DAILY ASSESSMENT FORMAT

Date:	20 may 2020	Name:	Nishanth
Course:	TCS ION SOFT SKILL	USN:	4al17ec063
Topic:	Ace corporate interviews, learn corporate etiquette, write effective emails	Semester & Section:	6th & B
GitHub Repository:	nishanthvr		

FORENOON SESSION DETAILS
Image of session
Report – Report can be typed or hand written for up to two pages.   **Ace corporate interviews**   **In this course i learn:**   1.interview process before the interview   2.importance of interview   3.imbibbe the skill and expertise an interview requires   4.identify the engagement rules of face to face interview   5.do's and don'ts of an effective interviews   6.develop a good opening and closing interview strategy   **Learn corporate etiquette**   1.business etiquette is essential to build healthy professional relationship   2.basic rules should be followed irrespective of your position   3.courtesies should be followed at the door and elevator   4.dressing and internet etiquette should be followed as per company policy   5.importance of business etiquette   6.basic rules of business etiquette

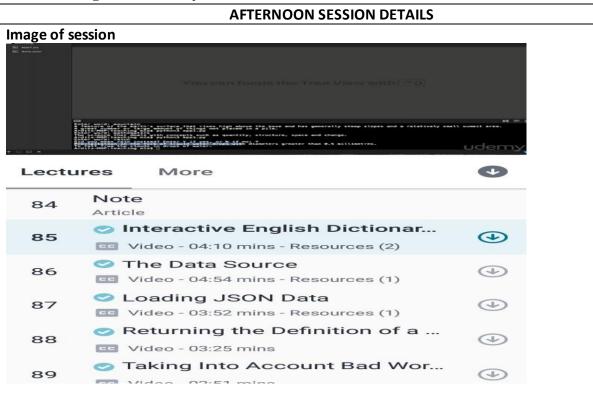|			
write effective emails:			

write effective emails:
In this course i learn**:**
1.the structure of an email
2.develop an effective subjects' line and text
3.utilize a few openings and closing phrases
4.state the do's and don'ts of email writing
5.draft an email using the pointers taught in the session

Date:	**19 may 2020**	**Name:**	**Nishanth**
Course:	**python**	**USN:**	**4al 17ec063**
Topic:	**Application, Build an Interactive English Dictionary**	**Semester & Section:**	**6th and b section**

AFTERNOON SESSION DETAILS

Image of session

Report – Report can be typed or hand written for up to two pages.
In this class I learn about application of for and while loop and how to build interactive English dictionary
Code 1

```
1. import json
2. from difflib import get_close_matches
3. data = json.load(open("data.json"))
4. def translate(w):
5.     w = w.lower()
6.     if w in data:
7.         return data[w]
8.     elif w.title() in data: #if user entered "texas" this will check for
   "Texas" as well.
```

```
9.          return data[w.title()]
10.     elif len(get_close_matches(w, data.keys())) > 0:
11.         yn = input("Did you mean %s instead? Enter Y if yes, or N if no:
   " % get_close_matches(w, data.keys())[0])
12.         if yn == "Y":
13.             return data[get_close_matches(w, data.keys())[0]]
14.         elif yn == "N":
15.             return "The word doesn't exist. Please double check it."
16.         else:
17.             return "We didn't understand your entry."
18.     else:
19.         return "The word doesn't exist. Please double check it."
20.
21.  word = input("Enter word: ")
22.  output = translate(word)
23.  if type(output) == list:
24.      for item in output:
25.          print(item)
26.  else:
27.      print(output)
```

code 2

```
1.  import json
2.  from difflib import get_close_matches
3.  data = json.load(open("data.json"))
4.  def translate(w):
5.      w = w.lower()
6.      if w in data:
7.          return data[w]
8.      elif w.title() in data:
9.          return data[w.title()]
10.     elif w.upper() in data: #in case user enters words like USA or NATO
11.         return data[w.upper()]
12.     elif len(get_close_matches(w, data.keys())) > 0:
13.         yn = input("Did you mean %s instead? Enter Y if yes, or N if no:
   " % get_close_matches(w, data.keys())[0])
14.         if yn == "Y":
15.             return data[get_close_matches(w, data.keys())[0]]
16.         elif yn == "N":
17.             return "The word doesn't exist. Please double check it."
18.         else:
19.             return "We didn't understand your entry."
20.     else:
21.         return "The word doesn't exist. Please double check it."
22.  word = input("Enter word: ")
23.  output = translate(word)
24.  if type(output) == list:
25.      for item in output:
26.          print(item)
27.  else:
28.      print(output)
```