Amrita Vishwa Vidyapeetham
Amrita School of Computing, Coimbatore — 2026

Java Practice Exercises: Operators, Conditionals, Loops, Arrays

I. Basic Looping & Arithmetic

1. **First 10 Natural Numbers**
   Problem: Display the first 10 natural numbers.
   Expected Output: 1 2 3 4 5 6 7 8 9 10
   Use: `for` loop.
2. **Sum of 10 Natural Numbers**
   Problem: Compute the sum of the first 10 natural numbers.
   Expected Output: The Sum is: 55
   Use: Accumulator variable.
3. **n Terms and Sum**
   Problem: Display n terms of natural numbers and their sum.
   Test Data: 7
   Expected Output: The Sum of Natural Number upto 7 terms: 28
   Use: `Scanner` for input.
4. **Sum and Average**
   Problem: Read 10 numbers and find their sum and average.
   Test Data: 2, 4, 6, 8, 10, 12, 14, 16, 18, 2
   Expected Output: Sum: 92, Average: 9.2
   Use: `for` loop and `double` type.
5. **Cube of Numbers**
   Problem: Display the cube of the number up to an integer.
   Test Data: 5
   Expected Output: Number is: 5 and cube of the 5 is: 125
   Use: `Math.pow()`.
6. **Multiplication Table**
   Problem: Display the multiplication table for a given integer.
   Test Data: 15
   Expected Output:
   15 X 1 = 15
   ...
   15 X 10 = 150
   Use: `for` loop.
7. **Vertical Multiplier Table**
   Problem: Display multiplier table vertically from 1 to n.
   Test Data: 8
   Expected Output: Tables for 1 to 8 in columns.
   Use: Nested `for` loops.
8. **Odd Numbers and Sum**
   Problem: Display n terms of odd numbers and their sum.
   Test Data: 10
   Expected Output: The odd numbers are: 1 3 5 7 9 11 13 15 17 19
   Use: `if` condition or step of 2.

II. Pattern Generation

9. **Asterisk Triangle**
   Problem: Display a right angle triangle using asterisks.
   Expected Output: * / ** / *** / ****
   Use: Nested `for` loops.
10. **Number Triangle**
    Problem: Display a right angle triangle with sequential numbers.
    Expected Output: 1 / 12 / 123 / 1234
    Use: Inner loop indexed by outer loop.
11. **Repeating Number Triangle**
    Problem: Right angle triangle with a repeated number in a row.
    Expected Output: 1 / 22 / 333 / 4444
    Use: Printing the row index.
12. **Increased Number Triangle**
    Problem: Right angle triangle with numbers increased by 1.
    Expected Output: 1 / 2 3 / 4 5 6
    Use: External counter variable.
13. **Number Pyramid**
    Problem: Make a pyramid pattern with numbers increased by 1.
    Use: Centered alignment with spaces.
14. **Asterisk Pyramid**
    Problem: Make a pyramid pattern with an asterisk.
    Use: Space padding and nested loops.
15. **Repeated Number Pyramid**
    Problem: Pyramid with a repeated number in the same row.
    Expected Output:
    1
    2 2
    3 3 3
    Use: Nested loops with space control.
16. **Odd Asterisk Pyramid**
    Problem: Pyramid where each row has an odd number of asterisks.
    Use: Formula $2i - 1$ for column limit.
17. **Floyd's Triangle**
    Problem: Print Floyd's Triangle using 0 and 1.
    Expected Output:
    1
    01
    101
    Use: `(i+j) % 2`.
18. **Diamond Pattern**
    Problem: Display a pattern like a diamond using asterisks.
    Use: Symmetrical nested loops.
19. **Pascal's Triangle**
    Problem: Display Pascal's triangle for n rows.
    Test Data: 5
    Use: Combinations or 2D array.
20. **Alphabet Pyramid**
    Problem: Display pyramid pattern using the alphabet.
    Expected Output:
    A
    ABA
    ABCBA
    Use: `char` increments.
21. **Palindrome Number Pattern**
    Problem: Pattern starting and ending with 1.
    Expected Output: 1 / 121 / 12321
    Use: Ascending then descending loops.

III. Mathematical Series

22. **Factorial Calculation**
    Problem: Calculate the factorial of a given number.
    Test Data: 5
    Expected Output: 120
    Use: `for` loop.
23. **Sum of Even Natural Numbers**
    Problem: Display the sum of n terms of even natural numbers.
    Test Data: 5
    Expected Output: 2 4 6 8 10, Sum: 30
    Use: `2 * i` logic.
24. **Cosine Series Sum**

Problem: Find sum of $[1 - X^2/2! + X^4/4! - ...]$.
Test Data: x=2, terms=5
Expected Output: -0.415873
Use: Power and factorial functions.

25. **Harmonic Series**
Problem: Display n terms of harmonic series and their sum.
Test Data: 5
Expected Output: Sum: 2.283334
Use: `1.0/i` for precision.
26. **Nines Series**
Problem: Display the sum of $[9 + 99 + 999 + ...]$.
Test Data: 5
Expected Output: 111105
Use: `t = (t * 10) + 9`.
27. **Exponential Series**
Problem: Find sum of $[1 + x + x^2/2! + x^3/3! + ...]$.
Test Data: x=2, n=5
Expected Output: 7.000000
28. **Alternate Sum Series**
Problem: Find sum of $[x - x^3 + x^5 - ...]$.
Test Data: x=2, n=5
Expected Output: 410
29. **Square Natural Numbers**
Problem: Display n terms of square numbers and their sum.
Expected Output: 1 4 9 16 25, Sum: 55
30. **Ones Series**
Problem: Find sum of $1 + 11 + 111 + ...n$ terms.
Test Data: 5
Expected Output: 12345
31. **Sum of A.P. Series**
Problem: Find the sum of an A.P. series.
Test Data: Start=1, n=10, diff=4
Expected Output: 190
32. **Sum of G.P. Series**
Problem: Find the sum of a G.P. series.
Test Data: Start=3, n=5, ratio=2
Expected Output: 93.000000

IV. Logic & Properties

33. **Perfect Number Check**
Problem: Check if a number is a 'Perfect' number.
Test Data: 56
Expected Output: Not perfect.
34. **Perfect Numbers in Range**
Problem: Find 'Perfect' numbers within a given range.
Test Data: 1 to 50
Expected Output: 6 28
35. **Armstrong Number Check**
Problem: Check if a number is an Armstrong number.
Test Data: 153
Expected Output: 153 is an Armstrong number.
36. **Armstrong Numbers in Range**
Problem: Find Armstrong numbers for a range.
Test Data: 1 to 1000
Expected Output: 1 153 370 371 407
37. **Prime Number Check**
Problem: Determine whether a given number is prime.
Test Data: 13
Expected Output: 13 is a prime number.
38. **Prime Numbers within a Range**
Problem: Find prime numbers within a range.
Test Data: 1 to 50
Expected Output: 2 3 5 7 11...47
39. **Fibonacci Series**
Problem: Display the first n terms of Fibonacci.
Test Data: 10
Expected Output: 0 1 1 2 3 5 8 13 21 34
40. **Reverse a Number**
Problem: Display a given number in reverse order.
Test Data: 12345
Expected Output: 54321
41. **Palindrome Number Check**
Problem: Check whether a number is a palindrome.
Test Data: 121
Expected Output: 121 is a palindrome.
42. **Divisible by 9**
Problem: Numbers between 100-200 divisible by 9 and their sum.
Expected Output: Sum: 1683
43. **Strong Number Check**
Problem: Check whether a number is a Strong Number.
Test Data: 15
Expected Output: 15 is not a Strong number.
44. **Strong Numbers in Range**
Problem: Find Strong Numbers within a range.
Test Data: 1 to 200
Expected Output: 1 2 145
45. **Sum of Two Primes**
Problem: Check if number is sum of two primes.
Test Data: 16
Expected Output: 16=3+13, 16=5+11
46. **n-Digit Armstrong**
Problem: Check Armstrong number of n digits.
Test Data: 1634
Expected Output: 1634 is an Armstrong number.

V. Conversions (No Arrays)

47. **Decimal to Binary**
Problem: Convert decimal to binary without using an array.
Test Data: 25
Expected Output: 11001
48. **Binary to Decimal (Strict)**
Problem: Binary to decimal without array, function or while loop.
Test Data: 1010101
Expected Output: 85
Use: `for` loop and `Math.pow()`.
49. **HCF Calculation**
Problem: Find the HCF of two numbers.
Test Data: 24, 28
Expected Output: 4
50. **LCM Using HCF**
Problem: Find LCM using the formula $(a \times b)/HCF$.
Test Data: 15, 20
Expected Output: 60
51. **Binary to Decimal (Math)**
Problem: Binary to decimal using the math function.
Test Data: 1010100
Expected Output: 84
52. **Decimal to Octal**
Problem: Convert decimal to octal without using an array.
Test Data: 79
Expected Output: 117
53. **Octal to Decimal**
Problem: Convert octal to decimal without using an array.
Test Data: 745
Expected Output: 485
54. **Binary to Octal**
Problem: Convert binary to octal.
Test Data: 1001
Expected Output: 11
55. **Octal to Binary**
Problem: Convert an octal number into binary.

Test Data: 57
Expected Output: 101111

56. **Decimal to Hexadecimal**
Problem: Convert a decimal number to hexadecimal.
Test Data: 79
Expected Output: 4F

VI. Strings & Applications

57. **Reverse a String**
Problem: Print a string in reverse order.
Test Data: Welcome
Expected Output: emocleW
Use: `charAt()` in reverse loop.
58. **String Length (Manual)**
Problem: Find length of a string without using library function.
Test Data: welcome
Expected Output: 7
Use: `toCharArray()` or similar.
59. **Count Characters (EOF)**
Problem: Count characters until end of file.
Test Data: w3resource
Expected Output: 10
Use: `hasNext()` check.
60. **Character Classifier**
Problem: Count Uppercase, Lowercase, and Other characters.
Use: `Character.isUpperCase()`.
61. **Smart Electricity Billing System**
Problem: Calculate electricity bill based on slab rates.
Rules: $10 Base; 1-100 @ $0.5; 101-300 @ $0.75; 300+ @ $1.2; 5% Late Fee.
Test Data: 250 units, No Late Payment.
Expected Output: Total Bill: $172.5
Use: Nested `if-else` logic.