

Giri's Tech Hub, Pune.

Core Java Machine Test

Batch: Nov-24 to March-25

Date : 11/07/2025

Time : 08 to 11 Am.

Instructions:

1. Solve any 9 questions & passing for 6 questions.
2. Input should be from user.
3. Indentation and comments mandatory.
4. Each program 1 marks & all comments 1 marks.

Q1. Write a java program to print this pattern.

```
A  
A B A  
A B C B A  
A B C D C B A  
A B C D E D C B A
```

Q2. Write a java program to print this pattern.

1	2	3	4	5
10	9	8	7	6
11	12	13	14	15
20	19	18	17	16
21	22	23	24	25

Q3. Write a Java program to repeatedly multiply all non-zero digits of a number until it becomes a single digit.

Sample Input :

Input: 679 Output: 6

Explanation: This question involves a while loop inside another loop, extracting digits using modulus/division, skipping zeros, and calculating product until a single-digit number is reached. It checks your grasp on loop control and digit-level processing.

Q4. Write a java program to find the union and intersection of two arrays.

Enter size of first array: 4

Enter elements of first array:

1 2 3 4

Enter size of second array: 3

Enter elements of second array:

3 4 5

Union of arrays: [1 , 2 , 3 , 4 , 5]

Intersection of arrays: [3 , 4]

Q5. Write a java program to remove duplicates value from array.

Input : 1 2 3 2 4 5 4 4 6 7

Output : 1 2 3 4 5 6 7

Q6. Given an integer array nums, return all the triplets [nums[i], nums[j], nums[k]] such that i != j, i != k, and j != k, and nums[i] + nums[j] + nums[k] == 0. Notice that the solution set must not contain duplicate triplets.

Example 1:

Input: nums = [-1,0,1,2,-1,-4]

Output: [[-1,-1,2],[-1,0,1]]

Example 2:

Input: nums = [0,1,1]

Output: []

Explanation: The only possible triplet does not sum up to 0.

Q7. Write a java program to merge two arrays.

Input - First Array :- 1 2 3 4 5

Second Array :- 6 7 8 9 10

Output - 1 10 2 9 3 8 4 7 5 6

Q8. Given a string, find the first non-repeated character using LinkedHashMap.

Input: "swiss"

Output: 'w'

Hint: Track character counts and preserve insertion order using LinkedHashMap.

Q9. Given a List<Integer> of student scores, write a Java program to:

- Sort the scores in descending order.
- Print the top 3 highest unique scores.

Sample Input: [78, 90, 90, 86, 85, 92, 70, 92]

Expected Output: 92, 90, 86

Q10. You have a list of Department objects. Each department has a list of Employee objects. An Employee has fields: id, name, salary.

Sort all employees across all departments based on the following rules:

- First by descending salary.
- If salaries are equal, sort by ascending name.
- If names are equal, sort by ascending id.

Requirement:

Use Comparator and Streams to collect the sorted employees into a single list.