Daily Qus Practice Sheet

**Conditional-statement Base Qus.=>**

1. Find Maximum: Write a Java program to find the maximum between Three numbers using an if statement.
2. Find Minimum: Write a Java program to find the minimum between three numbers using nested if-else statements.
3. Leap Year: Write a Java program to check if a given year is a leap year or not using if-else statements.
4. Palindrome Check: Write a Java program to check if a given string is a palindrome or not using if-else statements.
5. Positive/Negative/Zero: Write a Java program to check whether a given number is positive, negative, or zero using if-else statements.
6. Even/Odd: Write a Java program to check whether a given number is even or odd using if-else statements.
7. Vowel or Consonant: Write a Java program to check whether a given character is a vowel or a consonant using if-else statements.
8. Largest among Three: Write a Java program to find the largest among three numbers using ternary operator.
9. Grading System: Write a Java program to implement a simple grading system based on the marks obtained using if-else statements.
10. Weekday or Weekend: Write a Java program to check whether a given day of the week is a weekday or a weekend day using switch-case statements.
11. Traffic Light Simulation: Write a Java program to simulate a traffic light system using switch-case statements.
12. Simple Calculator: Write a Java program to implement a simple calculator that performs addition, subtraction, multiplication, and division based on user input using switch-case statements.
13. Month Name: Write a Java program to display the name of the month corresponding to a given number using switch-case statements.
14. Factorial Calculation: Write a Java program to calculate the factorial of a given number using if-else statements.
15. Quadratic Equation Roots: Write a Java program to find the roots of a quadratic equation ax^2 + bx + c = 0 using if-else statements.

**Loops Base Qus =>**

**Fizz Buzz Problem**

**Write a program that prints the numbers from 1 to 100. But for multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print "FizzBuzz".**

**Sum of Natural Numbers**

**Write a Java program to calculate the sum of the first 100 natural numbers using a loop.**

**Check Prime Number**

**Write a method that takes an integer input and returns true if it is a prime number, and false otherwise. Use a for-loop to check divisibility.**

**Print Fibonacci Series**

**Implement a program that prints the first 20 numbers in the Fibonacci series using a loop (0, 1, 1, 2, 3, 5, ...).**

**Factorial Calculation**

**Write a method that takes an integer n and returns its factorial value using a loop.**

**Reverse a String**

**Write a Java program to reverse a given string using a loop.**

**Count Vowels and Consonants**

**Write a program that counts the number of vowels and consonants in a given string using loops.**

**Find Maximum and Minimum in an Array**

**Implement a method that takes an array of integers and finds the maximum and minimum values using a loop.**

**Multiplication Table**

**Write a method in Java that takes an integer n and prints its multiplication table up to 10 using loops.**

**Sum of Digits**

**Write a program to calculate the sum of the digits of an integer (e.g., the sum of the digits of 123 is 1 + 2 + 3 = 6).**

**Palindrome Checker**

**Implement a method that checks if a given string is a palindrome using a loop.**

**Count the Number of Each Character**

**Write a program that counts the frequency of each character in a given string using loops.**

**Find All Factors of a Number**

**Write a method that prints all factors of a given number n using a for loop.**

**Print a Pattern**

**Write a method that prints the following pattern up to n lines (for n=5):**

**markdown**

**Copy code**

**\***

**\*\***

**\*\*\***

**\*\*\*\***

**\*\*\*\*\***

**Check Armstrong Number**

**Implement a Java method that checks if a number is an Armstrong number or not. An Armstrong number is an integer such that the sum of the cubes of its digits is equal to the number itself. For example, 153 is an Armstrong number because 1³ + 5³ + 3³ = 153.**

**Array Base Qus =>**

**Find the Maximum and Minimum Element in an Array**

**Example Input: [3, 1, 7, 5, 2]**

**Example Output: Max = 7, Min = 1**

**Reverse an Array**

**Example Input: [1, 2, 3, 4, 5]**

**Example Output: [5, 4, 3, 2, 1]**

**Check if an Array is Palindrome**

**Example Input: [1, 2, 3, 2, 1]**

**Example Output: true**

**Sort an Array of 0s, 1s and 2s**

**Example Input: [0, 1, 2, 0, 1, 2, 1, 0]**

**Example Output: [0, 0, 0, 1, 1, 1, 2, 2]**

**Find the "Kth" Largest Element in an Array**

**Example Input: [3, 2, 1, 5, 6, 4], k = 2**

**Example Output: 5**

**Find All Pairs in Array Whose Sum is Equal to a Given Number**

**Example Input: [2, 6, 3, 9, 11], sum = 9**

**Example Output: [(6,3)]**

**Rotate an Array to the Right by "K" Steps**

**Example Input: [1, 2, 3, 4, 5], k = 3**

**Example Output: [3, 4, 5, 1, 2]**

**Find if There is a Subarray with Sum Zero**

**Example Input: [4, 2, -3, 1, 6]**

**Example Output: true**

**Count the Number of Inversions in an Array**

**Example Input: [2, 4, 1, 3, 5]**

**Example Output: 3**

**Merge Two Sorted Arrays**

**Example Input: [1, 3, 5], [2, 4, 6]**

**Example Output: [1, 2, 3, 4, 5, 6]**

**Find Duplicate Numbers in an Array Where Elements are up to N-1**

**Example Input: [1, 2, 3, 6, 3, 6, 1]**

**Example Output: [1, 3, 6]**

**Find Intersection of Two Arrays**

**Example Input: [1, 2, 3, 4], [2, 3, 5, 7]**

**Example Output: [2, 3]**

**Move All Zeroes to End of Array**

**Example Input: [0, 1, 0, 3, 12]**

**Example Output: [1, 3, 12, 0, 0]**

**Find the Longest Consecutive Subsequence**

**Example Input: [100, 4, 200, 1, 3, 2]**

**Example Output: 4 (sequence is 1, 2, 3, 4)**

**Spiral Order Matrix Traversal**

**Example Input: [[1, 2, 3], [4, 5, 6], [7, 8, 9]]**

**Example Output: [1, 2, 3, 6, 9, 8, 7, 4, 5]**

**Certainly! Here are the top 15 interview questions about Strings in Java, along with their answers:**

**What is a String in Java?**

**Answer: In Java, a String is an object that represents a sequence of characters. The String class is immutable, meaning once a String object is created, it cannot be changed. Strings are widely used in Java programming and are represented by the String class.**

**How do you create a String in Java?**

**Answer: Strings can be created in several ways:**

**java**

**Copy code**

**String str1 = "Hello";**

**String str2 = new String("Hello");**

**char[] charArray = {'H', 'e', 'l', 'l', 'o'};**

**String str3 = new String(charArray);**

**What is the difference between String, StringBuilder, and StringBuffer?**

**Answer:**

**String is immutable.**

**StringBuilder is mutable and is not thread-safe.**

**StringBuffer is mutable and is thread-safe (synchronized).**

**Why are Strings immutable in Java?**

**Answer: Strings are immutable in Java to ensure security, synchronization, and efficiency. Immutable objects are inherently thread-safe and can be shared freely between threads without synchronization.**

**What is the String Pool?**

**Answer: The String Pool is a special storage area in the Java heap memory where String literals are stored. When a new String is created, the JVM checks the String Pool first. If the String already exists in the pool, it returns the reference to the existing String.**

**How can you compare two Strings in Java?**

**Answer: You can compare two Strings using:**

**equals(): Compares the content of two strings.**

**==: Compares references.**

**compareTo(): Compares two strings lexicographically.**

**java**

**Copy code**

**String str1 = "Hello";**

**String str2 = "Hello";**

**boolean isEqual = str1.equals(str2); // true**

**boolean isSameReference = (str1 == str2); // true**

**int comparison = str1.compareTo(str2); // 0**

**What does the intern() method do in Java?**

**Answer: The intern() method returns a canonical representation of the string object. It ensures that all identical strings share the same memory space in the String Pool.**

**How can you convert a String to a character array in Java?**

**Answer: You can use the toCharArray() method.**

**java**

**Copy code**

**String str = "Hello";**

**char[] charArray = str.toCharArray();**

**How do you check if a String contains only digits?**

**Answer: You can use the matches() method with a regular expression.**

**java**

**Copy code**

**String str = "12345";**

**boolean isNumeric = str.matches("\\d+"); // true**

**How can you convert a String to an integer in Java?**

**Answer: You can use Integer.parseInt() or Integer.valueOf().**

**java**

**Copy code**

**String str = "123";**

**int number = Integer.parseInt(str);**

**How do you split a String in Java?**

**Answer: You can use the split() method.**

**java**

**Copy code**

**String str = "Hello,World,Java";**

**String[] parts = str.split(",");**

**What is the use of the String.format() method?**

**Answer: String.format() is used to create formatted strings.**

**java**

**Copy code**

**String formattedString = String.format("My name is %s and I am %d years old.", "John", 25);**

**How do you join multiple Strings in Java?**

**Answer: You can use the String.join() method.**

**java**

**Copy code**

**String joinedString = String.join("-", "Hello", "World", "Java");**

**What is the substring() method in Java?**

**Answer: The substring() method returns a new string that is a substring of the original string.**

**java**

**Copy code**

**String str = "HelloWorld";**

**String substr = str.substring(5); // "World"**

**String substr2 = str.substring(0, 5); // "Hello"**

**How do you remove whitespace from the beginning and end of a String?**

**Answer: You can use the trim() method.**

**java**

**Copy code**

**String str = " Hello World ";**

**String trimmedStr = str.trim(); // "Hello World"**

**These questions and answers cover a range of topics related to Strings in Java, which are commonly asked in technical interviews.**

**Ans List**

**1. Find Maximum Between Three Numbers**

**public class FindMaximum {**

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

**int a = 10, b = 20, c = 30;**

**int max = a;**

**if (b > max) {**

**max = b;**

**}**

**if (c > max) {**

**max = c;**

**}**

**System.out.println("Maximum number is: " + max);**

**}**

**}**

**2. Find Minimum Between Three Numbers Using Nested If-Else**

**java**

**Copy code**

**public class FindMinimum {**

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

**int a = 10, b = 20, c = 5;**

**int min;**

**if (a < b) {**

**if (a < c) {**

**min = a;**

**} else {**

**min = c;**

**}**

**} else {**

**if (b < c) {**

**min = b;**

**} else {**

**min = c;**

**}**

**}**

**System.out.println("Minimum number is: " + min);**

**}**

**}**

**3. Check Leap Year**

**java**

**Copy code**

**public class LeapYear {**

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

**int year = 2024;**

**if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {**

**System.out.println(year + " is a leap year.");**

**} else {**

**System.out.println(year + " is not a leap year.");**

**}**

**}**

**}**

**4. Check if a String is a Palindrome**

**java**

**Copy code**

**public class PalindromeCheck {**

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

**String str = "radar";**

**String reversedStr = new StringBuilder(str).reverse().toString();**

**if (str.equals(reversedStr)) {**

**System.out.println(str + " is a palindrome.");**

**} else {**

**System.out.println(str + " is not a palindrome.");**

**}**

**}**

**}**

**5. Check if a Number is Positive, Negative, or Zero**

**java**

**Copy code**

**public class NumberCheck {**

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

**int num = 0;**

**if (num > 0) {**

**System.out.println(num + " is positive.");**

**} else if (num < 0) {**

**System.out.println(num + " is negative.");**

**} else {**

**System.out.println(num + " is zero.");**

**}**

**}**

**}**

**6. Check if a Number is Even or Odd**

**java**

**Copy code**

**public class EvenOddCheck {**

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

**int num = 5;**

**if (num % 2 == 0) {**

**System.out.println(num + " is even.");**

**} else {**

**System.out.println(num + " is odd.");**

**}**

**}**

**}**

**7. Check if a Character is a Vowel or Consonant**

**java**

**Copy code**

**public class VowelConsonantCheck {**

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

**char ch = 'a';**

**if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' ||**

**ch == 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U') {**

**System.out.println(ch + " is a vowel.");**

**} else {**

**System.out.println(ch + " is a consonant.");**

**}**

**}**

**}**

**8. Find Largest Among Three Numbers Using Ternary Operator**

**java**

**Copy code**

**public class LargestUsingTernary {**

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

**int a = 10, b = 20, c = 30;**

**int max = (a > b) ? (a > c ? a : c) : (b > c ? b : c);**

**System.out.println("Largest number is: " + max);**

**}**

**}**

**9. Simple Grading System**

**java**

**Copy code**

**public class GradingSystem {**

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

**int marks = 85;**

**char grade;**

**if (marks >= 90) {**

**grade = 'A';**

**} else if (marks >= 80) {**

**grade = 'B';**

**} else if (marks >= 70) {**

**grade = 'C';**

**} else if (marks >= 60) {**

**grade = 'D';**

**} else {**

**grade = 'F';**

**}**

**System.out.println("Grade: " + grade);**

**}**

**}**

**10. Check if a Day is Weekday or Weekend**

**java**

**Copy code**

**public class WeekdayWeekend {**

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

**int day = 6; // 1 for Monday, 2 for Tuesday, ..., 7 for Sunday**

**switch (day) {**

**case 1:**

**case 2:**

**case 3:**

**case 4:**

**case 5:**

**System.out.println("It's a weekday.");**

**break;**

**case 6:**

**case 7:**

**System.out.println("It's a weekend.");**

**break;**

**default:**

**System.out.println("Invalid day.");**

**}**

**}**

**}**

**11. Traffic Light Simulation**

**java**

**Copy code**

**public class TrafficLight {**

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

**String lightColor = "red";**

**switch (lightColor.toLowerCase()) {**

**case "red":**

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

**break;**

**case "yellow":**

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

**break;**

**case "green":**

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

**break;**

**default:**

**System.out.println("Invalid color");**

**}**

**}**

**}**

**12. Simple Calculator**

**java**

**Copy code**

**import java.util.Scanner;**

**public class SimpleCalculator {**

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

**Scanner scanner = new Scanner(System.in);**

**System.out.println("Enter first number: ");**

**double num1 = scanner.nextDouble();**

**System.out.println("Enter second number: ");**

**double num2 = scanner.nextDouble();**

**System.out.println("Enter operation (+, -, \*, /): ");**

**char operation = scanner.next().charAt(0);**

**double result;**

**switch (operation) {**

**case '+':**

**result = num1 + num2;**

**break;**

**case '-':**

**result = num1 - num2;**

**break;**

**case '\*':**

**result = num1 \* num2;**

**break;**

**case '/':**

**if (num2 != 0) {**

**result = num1 / num2;**

**} else {**

**System.out.println("Division by zero is not allowed.");**

**return;**

**}**

**break;**

**default:**

**System.out.println("Invalid operation.");**

**return;**

**}**

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

**}**

**}**

**13. Display Month Name**

**java**

**Copy code**

**public class MonthName {**

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

**int month = 5; // 1 for January, 2 for February, ...**

**switch (month) {**

**case 1:**

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

**break;**

**case 2:**

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

**break;**

**case 3:**

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

**break;**

**case 4:**

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

**break;**

**case 5:**

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

**break;**

**case 6:**

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

**break;**

**case 7:**

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

**break;**

**case 8:**

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

**break;**

**case 9:**

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

**break;**

**case 10:**

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

**break;**

**case 11:**

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

**break;**

**case 12:**

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

**break;**

**default:**

**System.out.println("Invalid month.");**

**}**

**}**

**}**

**14. Calculate Factorial**

**java**

**Copy code**

**public class FactorialCalculation {**

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

**int number = 5;**

**int factorial = 1;**

**if (number < 0) {**

**System.out.println("Factorial of a negative number is undefined.");**

**} else {**

**for (int i = 1; i <= number; i++) {**

**factorial \*= i;**

**}**

**System.out.println("Factorial of " + number + " is " + factorial);**

**}**

**}**

**}**

**15. Find Roots of Quadratic Equation**

**java**

**Copy code**

**public class QuadraticEquationRoots {**

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

**double a = 1, b = -7, c = 10;**

**double determinant = b \* b - 4 \* a \* c;**

**if (determinant > 0) {**

**double root1 = (-b + Math.sqrt(determinant)) / (2 \* a);**

**double root2 = (-b - Math.sqrt(determinant)) / (2 \* a);**

**System.out.println("Roots are real and distinct: " + root1 + " and " + root2);**

**} else if (determinant == 0) {**

**double root = -b / (2 \* a);**

**System.out.println("Roots are real and equal: " + root);**

**} else {**

**System.out.println("Roots are complex**