Name - Tejaswini Pandurang Patil

Mobile – 8262952153

Email id – tejaswinii.patil2804@gmail.com

Q1  What is the difference between a PRIMARY KEY and a UNIQUE KEY?

Ans = b) We can have only one primary key in a table while we can have multiple unique keys

Q2 Which of these methods of Thread class is used to Suspend a thread for a period of time?

Ans = a) sleep()

Q3 Select the correct statement about an Exception?

Ans = c) It occurs at run time

Q4 Choose the advantages of using generics?

Ans = d) All of the mentioned

HTML/Javascript Question: (Write the syntax for following question)

1. How do you fetch the first span on the page, which has the class ‘green’?

Ans = document.querySelector(‘span.green’);

2. How to hide a paragraph embedded in a page?

Ans =document.querySelector(‘p’).style.display=’none’;

3. How to select a div with ID 'main' and class "content"?

Ans = document.querySelector(‘#main.content’);

OR

document.getElementById(‘main’);

Programming Questions: (Write program in java/.NET)    
  
1. A program that will take any string as the input and will return the position of the longest repeating   
sequence of characters within the string and also the character itself.   
For example:  if the input string is:   
asakdudadkasdsaddddkkkkwwwwmsaassjwiwwwwwwwwwaudhwdkdosooowwwwttttwweee   
Then the answer will be the character 'w' has longest repeating sequence start from 37 position.

Ans =

public class LongestRepeatingSequence {

public static void main(String[] args) {

String input = "asakdudadkasdsaddddkkkkwwwwmsaassjwiwwwwwwwwwaudhwdkdosooowwwwttttwweee";

findLongestRepeatingSequence(input);

}

public static void findLongestRepeatingSequence(String input) {

if (input == null || input.isEmpty()) {

System.out.println("Input string is empty or null.");

return;

}

int maxLength = 0;

int maxStartIndex = 0;

char maxChar = '\0';

int currentLength = 1;

int currentStartIndex = 0;

char currentChar = input.charAt(0);

for (int i = 1; i < input.length(); i++) {

if (input.charAt(i) == currentChar) {

currentLength++;

} else {

if (currentLength > maxLength) {

maxLength = currentLength;

maxStartIndex = currentStartIndex;

maxChar = currentChar;

}

// Reset for the next sequence

currentChar = input.charAt(i);

currentStartIndex = i;

currentLength = 1;

}

}

// Final check for the last sequence

if (currentLength > maxLength) {

maxLength = currentLength;

maxStartIndex = currentStartIndex;

maxChar = currentChar;

}

if (maxLength > 1) {

System.out.println("The character '" + maxChar + "' has the longest repeating sequence.");

System.out.println("Starting position: " + maxStartIndex);

} else {

System.out.println("No repeating sequence found.");

}

}

}