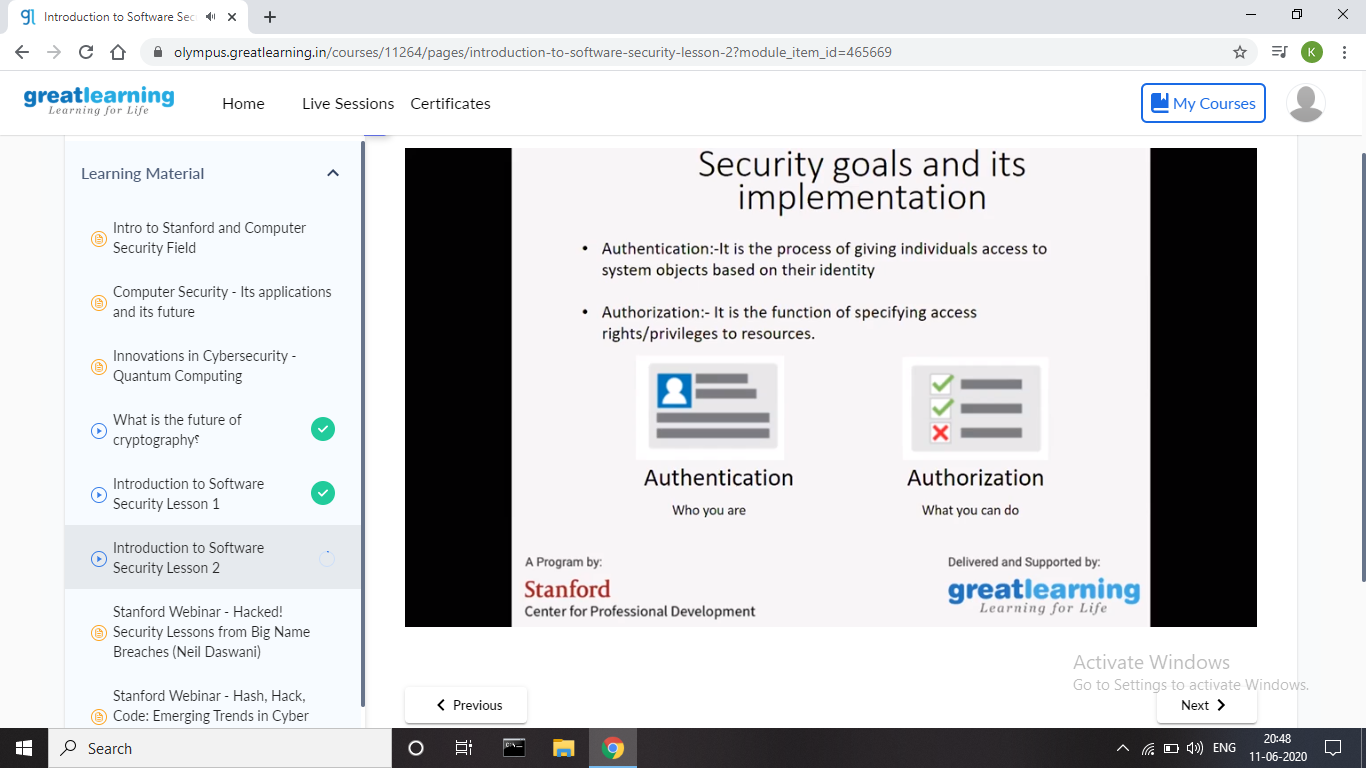
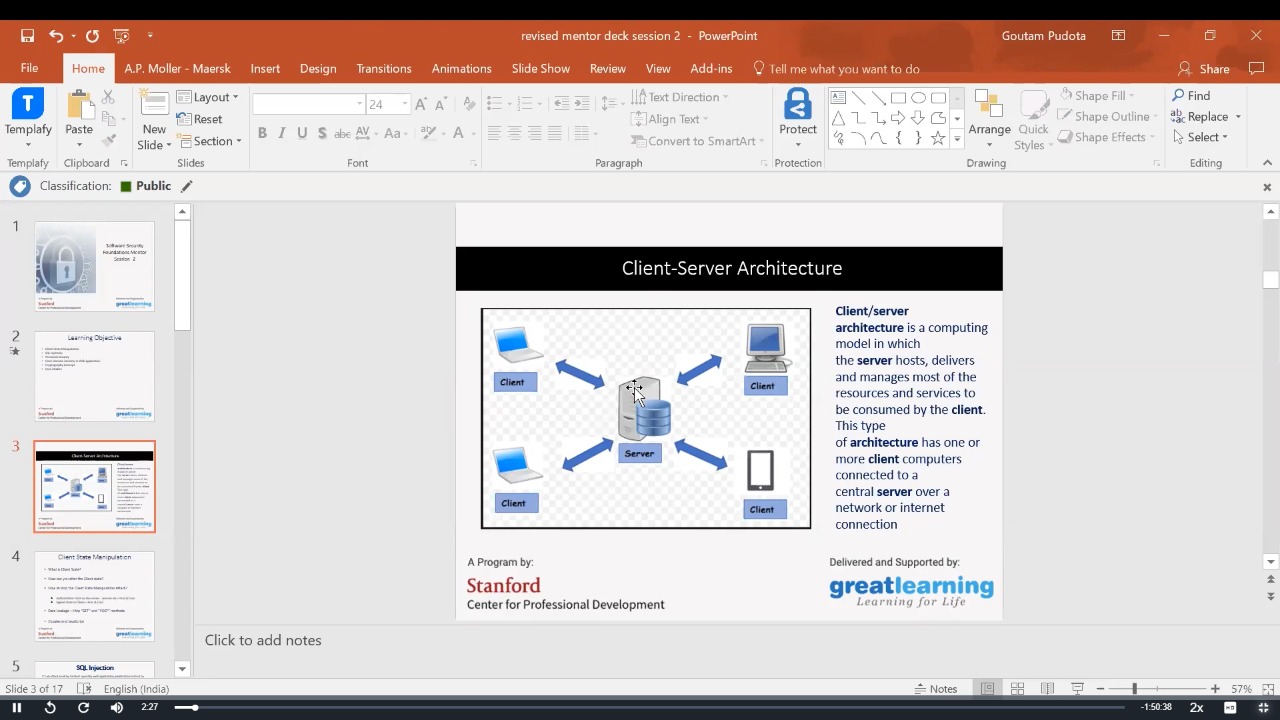
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 11/06/2020 | **Name:** | Krishnitha |
| **Sem & Sec** | 4th sem, A Section | **USN:** | 4AL18CS039 |
| **Online Test Summary** | | | |
| **Subject** | NA | | |
| **Max. Marks** | NA | **Score** | NA |
| **Certification Course Summary** | | | |
| **Course** | Introduction to Information Security | | |
| **Certificate Provider** | Great Learning Academy | **Duration:** | 3 hrs |
| **Coding Challenges** | | | |
| **Problem Statement:**  Write a Java Program to Segregate Even and Odd numbers. | | | |
| **Status:** Executed | | | |
| **Uploaded the report in GitHub** | | YES | |
| **If yes Repository name** | | <https://github.com/krishnitha/Java-coding> | |
| **Uploaded the report in slack** | | YES | |

**Certification Course Details:**

Today I have done the Course Introduction to Software Security. I learnt about cryptography and its future advances and also learnt about cryptography in quantum computing.

****

****

**Coding Challenges Details:**

**Problem:** Write a Java Program to Segregate Even and Odd numbers.

Given an array A[], write a function that segregates even and odd numbers. The functions should put all even numbers first, and then odd numbers.  
Example:

Input = {12, 34, 45, 9, 8, 90, 3}  
Output = {12, 34, 8, 90, 45, 9, 3}

Algorithm: segregateEvenOdd()

1. Initialize two index variables left and right:  
   left = 0, right = size -1
2. Keep incrementing left index until we see an odd number.
3. Keep decrementing right index until we see an even number.
4. If lef < right then swap arr[left] and arr[right]

**Solution:** Uploaded it in GitHub

