

ASSIGNMENT -2

1. Arithmetic & Assignment Operators

// Q1: Swap two numbers without using a third variable and arithmetic operators

```
class SwapXOR {  
    public static void main(String[] args) {  
        int a = 5, b = 10;  
        a = a ^ b;  
        b = a ^ b;  
        a = a ^ b;  
        System.out.println("After swap: a = " + a + ", b = " + b);  
    }  
}
```

// Q2: Check even or odd using bitwise operators

```
class EvenOdd {  
    public static void main(String[] args) {  
        int n = 7;  
        System.out.println((n & 1) == 0 ? "Even" : "Odd");  
    }  
}
```

// Q3: Sum of digits using modulus and division

```
class SumOfDigits {  
    public static void main(String[] args) {  
        int num = 12345, sum = 0;  
        while (num > 0) {  
            sum += num % 10;  
            num /= 10;  
        }  
        System.out.println("Sum of digits: " + sum);  
    }  
}
```

// Q4: Check divisibility by 3 without % or /

```
class DivisibilityBy3 {  
    public static boolean isDivisibleBy3(int num) {
```

```

while (num > 9) {
    int sum = 0;
    while (num > 0) {
        sum += num & 1;
        num >>= 1;
    }
    num = sum;
}
return num == 3 || num == 6 || num == 9;
}
public static void main(String[] args) {
    int n = 9;
    System.out.println(isDivisibleBy3(n) ? "Divisible by 3" : "Not divisible
by 3");
}
}

```

// Q5: Swap two numbers using += and -= operators

```

class SwapUsingOperators {
    public static void main(String[] args) {
        int a = 5, b = 10;
        a += b;
        b = a - b;
        a -= b;
        System.out.println("After swap: a = " + a + ", b = " + b);
    }
}

```

2. Relational & Logical Operators

```
import java.util.Scanner;

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

        // Q6: Find the largest of three numbers using only the ternary
        operator
        System.out.print("Enter three numbers: ");
        int a = sc.nextInt(), b = sc.nextInt(), c = sc.nextInt();
        int largest = (a > b) ? ((a > c) ? a : c) : ((b > c) ? b : c);
        System.out.println("Largest number: " + largest);

        // Q7: Check if a year is a leap year using logical operators
        System.out.print("Enter a year: ");
        int year = sc.nextInt();
        boolean isLeap = (year % 4 == 0 && year % 100 != 0) || (year % 400
        == 0);
        System.out.println(year + " is a leap year: " + isLeap);

        // Q8: Check if at least two out of three boolean inputs are true
        System.out.print("Enter three boolean values (true/false): ");
        boolean x = sc.nextBoolean(), y = sc.nextBoolean(), z =
        sc.nextBoolean();
        boolean atLeastTwoTrue = (x && y) || (y && z) || (x && z);
        System.out.println("At least two values are true: " +
        atLeastTwoTrue);

        // Q9: Check if a number is within the range (20 to 50) without using
        if-else
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        System.out.println("Number is in range 20 to 50: " + (num >= 20 &&
        num <= 50));
```

// Q10: Determine if a character is a vowel or a consonant using the ternary operator

```
System.out.print("Enter a character: ");
char ch = sc.next().toLowerCase().charAt(0);
String result = (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch ==
'u') ? "Vowel" : "Consonant";
System.out.println("Character " + ch + " is a " + result);

sc.close();
}
}
```

3. Bitwise Operators

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

```
public class BitwiseOperations {
```

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

```
        Scanner sc = new Scanner(System.in);
```

```
        // Q11: Check if a number is a power of 2 using bitwise AND
```

```
        System.out.print("Enter a number to check if it's a power of 2: ");
```

```
        int num = sc.nextInt();
```

```
        boolean isPowerOfTwo = (num > 0) && ((num & (num - 1)) == 0);
```

```
        System.out.println(num + " is a power of 2: " + isPowerOfTwo);
```

```
        // Q12: Multiply a number by 8 without using * or /
```

```
        System.out.print("Enter a number to multiply by 8: ");
```

```
        int n = sc.nextInt();
```

```
        int multipliedBy8 = n << 3; // Left shift by 3 ( $n * 2^3$ )
```

```
        System.out.println(n + " multiplied by 8 is: " + multipliedBy8);
```

```
        // Q13: Find the absolute value of an integer using bitwise operators
```

```
        System.out.print("Enter an integer to find its absolute value: ");
```

```
        int numAbs = sc.nextInt();
```

```
        int mask = numAbs >> 31; // Mask is -1 for negative numbers, 0 for  
positive
```

```
        int absoluteValue = (numAbs + mask) ^ mask;
```

```
System.out.println("Absolute value of " + numAbs + " is: " +  
absoluteValue);
```

```
// Q14: Count number of 1s (set bits) in binary representation using  
bitwise operations
```

```
System.out.print("Enter a number to count set bits: ");
```

```
int countNum = sc.nextInt();
```

```
int count = 0;
```

```
int temp = countNum;
```

```
while (temp > 0) {
```

```
    temp &= (temp - 1); // Removes the rightmost set bit
```

```
    count++;
```

```
}
```

```
System.out.println("Number of 1s in binary representation of " +  
countNum + " is: " + count);
```

```
// Q15: Swap odd and even bits of a number using bitwise operators
```

```
System.out.print("Enter a number to swap odd and even bits: ");
```

```
int swapNum = sc.nextInt();
```

```
int swappedBits = ((swapNum & 0xAAAAAAAA) >> 1) | ((swapNum &  
0x55555555) << 1);
```

```
System.out.println("Number after swapping odd and even bits: " +  
swappedBits);
```

```
sc.close();
```

```
}
```

```
}
```

4. Ternary Operator Challenges

```
import java.util.Scanner;

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

        // Q16: Determine if a number is positive, negative, or zero using
        ternary operator
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        String result = (num > 0) ? "Positive" : (num < 0) ? "Negative" :
        "Zero";
        System.out.println("The number is: " + result);

        // Q17: Find the minimum of four numbers using nested ternary
        operators
        System.out.print("Enter four numbers: ");
        int a = sc.nextInt(), b = sc.nextInt(), c = sc.nextInt(), d = sc.nextInt();
        int min = (a < b) ? ((a < c) ? ((a < d) ? a : d) : (c < d ? c : d))
            : ((b < c) ? ((b < d) ? b : d) : (c < d ? c : d));
        System.out.println("Minimum number: " + min);

        // Q18: Check if a student passed or failed using ternary operator
        System.out.print("Enter student's percentage: ");
        int percentage = sc.nextInt();
        String passOrFail = (percentage >= 40) ? "Pass" : "Fail";
        System.out.println("Result: " + passOrFail);

        // Q19: Check if a character is uppercase, lowercase, or not a letter
        using ternary operator
        System.out.print("Enter a character: ");
        char ch = sc.next().charAt(0);
        String charType = (ch >= 'A' && ch <= 'Z') ? "Uppercase"
            : (ch >= 'a' && ch <= 'z') ? "Lowercase"
            : "Not a Letter";
        System.out.println("Character type: " + charType);
    }
}
```

```
// Q20: Return the absolute value of a number using ternary
operator
System.out.print("Enter a number to find its absolute value: ");
int absNum = sc.nextInt();
int absoluteValue = (absNum < 0) ? -absNum : absNum;
System.out.println("Absolute value: " + absoluteValue);

sc.close();
}
}
```


5. Miscellaneous Operator Questions

```
import java.util.Scanner;

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

        // Q21: Increment a number without using + or ++
        System.out.print("Enter a number to increment: ");
        int num = sc.nextInt();
        int incrementedNum = ~num; // Using bitwise NOT
        System.out.println("Incremented Number: " + incrementedNum);

        // Q22: Calculator using switch-case
        System.out.print("Enter first number: ");
        int num1 = sc.nextInt();
        System.out.print("Enter an operator (+, -, *, /): ");
        char operator = sc.next().charAt(0);
        System.out.print("Enter second number: ");
        int num2 = sc.nextInt();
        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 '/': System.out.println(num2 != 0 ? "Result: " + (num1 /
num2) : "Cannot divide by zero"); break;
            default: System.out.println("Invalid operator");
        }

        // Q23: Check if a number is odd or even using bitwise &
        System.out.print("Enter a number to check odd or even: ");
        int checkNum = sc.nextInt();
        System.out.println((checkNum & 1) == 0 ? "Even" : "Odd");

        // Q24: Print all even numbers from 1 to 100 using bitwise AND
        System.out.println("Even numbers from 1 to 100:");
        for (int i = 1; i <= 100; i++) {
```

```
        if ((i & 1) == 0) // Bitwise AND to check even
            System.out.print(i + " ");
    }
    System.out.println();

    // Q25: Reverse an integer without using string conversion
    System.out.print("Enter a number to reverse: ");
    int n = sc.nextInt();
    int rev = 0;
    while (n != 0) {
        rev = rev * 10 + n % 10;
        n /= 10;
    }
    System.out.println("Reversed Number: " + rev);

    sc.close();
}
}
```