GSSPEC

Hirelines recruitment excellence

Java Freshers

Candidate-ID	0012400066
Candidate Name	Bharathi Kurupudi
Email	bharathikurupudi55@gmail.com
Mobile	7997669118
Source	Naukari

Screening Test

Completed on : 13-Dec-2024 08:13 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 coure on Java Technologies	Yes	1
3	Have you recently completed your education and training, and there no is major Gap (more then 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	No	0
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 candiate for this Job	video response	1

Overall Score : 10/11 (Promoted)

Coding Test

Completed on : 27-Dec-2024 08:04 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

```
public class NumberUtils {

  public static int findSmallest(int[] numbers) {
     if (numbers == null || numbers.length == 0) {
        throw new IllegalArgumentException("Array cannot be null or empty");
     }
     // Implement the logic here
     int smallest=numbers[0];
     for(int i=0;ilargest)
        {
           largest=numbers[i];
        }
     }
     return largest; // Placeholder return value
  }



  public static void main(String[] args)
  {
     int numbers[]={1,2,3,4,5,10,20};
     //NumberUtils n= new NumberUtils();
     System.out.println("Smallest number: "+ findSmallest(numbers));
     System.out.println("Largest number: " + findLargest(numbers));
  }
  }
```

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 WordProcessor {

  public static Set extractUniqueWords(String content) {
    /*Pattern pattern=Pattern.compile("\\b\\w{8,}\\b");
    Matcher matcher=pattern.matcher(content.toLowerCase());*/
    LinkedHashSet uniqueWords=new LinkedHashSet<>();
    String s[]=content.split(" ");
    for(String s1:s)
    {
      String uw=s1.toLowerCase();
      String uw1=uw.replace(".","");
      if(uw1.length()>=8)
      {
        //String uw=s1.toLowerCase();
        //String uw1=uw.replace(".","");
        uniqueWords.add(uw1);
      }
    }
    /*if()
    {
      //String word=matcher.group();
      uniqueWords.add(word);
    }*/
    return uniqueWords;



  }

  public static void main(String[] args) {
    // Example content for testing
    String content = "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 uniqueWords = extractUniqueWords(content);

    // Print the unique words
    System.out.println("Unique words with at least 8 characters:");
    uniqueWords.forEach(System.out::println);
  }
}
```

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 : 6

```
public 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)
     {
        synchronized (Singleton.class)
        {
           if(instance==null)
           {
              instance=new Singleton();
           }
        }
     }
     // Implement your code here
     return instance;
  }

  // Example method to demonstrate functionality
  public String getMessage() {
     // Implement your code here
     return "Hello from Singleton!";
  }

  // Main method to test the Singleton functionality
  public static void main(String[] args) {
     // Implement your code here
     Singleton instance1=Singleton.getInstance();
     Singleton instance2=Singleton.getInstance();
     System.out.println("Instance 1: " + instance1);
     System.out.println("Instance 2: " + instance2);
     if(instance1==instance2)
     {
        System.out.println("Both instances are the same (Singleton verified).");
     }
     System.out.println("Message from Singleton: " +instance1.getMessage());

     // Implement your code here
     //System.out.println("Are both instances the same? ");
  }
}
```

Overall Score : 6/12 (Promoted)