonusValue) {  
 this.name = name;  
 this.ID = id;  
 this.salary = salary;  
 this.tax = tax;  
 this.bonus = new Double(bonusValue);  
 calculateNetSalary();  
  
 System.out.println("Employee Added: ");  
 displayDetails();  
 }  
  
 void calculateNetSalary() {  
 double taxDeduction = salary \* (tax / 100);  
 this.netSalary = salary - taxDeduction + bonus;  
 }  
  
 void displayDetails() {  
 System.out.println("Name: " + name);  
 System.out.println("ID: " + ID);  
 System.out.println("Basic Salary: " + salary);  
 System.out.println("Tax Deduction (%): " + tax);  
 System.out.println("Bonus: " + bonus);  
 System.out.println("Net Salary: " + netSalary);  
 }  
  
 @Override  
 protected void finalize() throws Throwable {  
 System.out.println("Employee Deleted: " + ID);  
 System.out.println("Memory for Bonus is released.");  
 bonus = null;  
 }  
}  
  
public class Q4 {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
  
 Employee e1 = new Employee();  
  
 System.out.print("Employee Name: ");  
 String name = sc.nextLine();  
 System.out.print("ID: ");  
 int id = sc.nextInt();  
 System.out.print("Salary: ");  
 double salary = sc.nextDouble();  
 System.out.print("Tax Deduction (%): ");  
 double tax = sc.nextDouble();  
 System.out.print("Bonus: ");  
 double bonusValue = sc.nextDouble();  
  
 e1 = new Employee(name, id, salary, tax, bonusValue);  
  
 System.out.println("Delete Employee? 1)Yes 2)No");  
 int choice = sc.nextInt();  
  
 if (choice == 1) {  
 e1 = null; //  
 System.gc();  
 }  
  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
  
 sc.close();  
 }  
}

## Question-5.java

import java.util.\*;  
  
class Devices {  
 int ID;  
 String Type;  
 int Consumption;  
 boolean status;  
  
 Devices() {  
 ID = 0;  
 Type = "Unknown";  
 Consumption = 0;  
 status = false;  
 System.out.println("Default Devices Created");  
 printInfo();  
 }  
  
 Devices(int ID, String Type, int Consumption, boolean status) {  
 this.ID = ID;  
 this.Type = Type;  
 this.Consumption = Consumption;  
 this.status = status;  
 }  
  
 public void printInfo() {  
 System.out.println("ID: " + ID);  
 System.out.println("Type: " + Type);  
 System.out.println("Consumption: " + Consumption);  
 if (status == true) {  
 System.out.println("Status: On");  
 } else {  
 System.out.println("Status: Off");  
 }  
 }  
  
 public static int calPower(Devices dev[]) {  
 int total = 0;  
 for (int i = 0; i < dev.length; i++) {  
 total += dev[i].Consumption;  
 }  
  
 return total;  
 }  
  
}  
  
public class Q5 {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
  
 Devices d1 = new Devices();  
 int n;  
 System.out.print("Enter the number of devices to be added: ");  
 n = sc.nextInt();  
  
 Devices[] dev = new Devices[n];  
  
 for (int i = 0; i < n; i++) {  
 System.out.print("Enter Device ID : ");  
 int id = sc.nextInt();  
 sc.nextLine();  
 System.out.print("Enter Device Type : ");  
 String Type = sc.nextLine();  
 System.out.print("Power Comsumptions(in Watts) : ");  
 int Consumption = sc.nextInt();  
 System.out.print("Status of Device? 1)On 2)Off : ");  
 int choice = sc.nextInt();  
 boolean status;  
 if (choice == 1) {  
 status = true;  
 } else {  
 status = false;  
 }  
  
 dev[i] = new Devices(id, Type, Consumption, status);  
 }  
  
 for (int i = 0; i < n; i++) {  
 dev[i].printInfo();  
 }  
  
 int total = Devices.calPower(dev);  
 System.out.println("Total Power Consumption : " + total + " Watts");  
  
 System.out.print("Remove Devices? 1)Yes 2)No : ");  
 int choice = sc.nextInt();  
 if (choice == 1) {  
 System.out.print("Enter Device ID to be removed : ");  
 int id = sc.nextInt();  
 int count = 0;  
 for (int i = 0; i < n; i++) {  
 if (dev[i].ID == id) {  
 System.out.println("Device Record Deleted : " + dev[i].ID);  
 dev[i] = null;  
 }  
  
 if (count == n) {  
 count++;  
 System.out.println("Device not found");  
 }  
 }  
 }  
  
 }  
  
}

## Question-6.java

import java.util.\*;  
  
class Employee {  
 int id;  
 String name;  
  
 Employee() {  
 id = 0;  
 name = "Default";  
 }  
  
 Employee(int id, String name) {  
 this.id = id;  
 this.name = name;  
 }  
}  
  
class Manager extends Employee {  
 int teamSize;  
  
 Manager(int id, String name, int teamSize) {  
 super(id, name);  
 this.teamSize = teamSize;  
 }  
}  
  
public class Q6 {  
  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.in);  
  
 Employee e1 = new Employee(101, "Arham");  
 Manager m1 = new Manager(102, "Zaryab", 5);  
  
 System.out.println("Employee Details : ");  
 System.out.println("ID : " + e1.id);  
 System.out.println("Name : " + e1.name);  
  
 System.out.println("Manager Details : ");  
 System.out.println("ID : " + m1.id);  
 System.out.println("Name : " + m1.name);  
 System.out.println("Team Size : " + m1.teamSize);  
 }  
  
}

## Question-7.java

import java.util.\*;  
  
class Project {  
 String projectName;  
 String deadLine;  
 double budget;  
  
 Project() {  
 projectName = "Untitled";  
 deadLine = "Not Assigned";  
 budget = 0.0;  
 System.out.print("Default Project Created : \n");  
 System.out.println("Project Name : " + projectName);  
 System.out.println("DeadLine : " + deadLine);  
 System.out.println("Budget : " + budget);  
 }  
  
 Project(String projectName, String deadLine) {  
 this.projectName = projectName;  
 this.deadLine = deadLine;  
 System.out.println("Project Created : ");  
 System.out.println("Project Name : " + projectName);  
 System.out.println("DeadLine : " + deadLine);  
 }  
  
 Project(String projectName, String deadLine, double budget) {  
 this.projectName = projectName;  
 this.deadLine = deadLine;  
 this.budget = budget;  
 System.out.println("Project Created : ");  
 System.out.println("Project Name : " + projectName);  
 System.out.println("DeadLine : " + deadLine);  
 System.out.println("Budget : " + budget);  
 }  
  
}  
  
public class Q7 {  
 public static void main(String[] args) {  
  
 Project p1 = new Project();  
 Project p2 = new Project("AI Chatbot", "30th June 2025");  
 Project p3 = new Project("Education App", "31th Dec 2025");  
  
 }  
  
}

## Question-8.java

import java.util.\*;  
  
class Employee {  
 String name;  
 int id;  
 String Department;  
  
 Employee() {  
 name = "Default";  
 id = 0;  
 Department = "General";  
 System.out.println("Default Employee Added: ");  
 System.out.println("Name : " + name);  
 System.out.println("ID : " + id);  
 System.out.println("Department : " + Department);  
 }  
  
 Employee(String name, int id) {  
 this.name = name;  
 this.id = id;  
 Department = "General";  
 System.out.println("New Employee Added: ");  
 System.out.println("Name : " + name);  
 System.out.println("ID : " + id);  
 System.out.println("Department : " + Department);  
 }  
  
 Employee(String name, int id, String Department) {  
 this.name = name;  
 this.id = id;  
 this.Department = Department;  
 System.out.println("New Employee Added: ");  
 System.out.println("Name : " + name);  
 System.out.println("ID : " + id);  
 System.out.println("Department : " + Department);  
 }  
  
 @Override  
 protected void finalize() {  
 if (id == 0 && name.equals("Unknown") && Department.equals("General")) {  
 System.out.println("Default Employee Deleted:");  
 } else {  
 System.out.println("Employee Deleted: " + name + " (ID: " + id + ")");  
 }  
 }  
}  
  
class Manager extends Employee {  
 int teamSize;  
  
 Manager(String name, int id, int teamSize) {  
 super(name, id);  
 this.teamSize = teamSize;  
 System.out.println("New Manager Added: ");  
 System.out.println("Name : " + name);  
 System.out.println("ID : " + id);  
 System.out.println("Team Size : " + teamSize);  
  
 }  
}  
  
public class Q8 {  
  
 public static void main(String[] args) {  
  
 Employee e1 = new Employee();  
 Employee[] e2 = new Employee[5];  
 e2[0] = new Employee("Saad", 101);  
 e2[1] = new Employee("Ali", 102, "IT");  
 e2[2] = new Employee("Zaryab", 103, "HR");  
 e2[3] = new Employee("Abdul Rafay", 104, "AI");  
 e2[4] = new Employee("Syed Arham", 105);  
 Manager m1 = new Manager("Hamza", 1, 5);  
 e1 = null;  
 e2 = null;  
  
 System.gc();  
  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
  
}

## Question-9.java

import java.util.\*;  
  
class Project {  
 String projectName;  
 String deadLine;  
 double budget;  
  
 Project() {  
 projectName = "Untitled";  
 deadLine = "Not Assigned";  
 budget = 0.0;  
 System.out.println("Project Created : " + projectName + " , DeadLine : " + deadLine + " , Budget : " + budget);  
 }  
  
 Project(String projectName, String deadLine) {  
 this.projectName = projectName;  
 this.deadLine = deadLine;  
 System.out.println("Project Created : " + projectName + " , DeadLine : " + deadLine);  
 }  
  
 Project(String projectName, String deadLine, double budget) {  
 this.projectName = projectName;  
 this.deadLine = deadLine;  
 this.budget = budget;  
 System.out.println("Project Created : " + projectName + " , DeadLine : " + deadLine + " , Budget : " + budget);  
 }  
  
 @Override  
 protected void finalize() {  
 System.out.println("Project Deleted : " + projectName + " is being garbage collected ");  
 }  
}  
  
class ProjectAI extends Project {  
  
 ProjectAI(String projectName, String deadLine, double budget) {  
 super(projectName, deadLine, budget);  
 }  
  
 ProjectAI(String projectName, String deadLine) {  
 super(projectName, deadLine);  
 }  
  
 ProjectAI() {  
 super();  
 }  
}  
  
public class Q9 {  
 public static void main(String[] args) {  
  
 Project Project1 = new Project();  
 Project Project2 = new Project("Food Panda", "30th June 2025");  
 Project Project3 = new Project("Education App", "31th Dec 2025", 50000);  
 ProjectAI Project4 = new ProjectAI("AI Robot", "20th Feb 2025");  
 ProjectAI Project5 = new ProjectAI("AI Based Fraud Detection", "2 March 2026", 245000);  
 ProjectAI Project6 = new ProjectAI("AI Chatbot", "23rd September 2026", 245000);  
  
 Project1 = null;  
 Project2 = null;  
 Project3 = null;  
 Project4 = null;  
 Project5 = null;  
  
 System.gc();  
  
 try {  
 Thread.sleep(1000);  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
  
 }  
  
}

## Question-10.java

class Appliance {  
 Appliance() {  
 System.out.println("Appliance's Default Constructor");  
 }  
  
 Appliance(String type) {  
 this();  
 System.out.println("Appliance Type: " + type);  
 }  
  
 Appliance(String type, String brand) {  
 this(type);  
 System.out.println("Appliance Brand: " + brand);  
 }  
}  
  
class Electronic extends Appliance {  
 Electronic() {  
 this("General Electronic");  
 System.out.println("Electronic's Default Constructor");  
 }  
  
 Electronic(String category) {  
 super(category, "Default Brand");  
 System.out.println("Electronic Category: " + category);  
 }  
}  
  
class Laptop extends Electronic {  
 Laptop() {  
 this("No Brand", 8);  
 System.out.println("Laptop's Default Constructor");  
 }  
  
 Laptop(String brand) {  
 super("Electronic");  
 System.out.println("Laptop Brand: " + brand);  
 }  
  
 Laptop(String brand, int ram) {  
 super("Electronic");  
 System.out.println("Laptop Brand: " + brand);  
 System.out.println("Laptop RAM: " + ram + "GB");  
 }  
}  
  
public class Q10 {  
 public static void main(String[] args) {  
 System.out.println("Creating Laptop Object 1:");  
 Laptop l1 = new Laptop();  
  
 System.out.println("\nCreating Laptop Object 2:");  
 Laptop l2 = new Laptop("Dell", 16);  
 }  
}

## Question-11.java

class Tournament {  
 Tournament() {  
 System.out.println("Champions Trophy 2025 Tournament Created");  
 }  
  
 Tournament(String hostCountry) {  
 this();  
 System.out.println("Host Country: " + hostCountry);  
 }  
  
 Tournament(String hostCountry, int noOfTeams) {  
 this(hostCountry);  
 System.out.println("Number of Teams: " + noOfTeams);  
 }  
}  
  
class Team extends Tournament {  
 Team() {  
 this("Unknown Team");  
 System.out.println("Team's Default Constructor");  
 }  
  
 Team(String teamName) {  
 super("Pakistan", 8);  
 System.out.println("Participating Team: " + teamName);  
 }  
}  
  
class Player extends Team {  
 Player() {  
 this("Unknown Player", "All-Rounder");  
 System.out.println("Player's Default Constructor");  
 }  
  
 Player(String playerName) {  
 super("Pakistan Team");  
 System.out.println("Player Name: " + playerName);  
 }  
  
 Player(String playerName, String role) {  
 this(playerName);  
 System.out.println("Player Role: " + role);  
 }  
}  
  
public class Q11 {  
 public static void main(String[] args) {  
 System.out.println("Creating Player Object 1:");  
 Player p1 = new Player();  
  
 System.out.println("\nCreating Player Object 2:");  
 Player p2 = new Player("Virat Kohli", "Right Handed Batsman");  
 }  
}

## Question-12.java

import java.util.\*;  
  
class Books {  
 String title;  
 String authorName;  
 double price;  
  
 Books(String title) {  
 this.title = title;  
 authorName = "Unknown";  
 price = 0.0;  
 System.out.println("Book Title: " + title + " Author Name: " + authorName + " Book Price: " + price);  
 }  
  
 Books(String title, String authorName, double price) {  
 this.title = title;  
 this.authorName = authorName;  
 this.price = price;  
 System.out.println("Book Title: " + title + " Author Name: " + authorName + " Book Price: " + price);  
 }  
  
 Books(String title, String authorName) {  
 this.title = title;  
 this.authorName = authorName;  
 price = 0.0;  
 System.out.println("Book Title: " + title + " Author Name: " + authorName + " Book Price: " + price);  
 }  
}  
  
public class Q12 {  
 public static void main(String[] args) {  
  
 Books Book1 = new Books("Cyber Smoke", "James Overton", 500.0);  
 Books Book2 = new Books("Distro", "Henry");  
 Books Book3 = new Books("Crime and Punishment");  
  
 }  
}