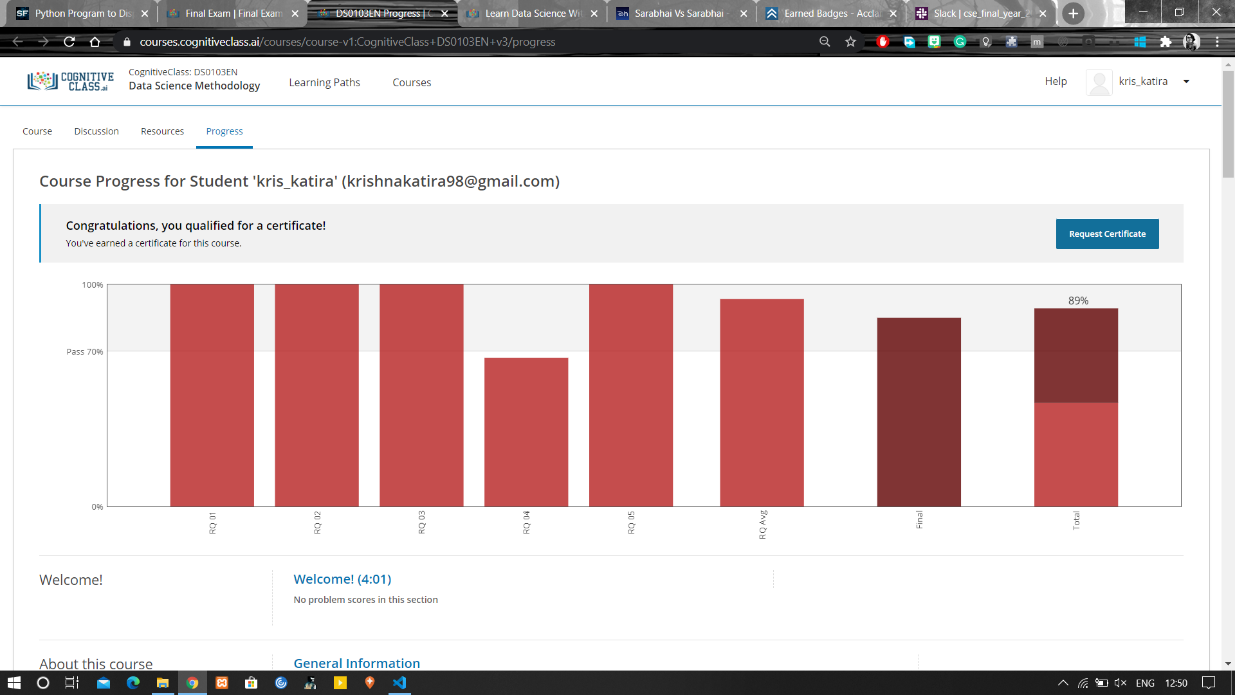
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **11/07/2020** | | | | | **Name:** | **Katira Krishna J** | |
| **Sem & Sec** | **8th A** | | | | | **USN:** | **4AL16CS045** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **-** | | | | | | |
| **Max. Marks** | | **-** | | **Score** | | | **-** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Data Science Methodology** | | | | | | | |
| **Certificate Provider** | | | **Cognitiveclass.ai** | | **Duration** | | | **5 hours** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement: Python program to implement stack using one queue** | | | | | | | | |
| **Status: Completed** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **Yes** | | | |
| **If yes Repository name** | | | | | **Krishna\_Katira** | | | |
| **Uploaded the report in slack** | | | | | **Yes** | | | |

Online Test Details:

No test conducted

Certification Course Details:



Coding Challenges Details:

**Program:**

class Stack:

def \_\_init\_\_(self):

self.q = Queue()

def is\_empty(self):

return self.q.is\_empty()

def push(self, data):

self.q.enqueue(data)

def pop(self):

for \_ in range(self.q.get\_size() - 1):

dequeued = self.q.dequeue()

self.q.enqueue(dequeued)

return self.q.dequeue()

class Queue:

def \_\_init\_\_(self):

self.items = []

self.size = 0

def is\_empty(self):

return self.items == []

def enqueue(self, data):

self.size += 1

self.items.append(data)

def dequeue(self):

self.size -= 1

return self.items.pop(0)

def get\_size(self):

return self.size

s = Stack()

print('Menu')

print('push <value>')

print('pop')

print('quit')

while True:

do = input('What would you like to do? ').split()

operation = do[0].strip().lower()

if operation == 'push':

s.push(int(do[1]))

elif operation == 'pop':

if s.is\_empty():

print('Stack is empty.')

else:

print('Popped value: ', s.pop())

elif operation == 'quit':

break