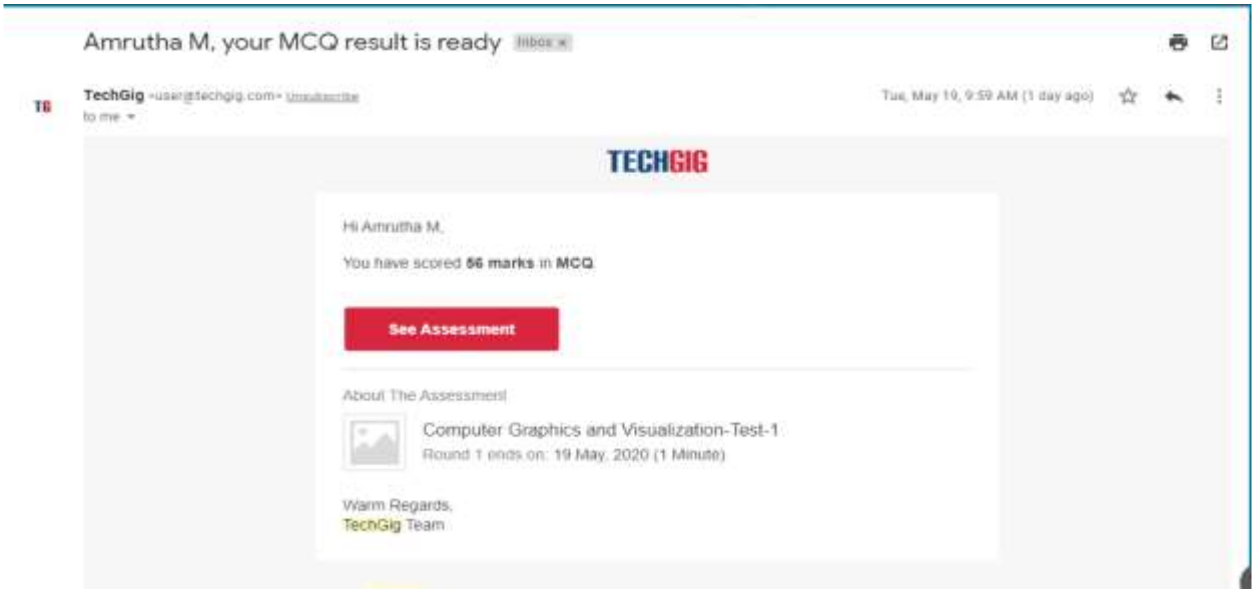
DAILY ONLINE ACTIVITIES SUMMARY

Date:	**19/05/2020**	**Name:**	**Amrutha M**
Sem & Sec	**6th sem & A sec**	**USN:**	**4AL17CS005**
Online Test Summary			
Subject	**CGV IA Test**		
Max. Marks	**60**	**Score**	**56**
Certification Course Summary			
Course	**Introduction to Full Stack Development**		
Certificate Provider	**Great Learning**	**Duration**	**1.5 hr(spent by me on that day to learn)**
Coding Challenges			
Problem Statement: **1.** Java code to find shortest palindrome for the given string. **2.** Write a simple code to identify given linked list is palindrome or not by using stack. First take a Stack. Traverse through each node of the linked list and push each node value to Stack. .			
Status: Completed			
Uploaded the report in Github	**Yes**		
If yes Repository name	https://github.com/Amrutha-M/Online-Coding		
Uploaded the report in slack	**Yes**		

Online Test Details

CGV TEST Details:



Amrutha M, your MCQ result is ready

TechGig <user@techgig.com>
to me

Tue, May 19, 9:59 AM (1 day ago)

TECHGIG

Hi Amrutha M,

You have scored **56 marks** in **MCQ**

About The Assessment

Computer Graphics and Visualization-Test-1
Round 1 ends on: 19 May, 2020 (1 Minute)

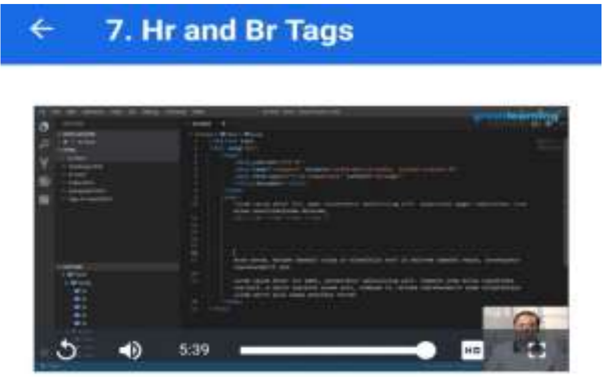
Warm Regards,
TechGig Team

Online Certification Details

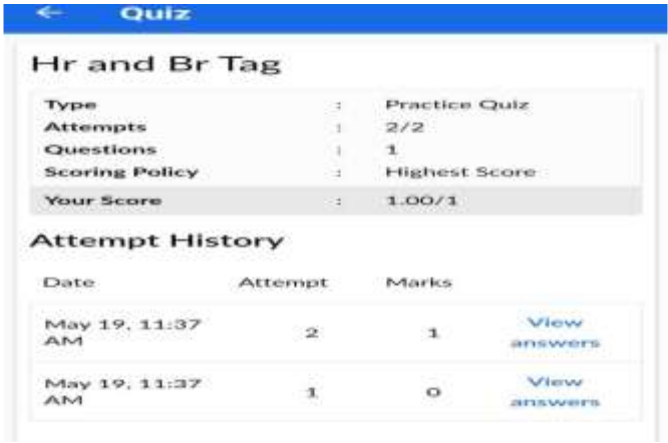
Lessons completed:

1. Heading Tag
2. Hr and Br Tags
3. Anchor Tags
4. Absolute and relative path
5. Link it Online

6. Heading tag



7. Hr and Br Tags

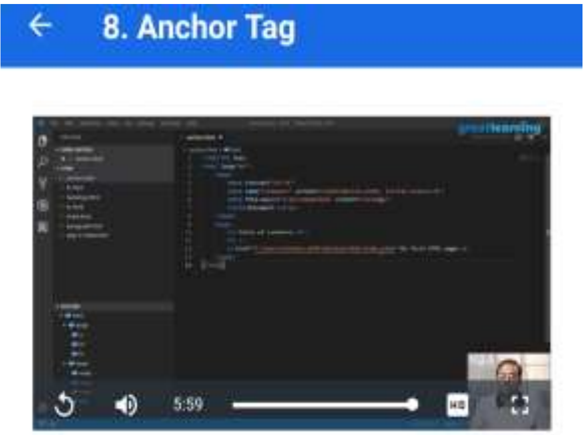


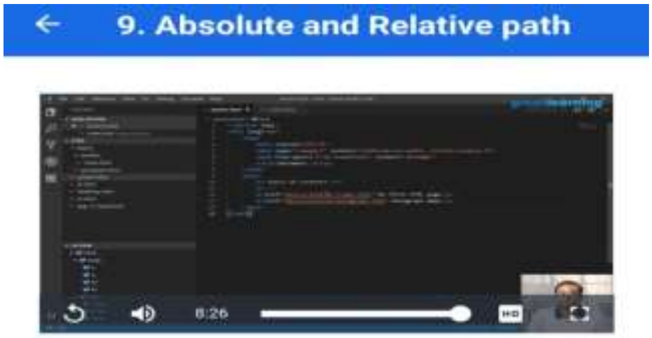
Hr and Br Tag

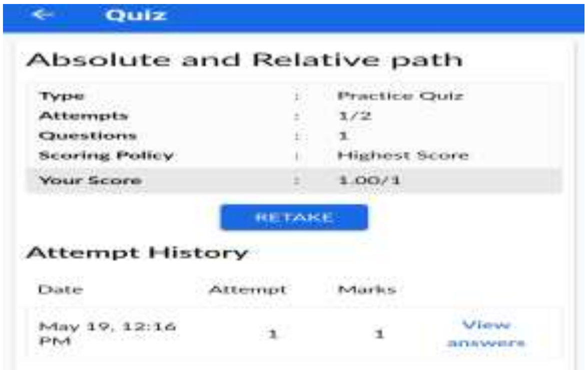
Type	:	Practice Quiz
Attempts	:	2/2
Questions	:	1
Scoring Policy	:	Highest Score
Your Score	:	1.00/1

Attempt History

Date	Attempt	Marks	
May 19, 11:37 AM	2	1	View answers
May 19, 11:37 AM	1	0	View answers





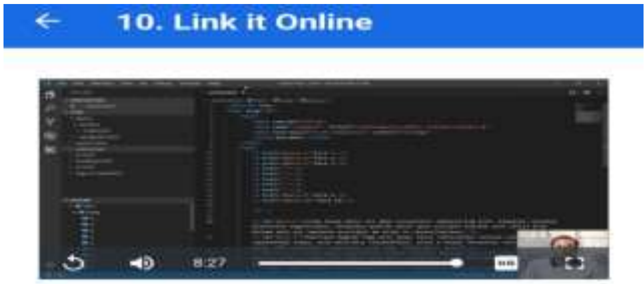


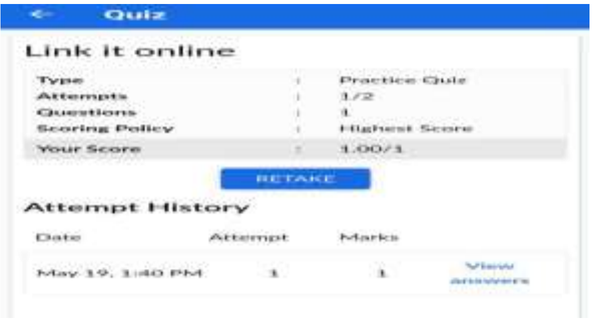
Absolute and Relative path

Type	:	Practice Quiz
Attempts	:	1/2
Questions	:	1
Scoring Policy	:	Highest Score
Your Score	:	1.00/1

Attempt History

Date	Attempt	Marks	
May 19, 12:16 PM	1	1	View answers





Link it online

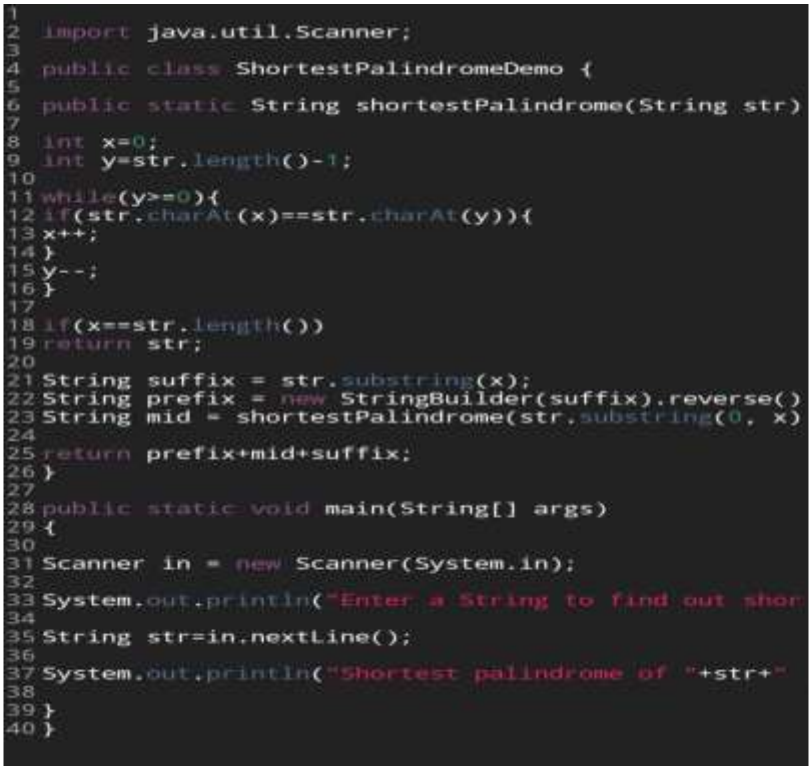
Type	:	Practice Quiz
Attempts	:	1/2
Questions	:	1
Scoring Policy	:	Highest Score
Your Score	:	1.00/1

Attempt History

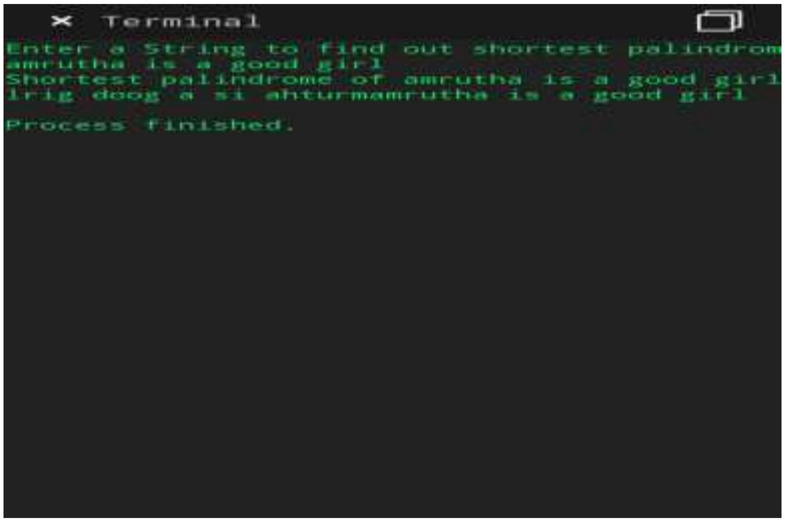
Date	Attempt	Marks	
May 19, 1:40 PM	1	1	View answers

Coding Challenge Details

1.We have a Letter or a word then we need add some letters to it and need to find out shortest palindrome
For example we take "S": S will be the shortest palindrome string.
If we take "xyz": zyxyz will be the shortest palindrome string
So we need to add some characters to the given string or character and find out what will be the shortest palindrome string by using simple java program.

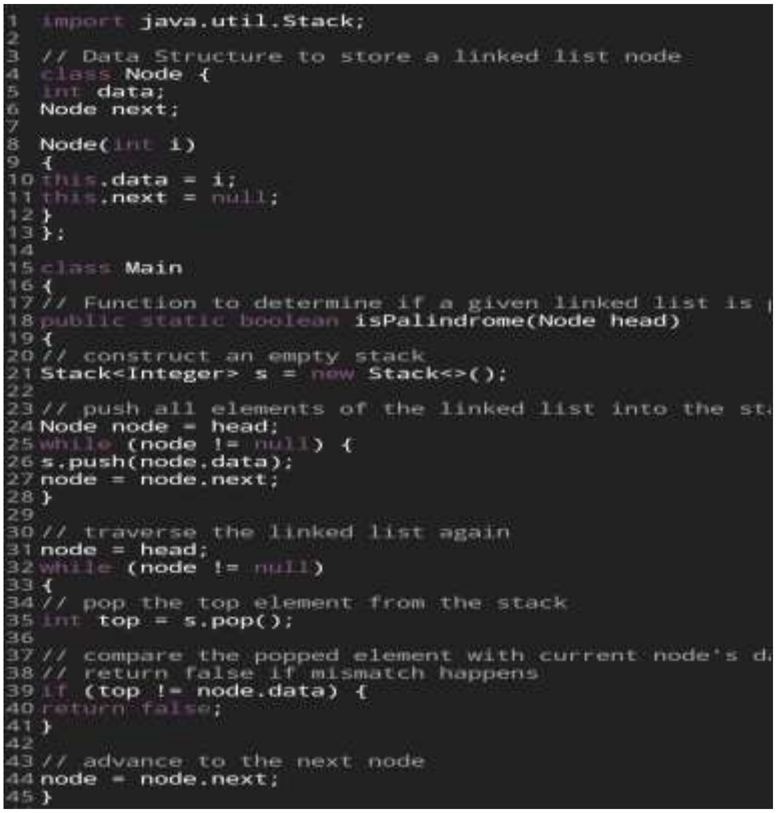


```

import java.util.Scanner;

public class ShortestPalindromeDemo {

public static String shortestPalindrome(String str)

int x=0;
int y=str.length()-1;

while(y>=0){
if(str.charAt(x)==str.charAt(y)){
x++;
}
y--;
}

if(x==str.length())
return str;

String suffix = str.substring(x);
String prefix = new StringBuilder(suffix).reverse()
String mid = shortestPalindrome(str.substring(0, x)

return prefix+mid+suffix;
}

public static void main(String[] args)
{

Scanner in = new Scanner(System.in);

System.out.println("Enter a String to find out shor

String str=in.nextLine();

System.out.println("Shortest palindrome of "+str+"

}
}
```

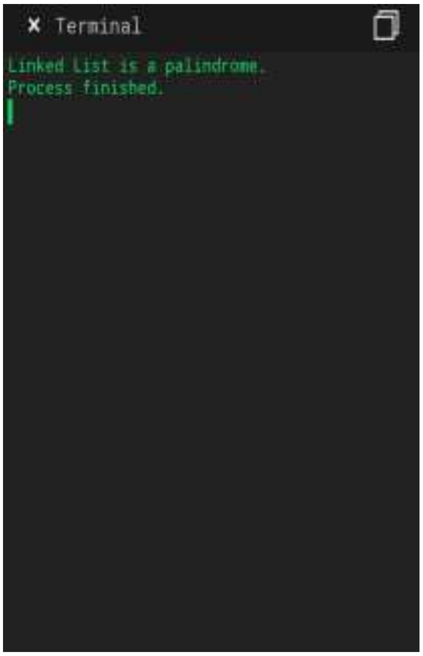


```
Enter a String to find out shortest palindrom
amrutha is a good girl
Shortest palindrome of amrutha is a good girl
lrig doog a si ahturmamrutha is a good girl

Process finished.
```

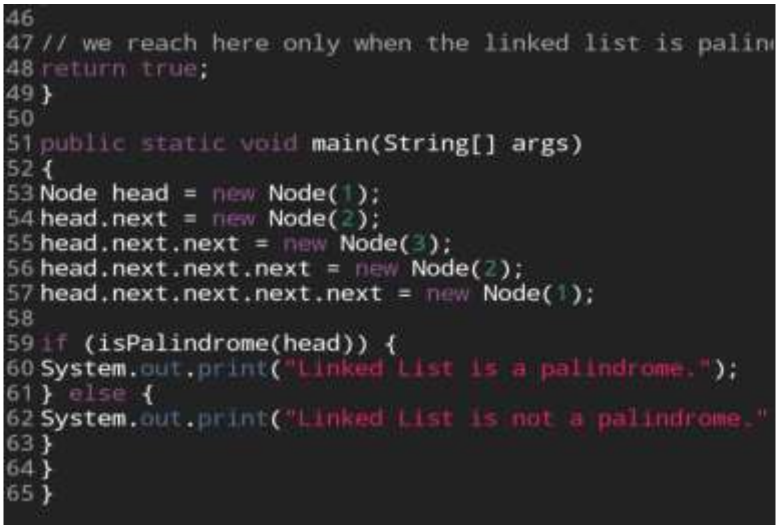
2. Write a simple code to identify given linked list is palindrome or not by using stack.
First take a Stack. Traverse through each node of the linked list and push each node value to Stack.
Once the traversal & copying is done, iterate through linked list from head node again.
In each iteration, pop one stack element and compare with node value in respective iteration. It is expected to match stack popped value with node value.
In case of all matches, its a palindrome. Any one element mismatch makes it not a palindrome.



```
import java.util.Stack;

// Data Structure to store a linked list node
class Node {
int data;
Node next;

Node(int i)
{
this.data = i;
this.next = null;
}
};

class Main
{
// Function to determine if a given linked list is
public static boolean isPalindrome(Node head)
{
// construct an empty stack
Stack<Integer> s = new Stack<>();

// push all elements of the linked list into the st
Node node = head;
while (node != null) {
s.push(node.data);
node = node.next;
}

// traverse the linked list again
node = head;
while (node != null)
{
// pop the top element from the stack
int top = s.pop();

// compare the popped element with current node's d
// return false if mismatch happens
if (top != node.data) {
return false;
}

// advance to the next node
node = node.next;
}
```



```
Linked List is a palindrome.
Process finished.
```



```

// we reach here only when the linked list is palin
return true;
}

public static void main(String[] args)
{
Node head = new Node(1);
head.next = new Node(2);
head.next.next = new Node(3);
head.next.next.next = new Node(2);
head.next.next.next.next = new Node(1);

if (isPalindrome(head)) {
System.out.print("Linked List is a palindrome.");
} else {
System.out.print("Linked List is not a palindrome."
}
}
}
```