**CONDITIONAL STATEMENTS**

**Exercise 1:**

1. Using while loop

public class ReverseCalculator {

public static void main(String[] args) {

int inputNumber = 27;

int reversedNumberWhile = *reverseNumberWhile*(inputNumber);

System.***out***.println("Reversed Number using while loop: " + reversedNumberWhile);

}

// Method to reverse a number using while loop

private static int reverseNumberWhile(int number) {

int reversedNumber = 0;

while (number != 0) {

int digit = number % 10;

reversedNumber = reversedNumber \* 10 + digit;

number /= 10;

}

return reversedNumber;

}

}

Output:



1. Using for loop

public class ReverseCalculator {

public static void main(String[] args) {

int inputNumber = 27;

int reversedNumberFor = *reverseNumberFor*(inputNumber);

System.***out***.println("Reversed Number using for loop: " + reversedNumberFor);

}

// Method to reverse a number using for loop

private static int reverseNumberFor(int number) {

int reversedNumber = 0;

for (; number != 0; number /= 10) {

int digit = number % 10;

reversedNumber = reversedNumber \* 10 + digit;

}

return reversedNumber;

}

}

Output:



1. Using do-while loop

public class ReverseCalculator {

public static void main(String[] args) {

int inputNumber = 27;

int reversedNumberDoWhile = *reverseNumberDoWhile*(inputNumber);

System.***out***.println("Reversed Number using do-while loop: " + reversedNumberDoWhile);

}

// Method to reverse a number using do-while loop

private static int reverseNumberDoWhile(int number) {

int reversedNumber = 0;

do {

int digit = number % 10;

reversedNumber = reversedNumber \* 10 + digit;

number /= 10;

} while (number != 0);

return reversedNumber;

}

}

Output:



**ARRAYS**

**Exercise:**

**public class EmployeeRecord {**

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

**// Salaries array**

**double salaries[] = {23500.0, 25080.0, 28760.0, 22340.0, 19890.0};**

**// Calculate average salary**

**double totalSalary = 0;**

**for (double salary : salaries) {**

**totalSalary += salary;**

**}**

**double averageSalary = totalSalary / salaries.length;**

**System.*out*.println("Average Salary: " + averageSalary);**

**// Count employees with salary greater and lesser than the average**

**int greaterThanAverage = 0;**

**int lesserThanAverage = 0;**

**for (double salary : salaries) {**

**if (salary > averageSalary) {**

**greaterThanAverage++;**

**} else if (salary < averageSalary) {**

**lesserThanAverage++;**

**}**

**}**

**System.*out*.println("Number of Employees with Salary Greater than Average: " + greaterThanAverage);**

**System.*out*.println("Number of Employees with Salary Lesser than Average: " + lesserThanAverage);**

**}**

**}**

Output:



**ENHANCED for LOOP**

**Exercise:**

**public class MarksManager {**

**private int[] marksArray = new int[5];**

**// Method to store marks in the array**

**public void storeMarks(int[] marks) {**

**if (marks.length == marksArray.length) {**

**System.*arraycopy*(marks, 0, marksArray, 0, marks.length);**

**System.*out*.println("Marks stored successfully.");**

**} else {**

**System.*out*.println("Invalid number of subjects. Expected 5 subjects.");**

**}**

**}**

**// Method to display marks from the array using enhanced for loop**

**public void displayMarks() {**

**System.*out*.println("Marks for 5 subjects:");**

**for (int mark : marksArray) {**

**System.*out*.println(mark);**

**}**

**}**

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

**// Create an instance of MarksManager**

**MarksManager marksManager = new MarksManager();**

**// Example: Store marks**

**int[] subjectMarks = {85, 90, 78, 92, 88};**

**marksManager.storeMarks(subjectMarks);**

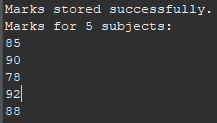
**// Example: Display marks using enhanced for loop**

**marksManager.displayMarks();**

**}**

**}**

Output:



**CONSTRUCTORS**

**Exercise:**

**class Chocolate {**

**private int barCode;**

**private String name;**

**private int weight;**

**private int cost;**

**// Constructor**

**public Chocolate() {**

**// Initialize default values**

**this.barCode = 101;**

**this.name = "Cadbury";**

**this.weight = 12;**

**this.cost = 10;**

**}**

**// Getter and Setter methods**

**public int getBarCode() {**

**return barCode;**

**}**

**public void setBarCode(int barCode) {**

**this.barCode = barCode;**

**}**

**public String getName() {**

**return name;**

**}**

**public void setName(String name) {**

**this.name = name;**

**}**

**public int getWeight() {**

**return weight;**

**}**

**public void setWeight(int weight) {**

**this.weight = weight;**

**}**

**public int getCost() {**

**return cost;**

**}**

**public void setCost(int cost) {**

**this.cost = cost;**

**}**

**}**

**public class ChocolateTester {**

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

**// Create an object of chocolate**

**Chocolate chocolate = new Chocolate();**

**// Use getter methods to display the default values**

**System.*out*.println("Default Values:");**

**System.*out*.println("Bar Code: " + chocolate.getBarCode());**

**System.*out*.println("Name: " + chocolate.getName());**

**System.*out*.println("Weight: " + chocolate.getWeight());**

**System.*out*.println("Cost: " + chocolate.getCost());**

**// Use setter methods to modify the values**

**chocolate.setBarCode(102);**

**chocolate.setName("Hershey's");**

**chocolate.setWeight(24);**

**chocolate.setCost(50);**

**// Use getter methods to display the modified values**

**System.*out*.println("\nModified Values:");**

**System.*out*.println("Bar Code: " + chocolate.getBarCode());**

**System.*out*.println("Name: " + chocolate.getName());**

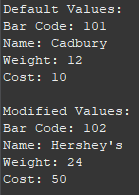
**System.*out*.println("Weight: " + chocolate.getWeight());**

**System.*out*.println("Cost: " + chocolate.getCost());**

**}**

**}**

Output:



**this KEYWORD**

**Exercise:**

**class Chocolate {**

**private int barCode;**

**private String name;**

**private double weight;**

**private double cost;**

**// Parameterized constructor**

**public Chocolate(int barCode, String name, double weight, double cost) {**

**this.barCode = barCode;**

**this.name = name;**

**this.weight = weight;**

**this.cost = cost;**

**}**

**// Default constructor**

**public Chocolate() {**

**// Initialize default values**

**this.barCode = 101;**

**this.name = "Cadbury";**

**this.weight = 12;**

**this.cost = 10;**

**}**

**// Getter and Setter methods**

**public int getBarCode() {**

**return barCode;**

**}**

**public void setBarCode(int barCode) {**

**this.barCode = barCode;**

**}**

**public String getName() {**

**return name;**

**}**

**public void setName(String name) {**

**this.name = name;**

**}**

**public double getWeight() {**

**return weight;**

**}**

**public void setWeight(double weight) {**

**this.weight = weight;**

**}**

**public double getCost() {**

**return cost;**

**}**

**public void setCost(double cost) {**

**this.cost = cost;**

**}**

**}**

**public class ChocolateTester {**

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

**// Create an object of chocolate using parameterized constructor**

**Chocolate chocolate1 = new Chocolate(101, "Cadbury", 12, 10);**

**// Use getter methods to display the values**

**System.*out*.println("Default Values:");**

**System.*out*.println("Bar Code: " + chocolate1.getBarCode());**

**System.*out*.println("Name: " + chocolate1.getName());**

**System.*out*.println("Weight: " + chocolate1.getWeight());**

**System.*out*.println("Cost: " + chocolate1.getCost());**

**// Create another object of chocolate using default constructor**

**Chocolate chocolate2 = new Chocolate();**

**// Use setter methods to modify the values**

**chocolate2.setBarCode(102);**

**chocolate2.setName("Hershey's");**

**chocolate2.setWeight(24);**

**chocolate2.setCost(50);**

**// Use getter methods to display the modified values**

**System.*out*.println("\nModified Values:");**

**System.*out*.println("Bar Code: " + chocolate2.getBarCode());**

**System.*out*.println("Name: " + chocolate2.getName());**

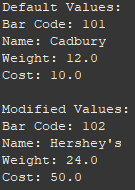
**System.*out*.println("Weight: " + chocolate2.getWeight());**

**System.*out*.println("Cost: " + chocolate2.getCost());**

**}**

**}**

Output:



**INHERITANCE**

**Exercise:**

**class Employee {**

**private int empId;**

**private String name;**

**private double salary;**

**// Constructors, getters, and setters for empId, name, and salary**

**public double getSalary() {**

**return salary;**

**}**

**public void setSalary(double salary) {**

**this.salary = salary;**

**}**

**public int getEmpId() {**

**return empId;**

**}**

**public void setEmpId(int empId) {**

**this.empId = empId;**

**}**

**public String getName() {**

**return name;**

**}**

**public void setName(String name) {**

**this.name = name;**

**}**

**}**

**class PermanentEmployee extends Employee {**

**private double basicPay;**

**private double hra;**

**private int experience;**

**// Constructors, getters, and setters for basicPay, hra, and experience**

**public double getBasicPay() {**

**return basicPay;**

**}**

**public void setBasicPay(double basicPay) {**

**this.basicPay = basicPay;**

**}**

**public double getHra() {**

**return hra;**

**}**

**public void setHra(double hra) {**

**this.hra = hra;**

**}**

**public int getExperience() {**

**return experience;**

**}**

**public void setExperience(int experience) {**

**this.experience = experience;**

**}**

**public void calculateSalary() {**

**double variableComponent = 0;**

**if (experience < 3) {**

**variableComponent = 0;**

**} else if (experience >= 3 && experience < 5) {**

**variableComponent = 0.07 \* basicPay;**

**} else if (experience >= 5 && experience < 10) {**

**variableComponent = 0.05 \* basicPay;**

**} else if (experience >= 10) {**

**variableComponent = 0.12 \* basicPay;**

**}**

**setSalary(variableComponent + basicPay + hra);**

**}**

**}**

**class ContractEmployee extends Employee {**

**private double wages;**

**private int hours;**

**// Constructors, getters, and setters for wages and hours**

**public double getWages() {**

**return wages;**

**}**

**public void setWages(double wages) {**

**this.wages = wages;**

**}**

**public int getHours() {**

**return hours;**

**}**

**public void setHours(int hours) {**

**this.hours = hours;**

**}**

**public void calculateSalary() {**

**setSalary(wages \* hours);**

**}**

**}**

**public class EmployeeRecords {**

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

**// Create an instance of PermanentEmployee**

**PermanentEmployee permanentEmployee = new PermanentEmployee();**

**// Populate the object with the inputs**

**permanentEmployee.setName("Anil");**

**permanentEmployee.setEmpId(101);**

**permanentEmployee.setBasicPay(10000);**

**permanentEmployee.setHra(1500);**

**permanentEmployee.setExperience(4);**

**// Invoke the calculateSalary method**

**permanentEmployee.calculateSalary();**

**// Display the salary of the permanent employee**

**System.*out*.println("Permanent Employee: Your salary is: " + permanentEmployee.getSalary());**

**// Create an instance of ContractEmployee**

**ContractEmployee contractEmployee = new ContractEmployee();**

**// Populate the object with the inputs**

**contractEmployee.setName("Ankit");**

**contractEmployee.setEmpId(102);**

**contractEmployee.setWages(500);**

**contractEmployee.setHours(10);**

**// Invoke the calculateSalary method**

**contractEmployee.calculateSalary();**

**// Display the salary of the contract employee**

**System.*out*.println("Contract Employee: Your salary is: " + contractEmployee.getSalary());**

**}**

**}**

Output:



**POLYMORPHISM**

**Exercise 1:**

**class PlayerRating {**

**private int playerPosition;**

**private String playerName;**

**private double criticOneRating;**

**private double criticTwoRating;**

**private double criticThreeRating;**

**private double averageRating;**

**private char category;**

**public PlayerRating(int playerPosition, String playerName) {**

**this.playerPosition = playerPosition;**

**this.playerName = playerName;**

**}**

**public void calculateAverageRating(double criticOneRating, double criticTwoRating) {**

**this.averageRating = (criticOneRating + criticTwoRating) / 2;**

**calculateCategory();**

**}**

**public void calculateAverageRating(double criticOneRating, double criticTwoRating, double criticThreeRating) {**

**this.averageRating = (criticOneRating + criticTwoRating + criticThreeRating) / 3;**

**calculateCategory();**

**}**

**private void calculateCategory() {**

**if (averageRating > 8) {**

**category = 'A';**

**} else if (averageRating > 5 && averageRating <= 8) {**

**category = 'B';**

**} else if (averageRating > 0 && averageRating <= 5) {**

**category = 'C';**

**}**

**}**

**public void display() {**

**System.*out*.println("The player name is " + playerName);**

**System.*out*.println("The player position is " + playerPosition);**

**System.*out*.println("The average rating is " + averageRating);**

**System.*out*.println("The category is " + category);**

**}**

**}**

**public class PlayerRatingTester {**

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

**// Test case with two critics**

**PlayerRating player1 = new PlayerRating(1, "Beckham");**

**player1.calculateAverageRating(9.0, 9.9);**

**player1.display();**

**System.*out*.println();**

**// Test case with three critics**

**PlayerRating player2 = new PlayerRating(1, "Oscar");**

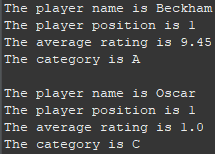
**player2.calculateAverageRating(1, 1, 1);**

**player2.display();**

**}**

**}**

Output:



**Exercise 2:**

**class Registration {**

**private String customerName;**

**private String passportNo;**

**private int voterId;**

**private int licenseNo;**

**private String panCardNo;**

**private long[] telephoneNo;**

**public Registration(String customerName, String passportNo, long[] telephoneNo) {**

**this.customerName = customerName;**

**this.passportNo = passportNo;**

**this.telephoneNo = telephoneNo;**

**}**

**public Registration(String customerName, int licenseNo, String panCardNo, long[] telephoneNo) {**

**this.customerName = customerName;**

**this.licenseNo = licenseNo;**

**this.panCardNo = panCardNo;**

**this.telephoneNo = telephoneNo;**

**}**

**public Registration(String customerName, int voterId, int licenseNo, long[] telephoneNo) {**

**this.customerName = customerName;**

**this.voterId = voterId;**

**this.licenseNo = licenseNo;**

**this.telephoneNo = telephoneNo;**

**}**

**public Registration(String customerName, String panCardNo, int voterId, long[] telephoneNo) {**

**this.customerName = customerName;**

**this.panCardNo = panCardNo;**

**this.voterId = voterId;**

**this.telephoneNo = telephoneNo;**

**}**

**public String getCustomerName() {**

**return customerName;**

**}**

**public String getPassportNo() {**

**return passportNo;**

**}**

**public int getVoterId() {**

**return voterId;**

**}**

**public int getLicenseNo() {**

**return licenseNo;**

**}**

**public String getPanCardNo() {**

**return panCardNo;**

**}**

**public long[] getTelephoneNo() {**

**return telephoneNo;**

**}**

**public void displayRegistrationDetails() {**

**System.*out*.println("Congratulations " + customerName + "!!! you have been successfully registered for our services with the following details:");**

**if (passportNo != null) {**

**System.*out*.println("Passport number: " + passportNo);**

**} else {**

**System.*out*.println("License number: " + licenseNo);**

**System.*out*.println("Pan card number: " + panCardNo);**

**}**

**System.*out*.println("Phone numbers:");**

**for (long phoneNumber : telephoneNo) {**

**System.*out*.println(phoneNumber);**

**}**

**}**

**}**

**public class RegistrationTester {**

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

**// Test case 1**

**Registration user1 = new Registration("Kevin", "MN9891IN", new long[]{9452425421L, 7676765252L});**

**user1.displayRegistrationDetails();**

**System.*out*.println();**

**// Test case 2**

**Registration user2 = new Registration("Julias", 123, "PN7878", new long[]{2345615451L, 6763562562L});**

**user2.displayRegistrationDetails();**

**System.*out*.println();**

**// Test case 3**

**Registration user3 = new Registration("Jammy", 45453, 765, new long[]{9634524353L, 9887373737L});**

**user3.displayRegistrationDetails();**

**System.*out*.println();**

**// Test case 4**

**Registration user4 = new Registration("Rose", "PN8934", 34356, new long[]{9867456367L, 7645367356L});**

**user4.displayRegistrationDetails();**

**}**

**}**

Output:

