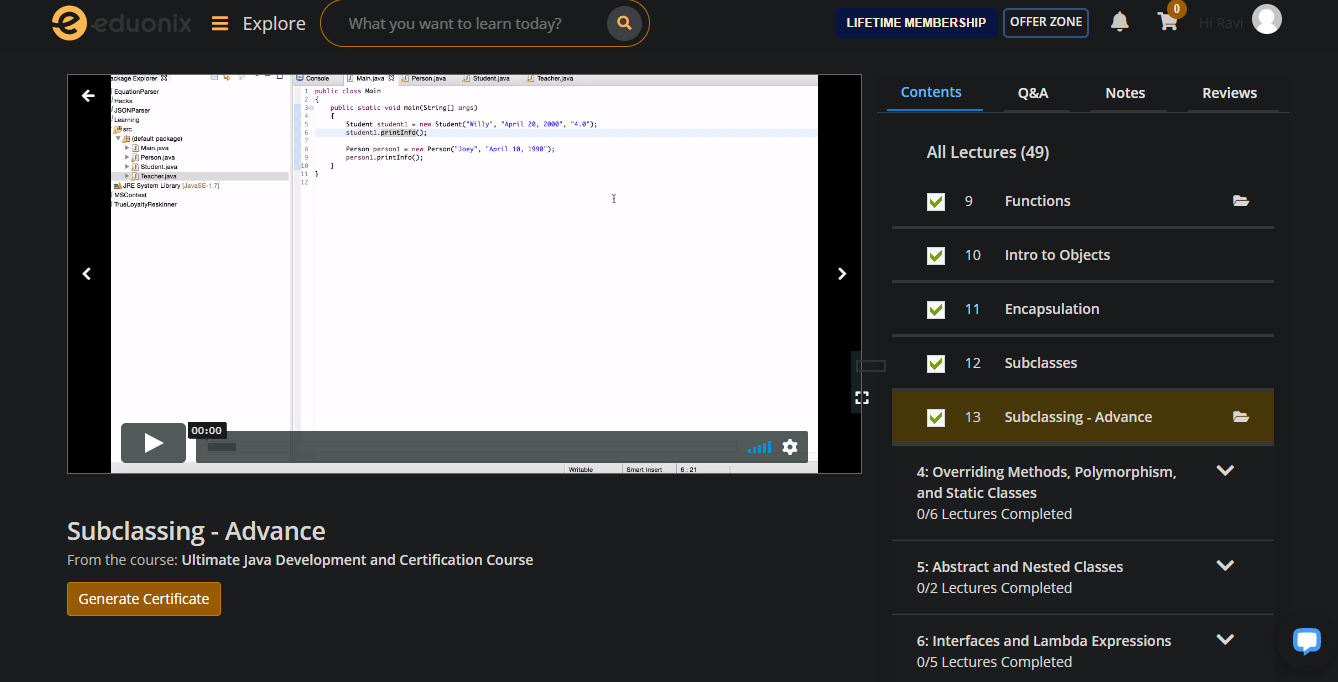
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **4/06/2020** | | | | **Name:** | **Ravi K R** | |
| **Sem & Sec** | **8th- B** | | | | **USN:** | **4AL16CS076** | |
| **Online Test Summary** | | | | | | | |
| **Subject** | | **SMS** | | | | | |
| **Max. Marks** | | **60** | | **Score** | | **60** | |
| **Certification Course Summary** | | | | | | | |
| **Course** | **Ultimate Java programming and certification** | | | | | | |
| **platform** | | | **Eduonix** | **Duration** | | | **10 hours** |
| **Coding Challenges** | | | | | | | |
| **Problem Statement:**  **1)** Java application create a list of numbers and then sort in ascending order as well as in descending order simultaneously. | | | | | | | |
| **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:

/\* Following Java application create a list of numbers and then sort in ascending order as well as in descending order simultaneously. \*/

import java.util.\*;

class Numbers {

public int result[] = new int[10];

void displayListOfNos()

{

System.out.println("Numbers stored in the array:");

for( int idx=0; idx<10; ++idx) {

System.out.println(result[idx]);

}

}

void fillTheArray(int aUpperLimit, int aArraySize)

{

if (aUpperLimit <=0) {

throw new IllegalArgumentException("UpperLimit must be positive: " + aUpperLimit);

}

if (aArraySize <=0) {

throw new IllegalArgumentException("Size of returned List must be greater than 0.");

}

Random generator = new Random();

for( int idx=0; idx<="" result[j])="" int="" temp="result[i];" result[i]="result[j];" result[j]="temp;" sortdescending()="" style="box-sizing: border-box;"> result[j]) {

int temp = result[i];

result[i] = result[j];

result[j] = temp;

}

}

}

displayListOfNos();

}

}

class ArrangementAscending implements Runnable {

Numbers n1 ;

ArrangementAscending(Numbers n) {

n1 = n;

new Thread(this).start();

}

public void run() {

n1.sortAscending();

}

}

class ArrangementDescending implements Runnable {

Numbers n2;

ArrangementDescending(Numbers n) {

n2 = n;

new Thread(this).start();

}

public void run() {

n2.sortDescending();

}

}

class ArrangingNos {

public static void main(String args[]) {

Numbers n = new Numbers();

n.fillTheArray(20,10);

ArrangementAscending a1 = new ArrangementAscending(n);

ArrangementDescending d1 = new ArrangementDescending(n);

}

}