

CHANDRAKANTH HV

PROBLEM STATEMENT 1 : EVERYDAY DECISIONS

ALGORITHM :

Step 1: Start
Step 2: Create a scanner object to read input
Step 3: Ask the user: Is it raining?
→ Read input as boolean (true/false)
Step 4: If it's raining, print "Carry an umbrella."
Step 5: Ask the user: Are you hungry?
→ Read input as boolean
Step 6: If hungry, print "Eat some food."
Step 7: Ask the user for the battery level
→ Read input as integer
Step 8: If battery level < 20, print "Charge the phone."
Step 9: Close the scanner
Step 10: End

PSEUDO CODE :

START

INPUT is Raining (true/false)

IF is Raining THEN

PRINT "Carry an umbrella."

INPUT is Hungry (true/false)

IF is Hungry THEN

PRINT "Eat some food."

INPUT battery Level (integer)

IF battery Level < 20 THEN

PRINT "Charge the phone."

CLOSE input

END

CODE :

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Is it raining? (true/false): ");

        boolean isRaining = scanner.nextBoolean();

        if (isRaining) {

            System.out.println("Carry an umbrella.");

        }

        System.out.print("Are you hungry? (true/false): ");

        boolean isHungry = scanner.nextBoolean();

        if (isHungry) {

            System.out.println("Eat some food.");

        }

        System.out.print("Enter battery level: ");

        int batteryLevel = scanner.nextInt();

        if (batteryLevel < 20) {

            System.out.println("Charge the phone.");

        }

        scanner.close(); } }
```

| TEST CASE | INPUT |
|-----------|--|
| TC1 | <pre> Are you hungry? (yes/no): yes You should eat. Are you tired? (yes/no): yes You should take rest or sleep. Do you have homework? (yes/no): no You can relax or do something fun. </pre> |
| TC2 | <pre> Are you hungry? (yes/no): no You don't need to eat right now. Are you tired? (yes/no): no You have enough energy. Do you have homework? (yes/no): no You can relax or do something fun. </pre> |
| TC3 | <pre> Are you hungry? (yes/no): yes You should eat. Are you tired? (yes/no): yes You should take rest or sleep. Do you have homework? (yes/no): yes You should study now. </pre> |

OBSERVATION :

- This program demonstrates the use of boolean and integer inputs.
- It applies simple conditional logic using `if` statements.
- It makes decisions based on real-world scenarios (rain, hunger, battery).
- Inputs are taken using the Scanner class.
- It helps beginners understand how to use boolean values and make decisions in Java.

PROBLEM STATEMENT 2 : smart home device.

PSEUDO CODE :

Start

create scanner to read user input

prompt user to enter current time

read time input

prompt user to enter current temperature

read temperature input

if time equals 19 then

turn on lights

else

print "no need to turn on lights yet"

if temperature greater than 30 then

turn on air conditioner

else

print "temperature is comfortable. Ac not needed"

close scanner

End

ALGORITHM :

1. Start the program.
2. Initialize the scanner to get user input.
3. Ask and read the current time (as an integer).
4. Ask and read the current temperature (as an integer).
5. If the time is exactly 19 (7 PM), turn on the lights.
6. Else, print a message that lights are not needed.
7. If the temperature is greater than 30°C, turn on the air conditioner.
8. Else, print that AC is not needed.
9. Close the scanner.

10. End the program.

CODE :

```
import java.util.Scanner;
public class thoughts {
    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the room temperature in (C):");
        int roomTemperature = scanner.nextInt();
        if (roomTemperature < 20)
        {
            System.out.println("Room is cold. Turning on the heater.");
        }
        else
        {
            System.out.println("Room is normal");
        }

        System.out.print("Enter the time in mintutes:");
        int Motion = scanner.nextInt();
        if (Motion > 30)
        {
            System.out.println("No activity detected. Turning off the lights.");
        }
        else {
            System.out.println("Motion detected.");
        }
        scanner.close();
    }
}
```

| Test cases | Input | Expected Output |
|------------|--|--|
| TC1 | Enter the Room temperature=25 Enter the time = 35 | Enter the room temperature in (C):25 Room is normal Enter the time in mintutes:35 No activity detected. Turning off the lights. |

| | | |
|-----|---|---|
| TC2 | Enter the Room temperature=0 Enter the time = 20 | <pre> Enter the room temperature in (C):0 Room is cold. Turning on the heater. Enter the time in minutes:20 Motion detected. </pre> |
| TC3 | Enter the Room temperature= -6 Enter the time = 0 | <pre> Enter the room temperature in (C):-6 Room is cold. Turning on the heater. Enter the time in minutes:0 Motion detected. </pre> |

OBSERVATION :

The Smart Home program uses user inputs for time and temperature to simulate basic automation. It turns on the lights if the time is exactly 19 (7 PM) and activates the air conditioner if the temperature exceeds 30°C. The logic is simple, clear, and demonstrates effective use of conditional statements to model real-world smart home behavior.

PROBLEM STATEMENT 3 : Age Checker.

PSEUDO CODE :

Start

create scanner to read input

ask "enter your age"

read age as integer

check if age is equal to 25 print result

check if age is greater than 18 print result

check if age is less than or equal to 65 print result

check if age is not equal to 30 print result

close scanner

End

ALGORITHM :

11. Start the program.
12. Create a Scanner to take user input.
13. Prompt the user to enter their age.
14. Read the input and store it in myAge.
15. Print whether myAge == 25.
16. Print whether myAge > 18.
17. Print whether myAge <= 65.
18. Print whether myAge != 30.
19. Close the Scanner.
20. End the program.

CODE :

```
import java.util.Scanner;
```

```
public class main2  
{
```

```

public static void main(String[] args)
{

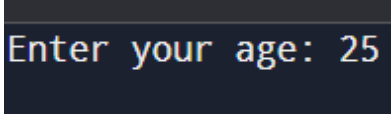
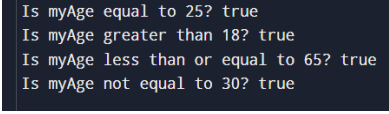
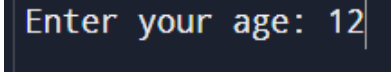
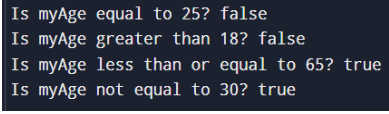
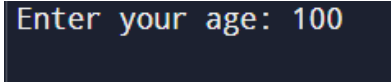
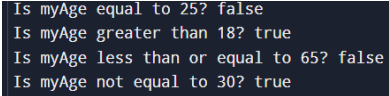
    Scanner scanner = new Scanner(System.in);

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

    System.out.println("Is my age equal to 25? " + (myAge == 25));
    System.out.println("Is my age greater than 18? " + (myAge > 18));
    System.out.println("Is my age less than or equal to 65? " + (myAge <= 65));
    System.out.println("Is my age not equal to 30? " + (myAge != 30));

    scanner.close();
}
}

```

| Test cases | Input | Expected Output |
|------------|--|---|
| TC1 |  |  |
| TC2 |  |  |
| TC3 |  |  |

OBSERVATION :

The program checks a user's age against multiple conditions using comparison operators. It clearly demonstrates how to evaluate and print Boolean results in Java, making it useful for learning basic decision-making and logical expressions.

PROBLEM STATEMENT 4 : LOGIN CREDENTIALS

ALGORITHM :

1. Start the program.
2. Store the correct username and password.
3. Prompt the user to input username.
4. Read and store entered username.
5. Prompt the user to input a password.
6. Read and store the entered password.
7. Compare both entered values with the correct credentials.
8. If both match, print "Login successful!".
9. Else, print "Invalid username or password".
10. Close the Scanner.
11. End the program.

PSEUDO CODE :

Start

set correct username = "admin"

set correct password = "password123"

ask user to enter username

read entered username

ask user to enter password

read entered password

if entered username equals correct username and entered password equals correct password
then

print "login successful!"

else

print "invalid username or password"close scanner

End

CODE :

```
import java.util.Scanner;

public class LoginCheck {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        String username = "admin";

        String password = "password123";

        System.out.print("Enter username: ");

        String enteredUsername = scanner.nextLine();

        System.out.print("Enter password: ");

        String enteredPassword = scanner.nextLine();

        if (correctUsername.equals(enteredUsername) &&
correctPassword.equals(enteredPassword))

        {

            System.out.println("Login successful!");

            } else if (correctUsername.equals(enteredUsername) &&
!correctPassword.equals(enteredPassword))

        {

            System.out.println("Password is incorrect.");

        }

        else if (!correctUsername.equals(enteredUsername) &&
correctPassword.equals(enteredPassword))

        {

            System.out.println("Username is incorrect.");

        } else {

            System.out.println("Both username and password are incorrect.");

        }

    }

}
```

```

    }
    scanner.close();
}
}

```

| Test cases | Input | Expected Output |
|------------|--|--|
| TC1 | <pre> Enter username: admin Enter password: password123 </pre> | <pre> Enter username: admin Enter password: password123 Login successful! </pre> |
| TC2 | <pre> Enter username: admin Enter password: sfhjk </pre> | <pre> Enter username: admin Enter password: sfhjk Invalid username or password. </pre> |
| TC3 | <pre> Enter username: sdfgj Enter password: password123 </pre> | <pre> Enter username: sdfgj Enter password: password123 Invalid username or password. </pre> |

OBSERVATION :

The program checks login credentials using string comparison and logical operators. It effectively demonstrates how to validate user input and control access, making it a good example of basic authentication logic in Java.

PROBLEM STATEMENT 6 : NUMBER RANGE

ALGORITHM :

1. Start the program.
2. Create a Scanner object.
3. Prompt the user to enter a number.
4. Read the input and store it in num.
5. Check if the number is greater than 10 **AND** less than 20.
6. Check if the number is less than 5 **OR** greater than 100.
7. Print both Boolean results.
8. Close the Scanner.
9. End the program.

PSEUDO CODE :

START

CREATE scanner to read input

PROMPT user to "Enter a number"

READ number into variable num

SET condition1 = (num > 10 AND num < 20)

SET condition2 = (num < 5 OR num > 100)

IF condition1 is true THEN

PRINT "The number satisfies: Greater than 10 AND less than 20."

ELSE IF condition2 is true THEN

PRINT "The number satisfies: Less than 5 OR greater than 100."

ELSE

PRINT "The number does not satisfy any condition."

CLOSE scanner

END

CODE :

```
import java.util.Scanner;

public class NumberRangeCheck {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = scanner.nextInt();
        boolean condition1 = (num > 10 && num < 20);
        boolean condition2 = (num < 5 || num > 100);
        if (condition1) {
            System.out.println("The number satisfies: Greater than 10 AND less than 20.");
        } else if (condition2) {
            System.out.println("The number satisfies: Less than 5 OR greater than 100.");
        } else {
            System.out.println("The number does not satisfy any condition.");
        }

        scanner.close();
    }
}
```

| Test cases | Input | Expected Output |
|------------|---------------------|---|
| TC1 | Enter the number=5 | Enter a number: 5 The number does not satisfy any condition. |
| TC2 | Enter the number=11 | Enter a number: 11 The number satisfies: Greater than 10 AND less than 20. |
| TC3 | Enter the number=72 | Enter a number: 72 The number does not satisfy any condition. |

OBSERVATION :

The program takes an integer input from the user.

It checks the number against two logical conditions using AND (&&) and OR (||) operators:

Condition 1: checks if the number is between 10 and 20
Condition 2: checks if the number is less than 5 or greater than 100.

It uses `if-else if-else` structure to print only the first condition that is true.

If neither condition is true, it displays a message saying the number doesn't satisfy any condition
Demonstrates the use of compound logical expressions in Java using boolean logic.

Problem Statement : Get an integer input from the user using Scanner

Pseudo code :

Start

create scanner to read input

ask user to enter an integer

read number

if number > 0 then

print "positive"

else if number < 0 then

print "negative"

else

print "zero"

close scanner

End

Algorithm: steps

1. Start the program
2. Create a Scanner for input
3. Prompt the user to enter an integer
4. Read and store the number
5. If number > 0, print "Positive"
6. Else if number < 0, print "Negative"
7. Else, print "Zero"
8. Close the Scanner
9. End the program

Code :

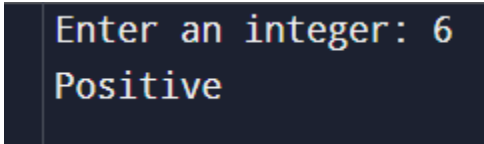
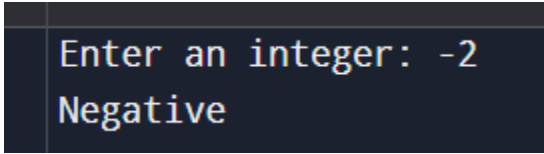
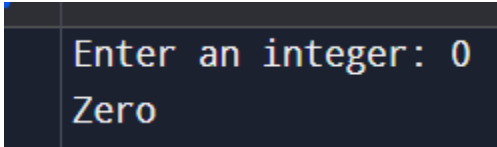
```
import java.util.Scanner;

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

        System.out.print("Enter an integer: ");
        int number = scanner.nextInt();

        if (number > 0) {
            System.out.println("Positive");
        } else if (number < 0) {
            System.out.println("Negative");
        } else {
            System.out.println("Zero");
        }

        scanner.close();
    }
}
```

| Test cases | Input | Expected Output |
|------------|------------------------|---|
| TC1 | Enter the integer : 6 |  |
| TC2 | Enter the integer : -2 |  |
| TC3 | Enter the integer : 0 |  |

Observation

The program takes an integer input and uses if-else statements to check whether the number is positive, negative, or zero. It clearly demonstrates basic number classification using conditional logic.

Problem Statement : Ask the user to input their age..

Pseudo code :

Start

create scanner to read input

ask user to enter age

read age

if age \geq 18 then

print "you are eligible to drive"

else

print "you are not eligible to drive"

close scanner

End

Algorithm: steps

1. Start the program
2. Create a Scanner for input
3. Prompt user to enter their age
4. Read and store age
5. If age \geq 18, print "Eligible to drive"
6. Else, print "Not eligible to drive"
7. Close the Scanner
8. End the program

Code :

```
import java.util.Scanner;
```

```
public class DrivingEligibility {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);
```

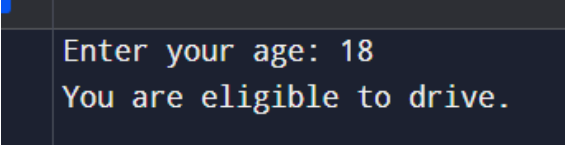
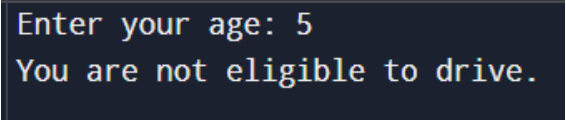
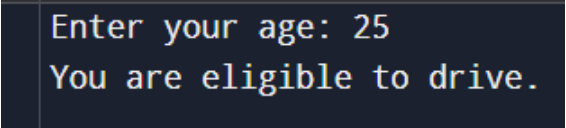
```

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

if (age >= 18) {
    System.out.println("You are eligible to drive.");
} else {
    System.out.println("You are not eligible to drive.");
}

scanner.close();
}
}

```

| Test cases | Input | Expected Output |
|------------|--------------------|---|
| TC1 | Enter the age : 18 |  |
| TC2 | Enter the age : 5 |  |
| TC3 | Enter the age : 25 |  |

Observation

The program checks if a user's age is 18 or above to determine driving eligibility. It uses a simple if-else condition to demonstrate decision-making based on numeric input.

Problem Statement :Get two double inputs and an operator (+, -, *, /) from the user

Pseudo code :

Start

create scanner to read input

ask user for first number

read num1

ask user for second number

read num2

ask user for an operator (+, -, *, /)

read operator

if operator is '+'

print num1 + num2

else if operator is '-'

print num1 - num2

else if operator is '**'

print num1 * num2

else if operator is '/'

if num2 \neq 0

print num1 / num2

else

print "error: cannot divide by zero"

else

print "invalid operator"

close scanner

End

Algorithm: steps

1. Start the program
2. Create a Scanner for input
3. Ask and read first number (num1)
4. Ask and read second number (num2)
5. Ask and read operator (+, -, *, /)
6. Use if-else to perform the corresponding operation
7. For division, check if num2 \neq 0 to avoid divide-by-zero error
8. Print the result or error message
9. Close the Scanner
10. End the program

Code :

```
import java.util.Scanner;

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

        System.out.print("Enter the first number: ");
        double num1 = scanner.nextDouble();

        System.out.print("Enter the second number: ");
        double num2 = scanner.nextDouble();

        System.out.print("Enter an operator (+, -, *, /): ");
        char operator = scanner.next().charAt(0);

        if (operator == '+') {
            System.out.println("Result: " + (num1 + num2));
        } else if (operator == '-') {
            System.out.println("Result: " + (num1 - num2));
        } else if (operator == '*') {
            System.out.println("Result: " + (num1 * num2));
        } else if (operator == '/') {
            if (num2 != 0) {
                System.out.println("Result: " + (num1 / num2));
            } else {
                System.out.println("Error: Division by zero is not allowed.");
            }
        } else {
            System.out.println("Invalid operator. Please use +, -, *, or /.");
        }

        scanner.close();
    }
}
```

```

        System.out.println("Error: Cannot divide by zero.");
    }
} else {
    System.out.println("Invalid operator.");
}

scanner.close();
}
}

```

| Test cases | Input | Expected Output |
|------------|---|--|
| TC1 | Enter the first number:2 Enter the second number : 10 Enter an operator : * | <pre> Enter the first number: 2 Enter the second number: 10 Enter an operator (+, -, *, /): * Result: 20.0 </pre> |
| TC2 | Enter the first number:10 Enter the second number : 5 Enter an operator : / | <pre> Enter the first number: 10 Enter the second number: 5 Enter an operator (+, -, *, /): / Result: 2.0 </pre> |
| TC3 | Enter the first number:5 Enter the second number : 0 Enter an operator : / | <pre> Enter the first number: 5 Enter the second number: 0 Enter an operator (+, -, *, /): / ERROR! Error: Cannot divide by zero. </pre> |

Observation

The program performs basic arithmetic operations based on user input. It uses if-else conditions to handle different operators and includes a check to prevent division by zero, demonstrating safe and interactive calculator logic.

Problem Statement : Get user age (int) and student status (boolean).

Pseudo code :

Start

create scanner to read input

ask user for age

read age

ask if user is a student (true/false)

read isstudent

if age < 5 or age > 65 then

ticketprice = 5

else if age between 5 and 18 and isstudent is true then

ticketprice = 8

else

ticketprice = 12

print ticket price

close scanner

End

Algorithm: steps

1. Start the program
2. Create a Scanner for input
3. Ask and read the user's age
4. Ask and read whether the user is a student (boolean)
5. If age is less than 5 or greater than 65, set price to \$5
6. Else if age is between 5 and 18 and student is true, set price to \$8
7. Else, set price to \$12
8. Print the ticket price
9. Close the Scanner

10. End the program

Code :

```
import java.util.Scanner;

public class MovieTicketPrice {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter your age: ");
        int age = scanner.nextInt();

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

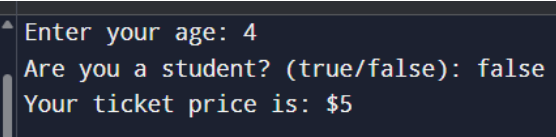
        int ticketPrice;

        if (age < 5 || age > 65) {
            ticketPrice = 5;
        } else if (age >= 5 && age <= 18 && isStudent) {
            ticketPrice = 8;
        } else {
            ticketPrice = 12;
        }

        System.out.println("Your ticket price is: $" + ticketPrice);

        scanner.close();
    }
}
```

| Test cases | Input | Expected Output |
|------------|--|---|
| TC1 | Enter your age : 5 Are you a student? : true | <pre>Enter your age: 5 Are you a student? (true/false): true Your ticket price is: \$8 === Code Execution Successful ===</pre> |
| TC2 | Enter your age : 7 Are you a student? : false | <pre>Enter your age: 7 Are you a student? (true/false): false Your ticket price is: \$12</pre> |

| | | |
|-----|--|---|
| TC3 | Enter your age : 4 Are you a student? : false |  |
|-----|--|---|

Observation

- The program calculates movie ticket prices based on age and student status using conditional logic.
- It demonstrates the use of logical operators and if-else statements to apply different pricing rules effectively.

SECTION 4 (Switch Logic - handling multiple choice)

1. Problem Statement : Ask the user to input an integer from 1-7.

Pseudo code :

Start

create scanner to read input

ask user to enter a number from 1 to 7

read day

switch (day)

case 1: print "sunday"

case 2: print "monday"

case 3: print "tuesday"

case 4: print "wednesday"

case 5: print "thursday"

case 6: print "friday"

case 7: print "saturday"

default: print "invalid input. Please enter a number from 1 to 7."

close scanner

End

Algorithm: steps

1. Start the program
2. Create a Scanner object
3. Ask the user to enter a number between 1 and 7
4. Read the number as day
5. Use a switch statement to check the value:
 - If 1 → print "Sunday"
 - If 2 → print "Monday" up to
 - If 7 → print "Saturday"
6. Otherwise, print "Invalid input"

7. Close the Scanner
8. End the program

Code :

```
import java.util.Scanner;

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

        System.out.print("Enter a number (1 to 7) to get the day of the week: ");
        int day = scanner.nextInt();

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

        scanner.close();
    }
}
```

| Test cases | Input | Expected Output |
|------------|--|--|
| TC1 | <pre>Enter a number (1 to 7) to get the day of the week: 7</pre> | <pre>Enter a number (1 to 7) to get the day of the week: 7 Saturday</pre> |
| TC2 | <pre>Enter a number (1 to 7) to get the day of the week: 0</pre> | <pre>Enter a number (1 to 7) to get the day of the week: 0 Invalid input. Please enter a number from 1 to 7.</pre> |
| TC3 | <pre>Enter a number (1 to 7) to get the day of the week: 8</pre> | <pre>Enter a number (1 to 7) to get the day of the week: 8 Invalid input. Please enter a number from 1 to 7.</pre> |

Observation

The program maps numbers 1 to 7 to days of the week using a switch statement. It demonstrates clear control flow and handles invalid input with a default case, making it a good example of menu or option-based selection logic.

2. Problem Statement : Simulate an ATM.

Get user input: 1 = Check Balance, 2 = Withdraw, 3 = Deposit, 4 =

Exit.

Use switch to print the action.

Handle invalid input with a default case.

Pseudo code :

Start

initialize balance to 1000

display menu options

ask user to choose an option (1-4)

read choice

switch (choice)

case 1: display current balance

case 2:

ask for withdrawal amount

if amount <= balance then

deduct amount from balance

display new balance

else

display "insufficient balance"

case 3:

ask for deposit amount

add amount to balance

display new balance

case 4:

display "thank you! Exiting"

default:

display "invalid option"

close scanner

End

Algorithm: steps

1. Start the program
2. Set initial balance to \$1000
3. Show user a menu with 4 options
4. Take user input for selected option
5. Use switch to handle the choice:
 - Case 1: Show balance
 - Case 2: Ask withdrawal amount → check if sufficient → update balance
 - Case 3: Ask deposit amount → update balance
 - Case 4: Exit message
 - Default: Invalid input message

6. Close the Scanner

End the program

Code :

```
import java.util.Scanner;

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

        double balance = 1000.00;

        System.out.println("1. Check Balance");
        System.out.println("2. Withdraw");
        System.out.println("3. Deposit");
        System.out.println("4. Exit");
        System.out.print("Choose an option (1-4): ");

        int choice = scanner.nextInt();

        switch (choice) {
            case 1:
                System.out.println("Your balance is: $" + balance);
                break;

            case 2:
                System.out.print("Enter amount to withdraw: ");
                double withdrawAmount = scanner.nextDouble();
                if (withdrawAmount <= balance) {
                    balance -= withdrawAmount;
                    System.out.println("Withdrawal successful. New balance: $" + balance);
                } else {
                    System.out.println("Insufficient Balance.");
                }
                break;

            case 3:
                System.out.print("Enter amount to deposit: ");
                double depositAmount = scanner.nextDouble();
                balance += depositAmount;
                System.out.println("Deposit successful. New balance: $" + balance);
                break;
```

```

    case 4:
        System.out.println("Thank you! Exiting");
        break;

    default:
        System.out.println("Invalid option. Please choose between 1 and 4.");
}

scanner.close();
}
}

```

| Test cases | Input | Expected Output |
|------------|--|--|
| TC1 | <pre> 1. Check Balance 2. Withdraw 3. Deposit 4. Exit Choose an option (1-4): 1 </pre> | <pre> 1. Check Balance 2. Withdraw 3. Deposit 4. Exit Choose an option (1-4): 1 Your balance is: \$1000.0 </pre> |
| TC2 | <pre> 1. Check Balance 2. Withdraw 3. Deposit 4. Exit Choose an option (1-4): 2 </pre> | <pre> 1. Check Balance 2. Withdraw 3. Deposit 4. Exit Choose an option (1-4): 2 Enter amount to withdraw: 500 Withdrawal successful. New balance: \$500.0 </pre> |
| TC3 | <pre> 1. Check Balance 2. Withdraw 3. Deposit 4. Exit Choose an option (1-4): 3 </pre> | <pre> 1. Check Balance 2. Withdraw 3. Deposit 4. Exit Choose an option (1-4): 3 Enter amount to deposit: 2000 Deposit successful. New balance: \$3000.0 </pre> |

Observation

- The program simulates basic ATM operations using a switch statement.
- It allows checking balance, withdrawing, and depositing money, and includes input validation, demonstrating simple banking logic with user interaction.

3. Problem Statement : Grade Remarks (Why switch is not ideal)**is not ideal)

Pseudo code :

Start

create scanner to read input

ask user to enter score (0–100)

read score

if score between 90 and 100 then

print "excellent"

else if score between 80 and 89 then

print "very good"

else if score between 70 and 79 then

print "good"

else if score between 60 and 69 then

print "pass"

else if score between 0 and 59 then

print "fail"

else

print "invalid score"

close scanner

End

Algorithm: steps

1. Start the program
2. Create a Scanner for input
3. Ask the user to enter a score
4. Read and store the score
5. Use if-else if to check score ranges:
 - 90–100 → "Excellent"
 - 80–89 → "Very Good"
 - 70–79 → "Good"
 - 60–69 → "Pass"
 - 0–59 → "Fail"
 - Otherwise → "Invalid score"
6. Close the Scanner
7. End the program

Code :

```
import java.util.Scanner;

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

        System.out.print("Enter your score (0-100): ");
        int score = scanner.nextInt();

        if (score >= 90 && score <= 100) {
            System.out.println("Excellent");
        } else if (score >= 80 && score <= 89) {
            System.out.println("Very Good");
        } else if (score >= 70 && score <= 79) {
            System.out.println("Good");
        } else if (score >= 60 && score <= 69) {
            System.out.println("Pass");
        } else if (score >= 0 && score < 60) {
            System.out.println("Fail");
        } else {
            System.out.println("Invalid score. Please enter a number between 0 and 100.");
        }

        scanner.close();
    }
}
```

| Test cases | Input | Expected Output |
|------------|-------------------------------|--|
| TC1 | Enter your score (0-100): 90 | Enter your score (0-100): 90 Excellent |
| TC2 | Enter your score (0-100): 167 | Enter your score (0-100): 167 Invalid score. Please enter a number between 0 and 100. |
| TC3 | Enter your score (0-100): 0 | Enter your score (0-100): 0 Fail |
| TC4 | Enter your score (0-100): -4 | Enter your score (0-100): -4 Invalid score. Please enter a number between 0 and 100. |

Observation

- The program evaluates a student's score and displays a remark based on defined grade ranges using if-else if statements.
- It effectively handles valid and invalid input, making it useful for understanding condition-based grading logic.

Problem Statement :Given the expression: $5+3*2>10 \ \&\&!(7==7)$

Break it down step-by-step.

Show the result after each stage of the operation and determine its final Boolean value.

Pseudo code :

Start

read values a, b, c, d, e from user

calculate $b * c$ multiplication

calculate $a +$ multiplication addition

check if addition $> d$ comparison

check if $e == e$ equality

negate equality notequality

final result = comparison and notequality

print each step and final result

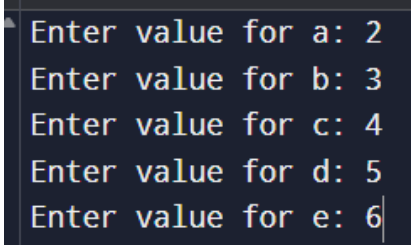
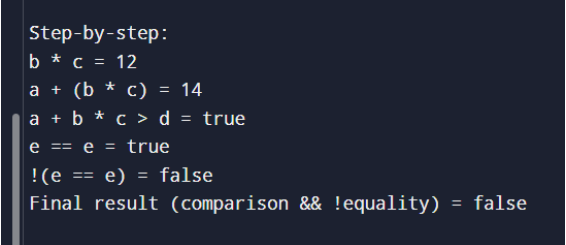
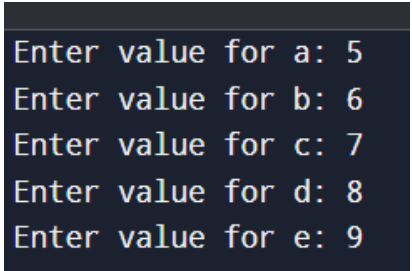
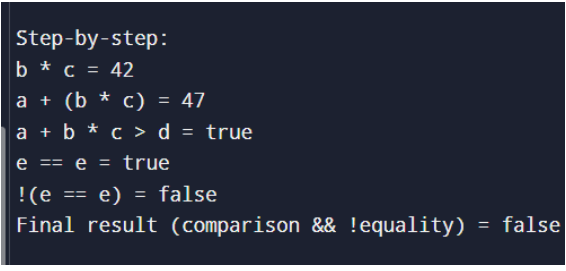
End

Algorithm: steps

1. Start program and get input for a, b, c, d, e
2. Compute $b * c$
3. Compute $a + (b * c)$
4. Check if result $> d$
5. Check if $e == e$
6. Negate that result using !
7. Combine using $\&\&$ to get final Boolean
8. Print all intermediate steps and final result
9. End program

Code :

```
public class OperatorPrecedence {  
    public static void main(String[] args) {  
  
        int a = 5;  
        int b = 3;  
        int c = 4;  
        int d = 20;  
        int e = 10;  
  
        (a + b * c) > d && !(e == e)  
        boolean result = (a + b * c) > d && !(e == e);  
        System.out.println("Final result: (a + b * c) > d && !(e == e) => " + result);  
    }  
}
```

| Test cases | Input | Expected Output |
|------------|---|---|
| TC1 |  |  |
| TC2 |  |  |

| | | |
|-----|--|--|
| TC3 | <pre>Enter value for a: 5 Enter value for b: 3 Enter value for c: 2 Enter value for d: 10 Enter value for e: 7</pre> | <pre>Step-by-step: b * c = 6 a + (b * c) = 11 a + b * c > d = true e == e = true !(e == e) = false Final result (comparison && !equality) = false</pre> |
|-----|--|--|

Observation :

The program shows how Java handles **operator precedence** by evaluating a complex expression step-by-step. It clearly illustrates the order of operations: multiplication, addition, comparison, equality, negation, and logical AND, making it useful for learning expression evaluation.