**Java Backend Developer:**

**Note: If the candidate found to be copying or browsing the mobile he/she will be disqualified without any further round.**

**Java 8 Stream API (3 Minutes only)**

Integer[] nums = {-6,2,-4,3,-5,1,-6,2,-4,3,-5,1};

**Question No:1:** Implement the below requirement for the above integer array using single Stream API statement:

**Requirements:**

1. remove duplicates

2. Sort it in ascending order

3. Square all the numbers

4. find greater than 20

5. find how many greater than 20 is there

6. Print the count of how many greater than 20 is there

**Conditions:**

7. condition is you have to implement requirements 1 to 5 altogether (in the same order of requirements) using only one stream() method in the first statement.

8. second statement is for Print

9. totally 2 statements are allowed

10. only 2 minutes allowed totally for solving this

11. you can use IDE/Notepad/Notepad++

12. No need to write full code, write only the stream implementation alone

**Core Java OOPS Concept (3 Minutes only)**

**Question No 2:** Write the exact output of the below one.

**Condition:** Totally 3 minutes time for writing the output.

public class SimpleClass {

public void method(Object object) {

System.out.println("Object method " + object);

}

public void method(String str){

System.out.println("String method " + str);

}

public static void main(String[] args) {

SimpleClass instance = new SimpleClass();

instance.method("I Will");

instance.method(new String("Get Selected"));

String str = new String("In The Interview");

instance.method((Object)str);

instance.method((Object)null);

instance.method(null);

}

}

**Core Java Collection Concept (3 Minutes only)**

**Question No 3:** Write CORE JAVA program and complete the code to remove the duplicates from the List very easily and assign it back to List without using Java 8 Features.

**Condition-1:** **Without using Java 8 features**

**Condition-2:** Totally 3 minutes time for writing the output.

**Hint:** *use only Core Java Collection Concepts.*

import java.util.\*;

public class Main{

public static void main(String[] args) {

ArrayList<Integer> al = new ArrayList<>();

al.add(1);

al.add(2);

al.add(1);

al.add(2);

**Programming Logic (3 Minutes only)**

**Question No 4:**

Write a program in Core JAVA that takes in a number **N as input from User** and does the following:

if N is divisible by 3, print only "Fizz"

if N is divisible by 5, print only "Buzz"

if N is divisible by both 3 and 5, print only "FizzBuzz"

Condition-1: You should use any number of if conditions only, not else or elseif or ternary operator or exit method or return keyword.

Condition-2: Write the complete code right from import to till the end.

Condition-3: Only main method is allowed.

Condition-4: Use only println() not print()

Condition-5: totally 3 minutes only allowed.

**Test Case:** Test it with the value 15, 5 and 3

**Question No 5: (3 Minutes only)**

Write a CORE JAVA program to PRINT all the duplicate elements without using Java 8 Features.

**Condition-1:** without using Java 8 Features.

**Condition-2:** without using Java Collection classes, use Core Java only.

**Condition-3:** Total 3 minutes allowed for this.

Input: int [] arr = new int [] {1, 2, 3, 4, 2, 7, 8, 8, 3};

**Question No 6: (5 Minutes only)**

Write a Core Java program to create a Student 2D ArrayList with the following fields named as 0th index has to store studentCode, 1st index as studentName, 2nd index as studentClass, 3rd index as studentAge. Create 2D array of Student[][] and then convert that 2D array into 2D ArrayList and print it. For example Student[][] = {“S01”,”Charan”,”5th Class”,”11”};

**Note:** Datatypes of all the fields are String.

**Condition:** Total 5 minutes allowed for this.

**Question No 7: (5 Minutes only)**

Write a CORE JAVA program to get a number N from the user and then PRINT \* in the Triangle pattern exactly as below.

**Condition-1:** Total 5 minutes allowed for this.

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

**Java 8 Optional Class (3 Minutes only)**

String[] str = new String[10];

**Question No:8:** Implement the Optional class for the above source.

**Requirements:**

1. Take str[5] only

2. Check whether str[5] is null or not null

3. If it is not null, convert it into lower case and then print it

4. If it is null print appropriate message

5. Requirement 2 to 4 you have implement it using Optional methods only not Core Java.

**Conditions:**

6. Requirement 2 to 4 you have to implement it using Optional methods only not Core Java

**Question No 9: (5 Minutes only)**

Write a Database(SQL) Query to perform CRUD Operation using Database Query.

Assume that there is Employee table with the below details.

|  |
| --- |
| **Table Name** |
| Employee |

|  |  |
| --- | --- |
| **Column Name** | **Datatype** |
| Id (PK) | Int |
| EmployeeName | varchar(100) |
| ManagerName | Varchar(100) |
| ManagerId | Int |

1. Write a Insert Query to add a record to the table Employee

2. Write a Read Query to select a record from the table Employee with condition.

3. Write an Update Query to update a record to the table Employee with condition.

4. Write a Delete Query to delete a record from the table Employee with condition.

**Question No 10: (2 Minutes only)**

Write the standard response message for the below one.

|  |  |
| --- | --- |
| **Status Code** | **High Level Status Message** |
| 1XX |  |
| 2XX |  |
| 3XX |  |
| 4XX |  |
| 5XX |  |

**Question No 11: (5 Minutes only)**

Write the standard response message for the below one.

|  |  |
| --- | --- |
| **Status Code** | **Status Message** |
| 100 |  |
| 101 |  |
| 200 |  |
| 201 |  |
| 202 |  |
| 204 |  |
| 301 |  |
| 302 |  |
| 308 |  |
| 400 |  |
| 401 |  |
| 403 |  |
| 404 |  |
| 500 |  |
| 501 |  |
| 502 |  |
| 503 |  |
| 504 |  |

**Basic Details from Candidate, L1 Panel, L2 Panel:**

L1 Panel Name (in Capital) :

L1 Panel CG Mail Id (in Capital) :

L2 Panel Name (in Capital) :

Candidate Name (in Capital) :

Candidate Mail Id (in Capital):

Mobile No:

Interview Date:

Interview Place: (Chennai / Bangalore)

Total Experience:

Relevant Experience:

Hands-on Experience: Java 8 - yrs, Core Java - yrs, Spring Boot - yrs, Microservices - yrs

Resigned / Still in Service :

Official Notice Period (in days): 1 / 15 / 30 / 60 / 90 / >90

Last Working Day Date:

Hybrid Model (3 days Work from Office) Yes / No:

Native / Hometown:

Chennai Location (Yes/ No):

Bangalore Location (Yes/ No):

Government Id (PAN/Aadhaar/Driving License) Number:

L1 Photo taken (Yes/ No):

L2 Photo taken (Yes/ No):

CI / CD (Please Specify):

Certification (Please tick, if any): OCJP / AWS / GCP / Azure / Azure DevOps

Certification names in Full :

Valid From : Valid To :

Cloud Environment (Knowledge in, tick if any) : Yes / No (AWS / GCP / Azure)

Cloud Environment (Worked in, tick if any) : Yes / No (AWS / GCP / Azure)

Cloud Environment (Worked in, List down the Cloud Services) :

**Score Card**: (to be filled by L1 Panel whatever assessed only): (1 – POOR, 2 – AVERAGE, 3 – GOOD, 4 – EXCELLENT, 5 – OUTSTANDING)

Java8 - /5,

Core Java - /5,

Programming Skill - /5,

Spring Boot - /5,

Rest - /5,

Microservices - /5,

Hibernate - /5,

JPA - /5,

SQL - /5

All UAN Numbers: (By Candidate)

|  |  |
| --- | --- |
| **Name of the organization** | **UAN Number** |
|  |  |
|  |  |
|  |  |

(To be filled by Panel)

L1 (Select / Reject):

L1 Panel Name:

L2 (Select / Reject):

L2 Panel Name: