Day 2 Shrayanth S 25/06/2025

Problem Statement 1: Age

Declare an int variable myAge and assign your age to it.
Write expressions using comparison operators to check if:

- myAge == 25
- myAge > 18
- myAge <= 65
- myAge != 30
  Print the Boolean result of each expression using System.out.println().

Algorithm :

1. Start
2. Declare an integer variable myAge and assign a value
3. Print the result of myAge == 25
4. Print the result of myAge > 18
5. Print the result of myAge <= 65
6. Print the result of myAge != 30
7. End

Pseudocode :

START

SET myAge = 22

PRINT myAge == 25

PRINT myAge > 18

PRINT myAge <= 65

PRINT myAge != 30

END

Java Code :

AgeChecker.java

Share Run

```java
public class AgeChecker {
    public static void main(String[] args) {
        int myAge = 22; // Replace with your actual age

        System.out.println("myAge == 25: " + (myAge == 25));
        System.out.println("myAge > 18: " + (myAge > 18));
        System.out.println("myAge <= 65: " + (myAge <= 65));
        System.out.println("myAge != 30: " + (myAge != 30));
    }
}
```

Test Cases :

Input myAge	Output
22	false, true, true, true
25	true, true, true, true
30	false, true, true, false
66	false, true, false, true

Output:

Output

```
myAge == 25: false
myAge > 18: true
myAge <= 65: true
myAge != 30: true

=== Code Execution Successful ===
```

Problem Statement 1.2: Login Credentials

Check if entered username and password match the predefined values.

- **Algorithm:**
  - Start
  - Declare username and password variables with predefined values
  - Take username and password input from user
  - Check if both match using logical AND
  - If match, print 'Login Successful'
  - Else, print 'Login Failed'
  - End
- **Pseudocode:**
  - START
  - SET validUsername = 'admin', validPassword = 'password123'
  - GET inputUsername, inputPassword
  - IF inputUsername == validUsername AND inputPassword == validPassword THEN
  - PRINT 'Login Successful'
  - ELSE
  - PRINT 'Login Failed'
  - END IF
  - END

Java Code:

LoginCredentials.java

Share Run

```java
import java.util.Scanner;
public class LoginCredentials {
    public static void main(String[] args) {
        String validUsername = "admin";
        String validPassword = "password123";

        Scanner sc = new Scanner(System.in);
        System.out.print("Enter username: ");
        String inputUsername = sc.nextLine();

        System.out.print("Enter password: ");
        String inputPassword = sc.nextLine();

        if (inputUsername.equals(validUsername) && inputPassword
            .equals(validPassword)) {
            System.out.println("Login Successful");
        } else {
            System.out.println("Login Failed");
        }
    }
}
```

Test Cases:

- Input: admin, password123 → Output: Login Successful
- Input: user, pass → Output: Login Failed

OUTPUT :

```
Output

Enter username: admin
Enter password: password123
Login Successful

=== Code Execution Successful ===
```

```
Output

Enter username: admin
Enter password: pass
Login Failed

=== Code Execution Successful ===
```

```
Output

Enter username: ram
Enter password: password123
Login Failed

=== Code Execution Successful ===
```

Problem Statement 1.3: Number Range

Check a number against multiple conditions (range).

- **Algorithm:**
  - Start
  - Input a number
  - Check if number > 10 and < 20
  - Check if number < 5 or > 100
  - Print the results
  - End

- **Pseudocode:**
  - START
  - GET number
  - IF number > 10 AND number < 20 THEN PRINT 'In Range 10-20'
  - IF number < 5 OR number > 100 THEN PRINT 'Out of Range'
  - END

Java Code:

NumberRange.java

Share Run

```java
import java.util.Scanner;
public class NumberRange {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int number = sc.nextInt();

        if (number > 10 && number < 20) {
            System.out.println("In Range 10-20");
        }
        if (number < 5 || number > 100) {
            System.out.println("Out of Range");
        }
    }
}
```

- **Test Cases:**
  - Input: 15 → Output: In Range 10-20
  - Input: 3 → Output: Out of Range

OUTPUT :

Output

```
Enter a number: 14
In Range 10-20

=== Code Execution Successful ===
```

Output

```
Enter a number: 7777
Out of Range

=== Code Execution Successful ===
```

Problem Statement 1.4: Operator Precedence Challenge

Evaluate the Boolean expression step-by-step.

- **Algorithm:**
  - Start
  - Break down expression: 5 + 3 * 2 > 10 && !(7 == 7)
  - Evaluate multiplication: 3 * 2 = 6
  - Then addition: 5 + 6 = 11
  - Then comparison: 11 > 10 = true
  - Evaluate (7 == 7) = true, !(true) = false
  - Final: true && false = false
  - End

Pseudocode:

- START
- Evaluate 3 * 2 = 6
- Evaluate 5 + 6 = 11
- Evaluate 11 > 10 → true
- Evaluate 7 == 7 → true → !true = false
- Evaluate true && false → false
- PRINT final result: false
- END

Java Code:

OperatorPrecedence.java

Share Run

```java

public class OperatorPrecedence {
    public static void main(String[] args) {
        boolean result = 5 + 3 * 2 > 10 && !(7 == 7);
        System.out.println("Result: " + result); // false
    }
}

```

- **Test Cases:**

Input: N/A → Output: Result: false

OUTPUT:

```
Output
Result: false

=== Code Execution Successful ===
```

Problem 2.1: Positive, Negative, or Zero

- Get an integer input from the user using Scanner.
- Use an if-else if-else structure to:
    - Print "Positive" if the number is greater than 0
    - Print "Negative" if the number is less than 0
    - Print "Zero" if the number is exactly 0

Algorithm

1. Start
2. Declare a variable num
3. Take integer input from user using Scanner
4. If num > 0, print "Positive"
5. Else if num < 0, print "Negative"
6. Else, print "Zero"
7. End

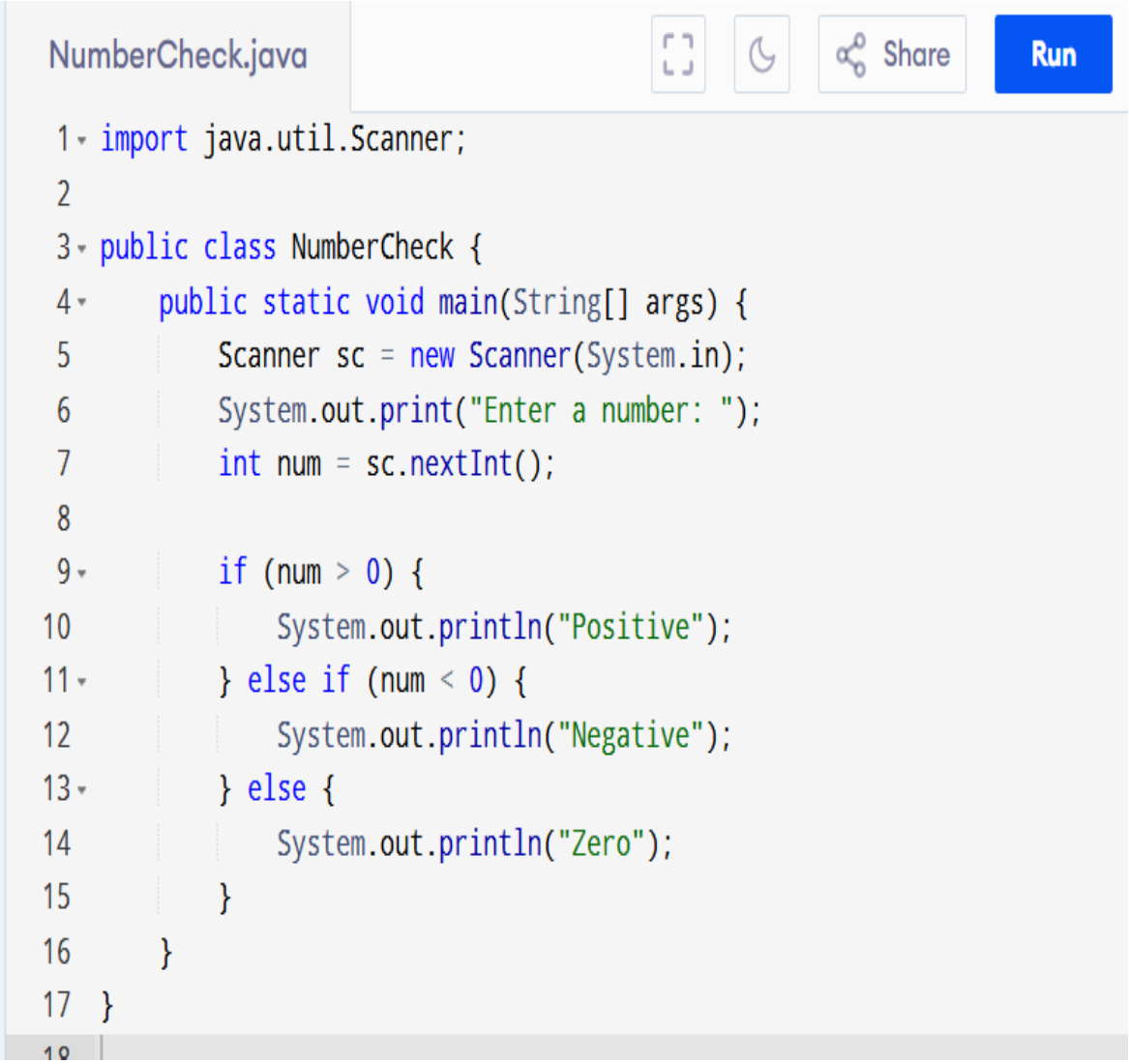
Pseudocode

START

GET number from user

```
IF number > 0 THEN
   PRINT "Positive"
ELSE IF number < 0 THEN
   PRINT "Negative"
ELSE
   PRINT "Zero"

END
```

Java Code



Test Cases

Input Output

12 Positive

-8 Negative

0 Zero

OUTPUT:

```
Output
Enter a number: -8
Negative

=== Code Execution Successful ===
```

```
Output
Enter a number: 12
Positive

=== Code Execution Successful ===
```

```
Output
Enter a number: 0
Zero

=== Code Execution Successful ===
```

Problem Statement 2.2: Driving Eligibility

Check if the user is eligible to drive (age ≥ 18).

- **Algorithm:**
  - Start
  - Input age
  - If age >= 18, print 'Eligible to drive'
  - Else, print 'Not eligible'
  - End

- **Pseudocode:**
  - START
  - GET age
  - IF age >= 18 THEN PRINT 'Eligible to drive'
  - ELSE PRINT 'Not eligible'
  - END

Java Code:

DrivingEligibility.java

Share Run

```java
import java.util.Scanner;
public class DrivingEligibility {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter age: ");
        int age = sc.nextInt();

        if (age >= 18) {
            System.out.println("Eligible to drive");
        } else {
            System.out.println("Not eligible");
        }
    }
}
```

- **Test Cases:**
  - Input: 20 → Output: Eligible to drive
  - Input: 17 → Output: Not eligible

Output:

Output

```
Enter age: 21
Eligible to drive

=== Code Execution Successful ===
```

Output

```
Enter age: 10
Not eligible

=== Code Execution Successful ===
```

Problem Statement 2.3: Simple Calculator

Problem Statement:
Take two numbers and an operator (+, -, *, /) and print the result. Handle divide by zero.

-
- **Algorithm:**
  - Start
  - Input two numbers
  - Input operator
  - Use switch to perform operation
  - Handle division by zero
  - Print result
  - End
- **Pseudocode:**
  - START
  - GET num1, num2, operator
  - SWITCH operator
  - CASE '+': PRINT num1 + num2
  - CASE '-': PRINT num1 - num2
  - CASE '*': PRINT num1 * num2
  - CASE '/': IF num2 == 0 THEN PRINT 'Cannot divide by zero' ELSE PRINT num1 / num2
  - DEFAULT: PRINT 'Invalid operator'
  - END SWITCH
  - END

Java Code:

SimpleCalculator.java

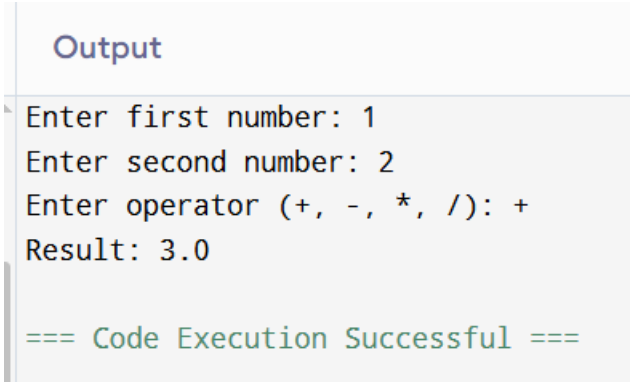
Share Run

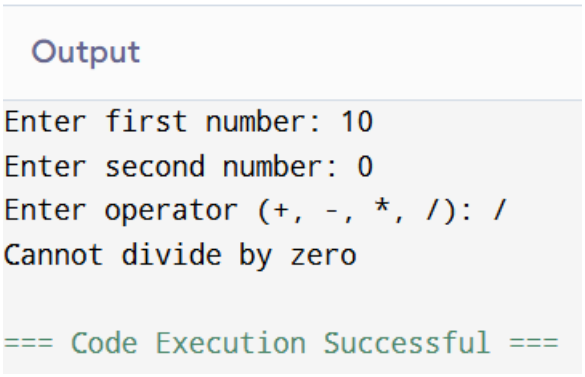
```
        System.out.print("Enter first number: ");
        double num1 = sc.nextDouble();
        System.out.print("Enter second number: ");
        double num2 = sc.nextDouble();
        System.out.print("Enter operator (+, -, *, /): ");
        char operator = sc.next().charAt(0);

        switch (operator) {
            case '+': System.out.println("Result: " + (num1 + num2
                )); break;
            case '-': System.out.println("Result: " + (num1 - num2
                )); break;
            case '*': System.out.println("Result: " + (num1 * num2
                )); break;
            case '/':
                if (num2 == 0)
                    System.out.println("Cannot divide by zero");
                else
                    System.out.println("Result: " + (num1 / num2));
                break;
            default: System.out.println("Invalid operator");
        }
    }
}
```

Test Cases:

- Input: 10, 5, + → Output: Result: 15.0
- Input: 10, 0, / → Output: Cannot divide by zero

Output :





Problem Statement 2.4: Movie Ticket Pricing

Determine ticket price based on age and student status.

- **Algorithm:**

- Start
- Get age and student status as input
- If age < 5 or age > 65 → ticket = $5
- Else if age between 5 and 18 **and** student is true → ticket = $8
- Else → ticket = $12
- Print the ticket price
- End

- **Pseudocode:**

  - START
  - GET age
  - GET studentStatus
  - IF age < 5 OR age > 65 THEN
  - PRINT "$5"
  - ELSE IF age >= 5 AND age <= 18 AND studentStatus == true THEN

- PRINT "$8"
- ELSE
- PRINT "$12"

END

Java Code:

MovieTicketPrice.java

```java
import java.util.Scanner;

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

        System.out.print("Enter your age: ");
        int age = sc.nextInt();

        System.out.print("Are you a student (true/false): ");
        boolean isStudent = sc.nextBoolean();

        if (age < 5 || age > 65) {
            System.out.println("Ticket Price: $5");
        } else if (age >= 5 && age <= 18 && isStudent) {
            System.out.println("Ticket Price: $8");
        } else {
            System.out.println("Ticket Price: $12");
        }
    }
}
```

- **Test Cases:**
  - Input: 70, false → Output: Ticket Price: $5
  - Input: 15, true → Output: Ticket Price: $8
  - Input: 15, false → Output: Ticket Price: $12

Output :

Output

```
Enter your age: 70
Are you a student (true/false): false
Ticket Price: $5

=== Code Execution Successful ===
```

Output

```
Enter your age: 15
Are you a student (true/false): true
Ticket Price: $8

=== Code Execution Successful ===
```

Problem Statement 3.1: Day of the Week

Print day of the week using switch (1–7), handle invalid input.

- **Algorithm:**
  - Start
  - Input day (1 to 7)
  - Use switch to print day
  - Default case for invalid input
  - End
- **Pseudocode:**
  - START
  - GET day
  - SWITCH day
  - 1: PRINT 'Monday'
  - 2: PRINT 'Tuesday'
  - 3: PRINT 'Wednesday'
  - 4: PRINT 'Thursday'
  - 5: PRINT 'Friday'
  - 6: PRINT 'Saturday'
  - 7: PRINT 'Sunday'
  - DEFAULT: PRINT 'Invalid input'
  - END

Java Code:

DayOfWeek.java

Share Run

```java
import java.util.Scanner;
public class DayOfWeek {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number (1-7): ");
        int day = sc.nextInt();

        switch(day) {
            case 1: System.out.println("Monday"); break;
            case 2: System.out.println("Tuesday"); break;
            case 3: System.out.println("Wednesday"); break;
            case 4: System.out.println("Thursday"); break;
            case 5: System.out.println("Friday"); break;
            case 6: System.out.println("Saturday"); break;
            case 7: System.out.println("Sunday"); break;
            default: System.out.println("Invalid input");
        }
    }
}

```

Test Cases:

- Input: 5 → Output: Friday
- Input: 8 → Output: Invalid input

- **Output:**

```
Output

Enter a number (1-7): 8
Invalid input

=== Code Execution Successful ===
```

```
Output

Enter a number (1-7): 5
Friday

=== Code Execution Successful ===
```

Problem Statement 3.2: Simple Menu Selection (ATM Simulation)

Problem Statement:
Simulate ATM menu using switch. 1=Check Balance, 2=Withdraw, 3=Deposit, 4=Exit.

- **Algorithm:**

  - Start
  - Display menu and input choice
  - Use switch to process option
  - Default for invalid option
  - End

- **Pseudocode:**

  - START
  - GET choice
  - SWITCH choice
  - 1: PRINT 'Checking Balance'
  - 2: PRINT 'Withdrawing Money'
  - 3: PRINT 'Depositing Money'
  - 4: PRINT 'Exiting'
  - DEFAULT: PRINT 'Invalid input'
  - END

Java Code:

ATMMenu.java

Share Run

```java
import java.util.Scanner;
public class ATMMenu {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("1: Check Balance\n2: Withdraw\n3:
            Deposit\n4: Exit");
        System.out.print("Enter your choice: ");
        int choice = sc.nextInt();

        switch (choice) {
            case 1: System.out.println("Checking Balance"); break;
            case 2: System.out.println("Withdrawing Money"); break;
            case 3: System.out.println("Depositing Money"); break;
            case 4: System.out.println("Exiting"); break;
            default: System.out.println("Invalid input");
        }
    }
}
```

- **Test Cases:**
  - Input: 1 → Output: Checking Balance
  - Input: 5 → Output: Invalid input

- **Output:**

Output

```
1: Check Balance
2: Withdraw
3: Deposit
4: Exit
Enter your choice: 1
Checking Balance

=== Code Execution Successful ===
```

Output

```
1: Check Balance
2: Withdraw
3: Deposit
4: Exit
Enter your choice: 5
Invalid input

=== Code Execution Successful ===
```

Problem Statement 3.3: Grade Remarks (Why switch is not ideal)

Problem Statement:
Use if-else ladder to print grade remarks based on score (0–100).

- **Algorithm:**
  - Start
  - Input score
  - Use if-else to print remarks
  - End
- **Pseudocode:**
  - START
  - GET score
  - IF score >= 90 THEN PRINT 'Excellent'
  - ELSE IF score >= 80 THEN PRINT 'Very Good'
  - ELSE IF score >= 70 THEN PRINT 'Good'
  - ELSE IF score >= 60 THEN PRINT 'Pass'
  - ELSE PRINT 'Fail'
  - END

Java Code:

GradeRemarks.java

Share Run

```java
import java.util.Scanner;
public class GradeRemarks {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter score (0-100): ");
        int score = sc.nextInt();

        if (score >= 90) {
            System.out.println("Excellent");
        } else if (score >= 80) {
            System.out.println("Very Good");
        } else if (score >= 70) {
            System.out.println("Good");
        } else if (score >= 60) {
            System.out.println("Pass");
        } else {
            System.out.println("Fail");
        }
    }
}

```

Test Cases:

- Input: 95 → Output: Excellent
- Input: 45 → Output: Fail

Output:

Output

```
Enter score (0-100): 45
Fail

=== Code Execution Successful ===
```

Output

```
Enter score (0-100): 95
Excellent

=== Code Execution Successful ===
```