Java Programming Assignments-II

Question 1:

Write a program to check if a number is even or odd.

Program Name: EvenOddChecker.java

Answer 1:

Question 2:

Write a program to check if a number is positive, negative, or zero.

Program Name: PosNegZeroChecker.java

Answer 2:

Question 3:

Write a program to calculate the grade of a student based on marks.

Program Name: GradeCalculator.java

Answer 3:

Question 4:

Write a program to check if a year is a leap year or not.

Program Name: LeapYearChecker.java

Answer 4:

Question 5:

Write a program to check if a character is a vowel or consonant.

Program Name: VowelConsonantChecker.java

Answer 5:

Question 6:

->Write a program to display the day of the week based on user input. ->

->Program Name: -> DayOfWeek.java

Answer 6:

Question 7:

->Write a program to calculate income tax based on income slabs. ->

->Program Name: -> IncomeTaxCalculator.java

Answer 7:

Question:8

Write a Java program to read student details and marks for 6 subjects, and calculate the total marks, percentage, and result.

 -> Program Name: StuMainClass.java

 -> Concepts to Learn:
- Class and Object Creation
- Instance Variables and Methods
- Conditional Statements (if-else)
- Type Casting
- User Input Handling
- Method Calls

 -> Read:
- Student Name
- Roll Number
- Branch
- Marks for six subjects

 -> Calculations:
- Total Marks
- Percentage
- Result based on the percentage:
  - 70 to 100 => Distinction
  - 60 to <70 => FirstClass
  - 50 to <60 => SecondClass
  - 35 to <50 => ThirdClass
  - else => Fail

 -> Output:
- Student Name
- Roll Number

- Branch
- Total Marks
- Percentage
- Result

 -> Conditions:
- All subject marks must be between 0 and 100. If any mark is outside this range, display the message "Invalid marks".

 -> Java Classes:
- `StudentResult`: A class to calculate the result based on the percentage.
- `StudentPercentage`: A class to calculate the percentage from the total marks.
- `StuMainClass`: The main class to read input, validate marks, and display the student result details.

 -> Example Interaction:
```plaintext
Enter the StuName:
John Doe
Enter the RollNo:
12345
Enter the Branch:
Computer Science
===Enter 6 Sub Marks===
Enter the Sub1:
85
Enter the Sub2:
90
Enter the Sub3:
78
Enter the Sub4:
88
Enter the Sub5:
76
Enter the Sub6:
95
===StudentResultDetails===
StuName: John Doe
RollNo: 12345
Branch: Computer Science
TotMarks: 512
Percentage: 85.33
Result: Distinction
```

 -> Implement the Java program based on the above requirements.

Answer:

Question:9

Write a Java program to read two integer values and perform arithmetic operations based on the user's choice. The program should provide options for addition, subtraction, multiplication, division,

and modulus division.

 -> Program Name: DemoMethods5.java

 -> Concepts to Learn:
- Class and Object Creation
- Method Definition and Invocation
- Conditional Statements (if-else)
- User Input Handling
- Type Casting

 -> Read:
- Two integer values

 -> Arithmetic Operations:
- Addition
- Subtraction
- Multiplication
- Division
- Modulus Division

 -> Output:
- The result of the chosen arithmetic operation

 -> Java Classes:
- `Addition`: A class to perform the addition operation.
- `Subtraction`: A class to perform the subtraction operation.
- `Multiplication`: A class to perform the multiplication operation.
- `Division`: A class to perform the division operation.
- `ModDivision`: A class to perform the modulus division operation.
- `DemoMethods5`: The main class to read input, display the operation choices, and display the result based on user input.

 -> Example Interaction:
Enter the int value1:
10
Enter the int value2:
5
====Choice====
1. add
2. sub
3. mul
4. div
5. modDiv
Enter the Choice:
1
Sum: 15

 -> Implement the Java program based on the above requirements.

Answer:

Question: 10

Write a Java program to simulate a bank transaction process. The program should include the following functionalities:

1. Check PIN Number:
   - Read the user's PIN number.
   - Validate the PIN number. The valid PIN numbers are `1111`, `2222`, and `3333`.
   - If the PIN number is invalid, display an error message "Invalid pinNo" and terminate the process.

2. Transaction Options:
   - If the PIN number is valid, display the following transaction options:
     1. Withdraw
     2. Deposit

3. Withdraw:
   - Prompt the user to enter the amount to withdraw.
   - The amount must be greater than zero and a multiple of 100, otherwise display "Invalid amt...".
   - If the amount is valid, check if the amount is less than the balance. If true, perform the withdrawal, update the balance, and display the message "Amt Withdrawn:", followed by the withdrawn amount and the updated balance. Finally, display "Transaction Completed...".
   - If the amount is greater than the balance, display "Insufficient fund...".

4. Deposit:
   - Prompt the user to enter the amount to deposit.
   - The amount must be greater than zero and a multiple of 100, otherwise display "Invalid amt...".
   - If the amount is valid, perform the deposit, update the balance, and display the message "Amt deposited:", followed by the deposited amount and the updated balance. Finally, display "Transaction Completed...".

5. Balance:
   - The initial balance is set to `2000`.
   - Include a method to display the current balance.

-> Java Classes:
- `Balance`: A class to manage the balance amount and display the current balance.
- `CheckPinNo`: A class to validate the PIN number.
- `WithDraw`: A class to handle the withdrawal process.
- `Deposit`: A class to handle the deposit process.
- `BankMainClass`: The main class to run the program, handle user inputs, and manage the transaction flow.

-> Expected Output:
The program should guide the user through the process of entering a PIN, choosing a transaction type (withdraw or deposit), and performing the transaction while validating the inputs and updating the balance accordingly. If the PIN is invalid, it should terminate with an appropriate error message.

-> Example Interaction:
```plaintext
Enter the PinNo:
1111
====Choice====
1.WithDraw
2.Deposit
Enter the Choice:
```

1
Enter the amt:
500
Amt Withdrawn: 500
Balance Amt: 1500
Transaction Completed...

Enter the PinNo:
1234
Invalid pinNo...
```

Implement the Java program based on the above requirements.

**Answer:**