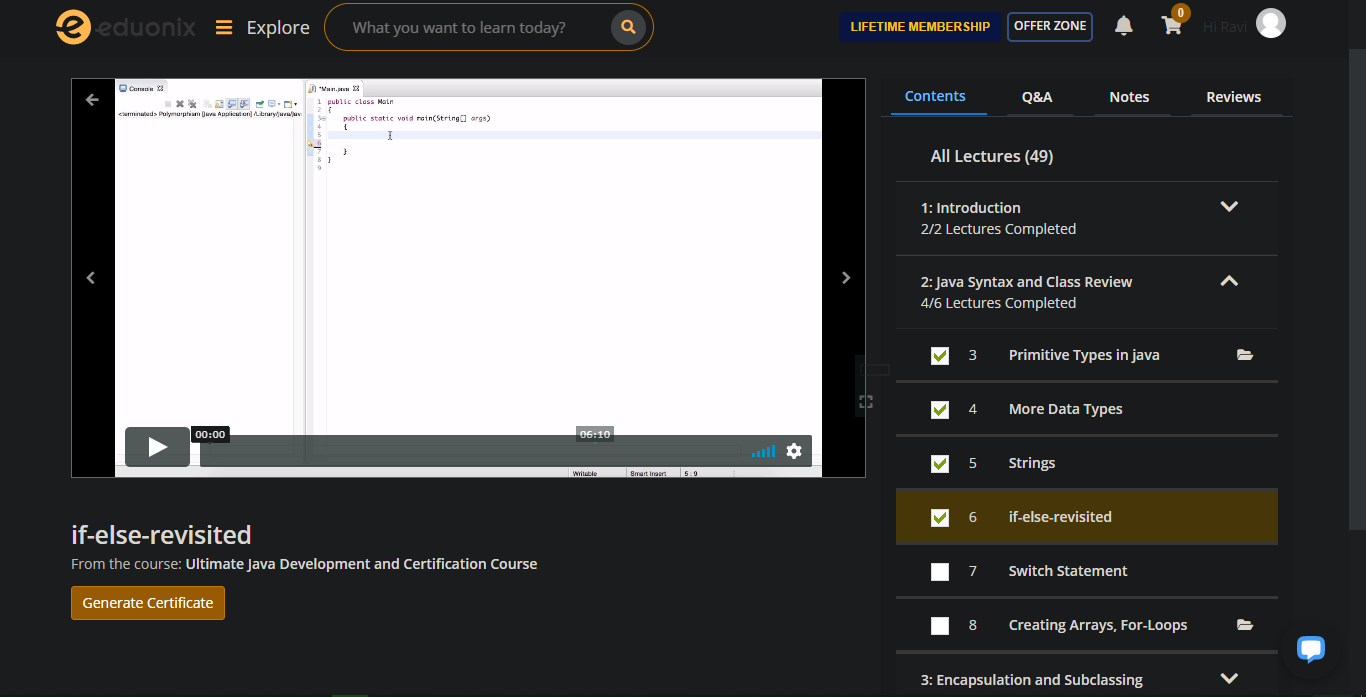
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **30/05/2020** | | | | **Name:** | **Ravi K R** | |
| **Sem & Sec** | **8th- B** | | | | **USN:** | **4AL16CS076** | |
| **Online Test Summary** | | | | | | | |
| **Subject** | | **No test shuduled** | | | | | |
| **Max. Marks** | |  | | **Score** | |  | |
| **Certification Course Summary** | | | | | | | |
| **Course** | **Ultimate java development and certification course** | | | | | | |
| **platform** | | | **Eduonix** | **Duration** | | | **20 hours** |
| **Coding Challenges** | | | | | | | |
| **Problem Statement:**  **1)** Java program to demonstrate ConcurrentSkipListSet  You may assume that each input would have **exactly** one solution, and you may not use the same element twice. | | | | | | | |
| **Status: Executed** | | | | | | | |
| **Uploaded the report in Github** | | | | **Yes** | | | |
| **If yes Repository name** | | | | <https://github.com/alvas-education-foundation/Ravi_kr> | | | |
| **Uploaded the report in slack** | | | | **Yes** | | | |

Online Test Details:

Certification:



Coding Challenges Details:

**// Java program to demonstrate ConcurrentSkipListSet**

**import java.util.concurrent.ConcurrentSkipListSet;**

**class ConcurrentSkipListSetLastExample1 {**

**public static void main(String[] args)**

**{**

**// Initializing the set using ConcurrentSkipListSet()**

**ConcurrentSkipListSet<Integer>**

**set = new ConcurrentSkipListSet<Integer>();**

**// Adding elements to this set**

**// using add() method**

**set.add(78);**

**set.add(64);**

**set.add(12);**

**set.add(45);**

**set.add(8);**

**// Printing the ConcurrentSkipListSet**

**System.out.println("ConcurrentSkipListSet: "**

**+ set);**

**// Printing the highest element of the set**

**// using last() method**

**System.out.println("The highest element of the set: "**

**+ set.last());**

**// Retrieving and removing first element of the set**

**System.out.println("The first element of the set: "**

**+ set.pollFirst());**

**// Checks if 9 is present in the set**

**// using contains() method**

**if (set.contains(9))**

**System.out.println("9 is present in the set.");**

**else**

**System.out.println("9 is not present in the set.");**

**// Printing the size of the set**

**// using size() method**

**System.out.println("Number of elements in the set = "**

**+ set.size());**

**}**

**}**