

Java Freshers

Candidate-ID	0012400062
Candidate Name	Anuradha Roy
Email	anuradharoy5568@gmail.com
Mobile	7439240802

Screening Test

Completed on : 10-Dec-2024 12:48 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	Skipped	0

Overall Score : 10/11 (Promoted)

Coding Test

Completed on : 11-Dec-2024 11:10 AM

1. Question: Implement an EMI Calculator in Java. The objective is to create a program that calculates the Equated Monthly Installment (EMI) for a given loan. The EMI is calculated based on the principal loan amount, annual rate of interest, and loan tenure. Sample Input : Principal amount: 500000 Annual rate of interest: 7.5 Loan tenure: 20 Expected Output : The calculated EMI is: 3982.71
Marks Scored : 0

```
public class EMI_Calculator {

    // Method to calculate EMI
    public double calculateEMI(double principal, double annualRateOfInterest, int loanTenure)
    {
        annualRateOfInterest=annualRateOfInterest/(12*100);
        if (principal < 50) {
            throw new IllegalArgumentException("Principal amount cannot be negative");
        }
        if (annualRateOfInterest < 05) {
            throw new IllegalArgumentException("Interest rate cannot be negative");
        }
        if (loanTenure <= 2) {
            throw new ArithmeticException("Loan tenure cannot be zero or negative");
        }

        // implement your code here
        return 0.0;
    }

    // Main method to test the EMI calculation
    public static void main(String[] args) {
        // Create an instance of EMI_Calculator
        EMI_Calculator calculator = new EMI_Calculator();

        // implement your code here
        calculator. double calculateEMI();

    }
}
```

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) {
        int count;
        String[] word=content.split();
        for(int i=0;i < word.length;i++) uniqueWords = extractUniqueWords(word[i]);

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

3. Question Write a program to demonstrate most simple usage, application of design pattern prototype. Expected Output: Original: Shape[type=Circle, color=Red] Cloned: Shape[type=Circle, color=Red] After modifying cloned object: Original: Shape[type=Circle, color=Red] Cloned: Shape[type=Circle, color=Blue]

Marks Scored : 0

```
// PrototypeDemo.java
class Shape implements Cloneable {
    private String type;
    private int[] dimensions;

    public Shape(String type, int[] dimensions) {
        // CODE HERE
    }

    public String getType() {
        // CODE HERE
    }

    public int[] getDimensions() {
        // CODE HERE
    }

    @Override
    protected Shape clone() {
        try {
            // CODE HERE
            return clone;
        } catch (CloneNotSupportedException e) {
            return null;
        }
    }
}
```

```
}  
}  
  
public class PrototypeDemo {  
    public static void main(String[] args) {  
        Shape original = new Shape("Circle", new int[]{7});  
        Shape cloned = original.clone();  
        cloned.getDimensions()[0] = 14;  
        System.out.println("Original Shape: " + original.getType() + " with dimensions " + java.util.Arrays.toString(original.getDimensions()));  
        System.out.println("Cloned Shape: " + cloned.getType() + " with dimensions " + java.util.Arrays.toString(cloned.getDimensions()));  
    }  
}
```

Overall Score : 0/12 (Failed)