

Java Freshers

Candidate-ID	0012400060
Candidate Name	Vijaya Vardhini Dommeti
Email	vijayavardhini99@gmail.com
Mobile	9502290290

Screening Test

Completed on : 03-Dec-2024 12:43 PM

#	Question	Response	Marks
1	Have you fully completed your education and prepared to work	Yes	1
2	Have you taken any long term training course on Java Technologies	Yes	1
3	Have you recently completed your education and training, and there is no major Gap (more than one year after the education)	Yes	1
4	Are you, willing to work on-premises (there is no work from home)	Yes	1
5	Can you write simple Java Programs on your own	Yes	1
6	Are you, available immediately to join us, if we hire you	Yes	1
7	Did you complete any kind of Developer Certification in Java	Yes	1
8	Do you know how to write unit testcases using Junit	Yes	1
9	if we hire you, will you be prepared to stay and work with us at least for two years	Yes	1
10	Will you be flexible any role in Software Engineering	Yes	1
11	Explain about yourself why you are the right candidate for this Job	video response	1

Overall Score : 11/11 (Promoted)

Coding Test

Completed on : 03-Dec-2024 06:26 PM

1. Write a program or method to find the smallest and largest numbers in an array of integers. Implement two methods: `findSmallest(int[] numbers)`: Returns the smallest number in the array. `findLargest(int[] numbers)`: Returns the largest number in the array. Sample Input : If the input array is null or empty, the method should throw an `IllegalArgumentException` with the message "Array cannot be null or empty". Use Java's Stream API to solve the problem. Expected Output : Smallest number: 1 Largest number: 20

Marks Scored : 0

```
import java.util.Arrays;

public class NumberUtils{

    public static int findSmallest(int[] numbers){
        if(numbers == null||numbers.length == 0){
            throw new IllegalArgumentException("Array cannot be null or empty");
        }
        return Arrays.stream(numbers).min().getAsInt();
    }

    public static int findLargest(int[] numbers){
        if(numbers==null|| numbers.length == 0){
            throw new IllegalArgumentException("Array cannot be null or empty");
        }
        return Arrays.stream(numbers).max().getAsInt();
    }
    public static void main(String[] args){
        int[] numbers={5,3,8,1,20};

        try{
            int smallest=findSmallest(numbers);
            System.out.println("Smallest number:"+smallest);

            int largest=findLargest(numbers);
            System.out.println("Largest number:"+largest);
        }catch(IllegalArgumentException e){
            System.out.println(e.getMessage());
        }
    }
}
```

2. Question: Unique Words Extractor Write a program to extract all unique words from a given text. A word is defined as a sequence of alphanumeric characters separated by non-word characters (e.g., spaces, punctuation). The program should meet the following criteria: Extract words that are at least 8 characters long. Convert all words to lowercase for uniqueness. Preserve the order of their first appearance in the text. Implement the functionality in the `extractUniqueWords` method. The method should return a `Set` containing the filtered words. Sample Input : "The development of artificial intelligence is advancing rapidly. Innovative technologies are shaping the future of the world." Expected Output : Unique words with at least 8 characters: development artificial intelligence advancing

innovative technologies

Marks Scored : 0

```
import java.util.*;
import java.util.stream.Collectors;

public class UniqueWordsExtractor {

    public static Set extractUniqueWords(String text) {
        if(text == null|| text.isEmpty()){
            return Collections.emptySet();
        }

        Set uniqueWords = new LinkedHashSet<>();

        Pattern pattern = Pattern.compile("\\b\\w+\\b");
        Matcher matcher = pattern.matcher(text);

        while (matcher.find()){
            String word =matcher.group().toLowerCase();
            if (word.length() >= 8){
                uniqueWords.add(word);
            }
        }
        return uniqueWords;
    }

    public static void main(String[] args) {

        String text = "The development of artificial intelligence is advancing rapidly. " +
            "Innovative technologies are shaping the future of the world.";

        // Extract unique words with at least 8 characters
        Set result = extractUniqueWords(text);

        // Print the unique words
        System.out.println("Unique words with at least 8 characters:");
        for(String word : result){
            System.out.println(word);
        }
    }
}
```

3. Question: Write Java Program to use and apply the concept of Design Pattern Singleton with simplest program. Expected Output: Instance 1: Singleton@15db9742 Instance 2: Singleton@15db9742 Both instances are the same (Singleton verified). Message from Singleton: Hello from Singleton!

Marks Scored : 0

```
class Singleton {

    // Private static instance of the class
    private static Singleton instance;

    // Private constructor to prevent instantiation
    private Singleton() {}

    // Public method to provide access to the instance
    public static Singleton getInstance() {
        if (instance == null){
            instance = new Singleton();
        }
        return instance;
    }

    // Example method to demonstrate functionality
    public void showMessage() {
        System.out.println("Message from Singleton: Hello from Singleton!");
    }
}

public class SingletonExample{
    public static void main(String[] args) {

        Singleton instance1 = Singleton.getInstance();
        System.out.println("Instance 1:" + instance1);

        Singleton instance2 = Singleton.getInstance();
        System.out.println("Instance 2:" + instance2);

        if(instance1 == instance2){
            System.out.println(" both instances are the same(Singleton verified). ");
        }
        instance1.showMessage();
    }
}
```

Overall Score : 0/12 (Failed)