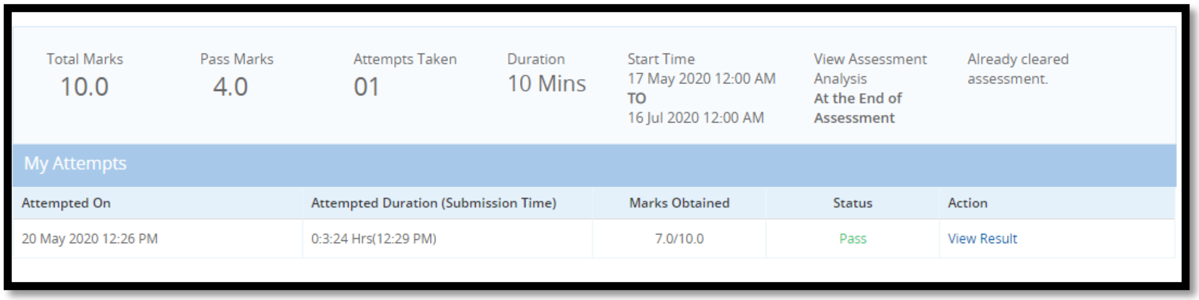
DAILY ASSESSMENT FORMAT

Date:	**20/05/2020**	**Name:**	**Davis S. Patel**
Course:	**TCS ION**	**USN:**	**4AL16EC045**
Topic:	**Ace Corporate Interviews Learn Corporate Etiquette Write Effective Emails**	**Semester & Section:**	**8th - A**
GitHub Repository:	**Davis**		

FORENOON SESSION DETAILS

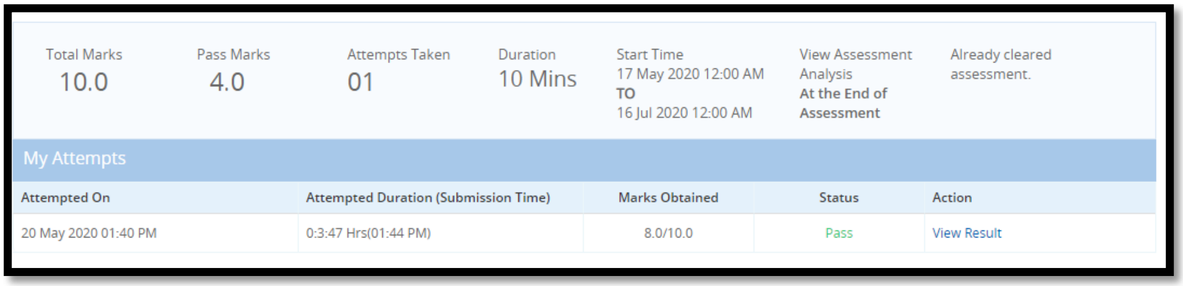
Image of session



Total Marks	Pass Marks	Attempts Taken	Duration	Start Time	View Assessment Analysis	Already cleared assessment.
10.0	4.0	01	10 Mins	17 May 2020 12:00 AM TO 16 Jul 2020 12:00 AM	At the End of Assessment	

My Attempts

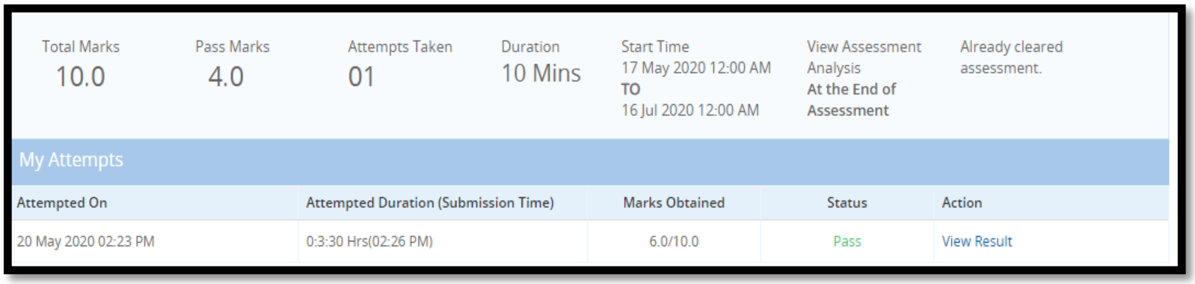
Attempted On	Attempted Duration (Submission Time)	Marks Obtained	Status	Action
20 May 2020 12:26 PM	0:3:24 Hrs(12:29 PM)	7.0/10.0	Pass	View Result



Total Marks	Pass Marks	Attempts Taken	Duration	Start Time	View Assessment Analysis	Already cleared assessment.
10.0	4.0	01	10 Mins	17 May 2020 12:00 AM TO 16 Jul 2020 12:00 AM	At the End of Assessment	

My Attempts

Attempted On	Attempted Duration (Submission Time)	Marks Obtained	Status	Action
20 May 2020 01:40 PM	0:3:47 Hrs(01:44 PM)	8.0/10.0	Pass	View Result



Total Marks	Pass Marks	Attempts Taken	Duration	Start Time	View Assessment Analysis	Already cleared assessment.
10.0	4.0	01	10 Mins	17 May 2020 12:00 AM TO 16 Jul 2020 12:00 AM	At the End of Assessment	

My Attempts

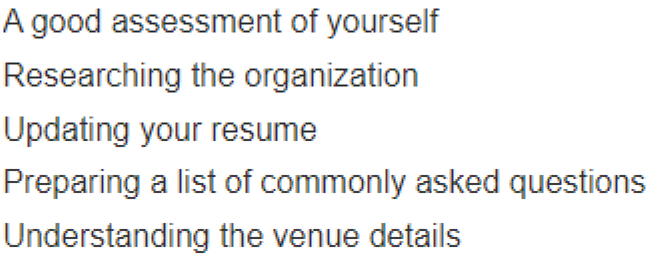
Attempted On	Attempted Duration (Submission Time)	Marks Obtained	Status	Action
20 May 2020 02:23 PM	0:3:30 Hrs(02:26 PM)	6.0/10.0	Pass	View Result

Report –

An interview is essentially a structured conversation where one participant asks questions, and the other provides answers. In common parlance, the word "interview" refers to a one-on-one conversation between an *interviewer* and an interviewee. The interviewer asks questions to which the interviewee responds, usually so information is offered by the interviewee to interviewer -- and that information may be used or provided to other audiences, whether in real time or later.

This feature is common to many types of interviews - a job interview or interview with a witness to an event may have no other audience present at the time, but the answers will be later provided to others in the employment or investigative process.

Preparation for Job interview includes –



A good assessment of yourself

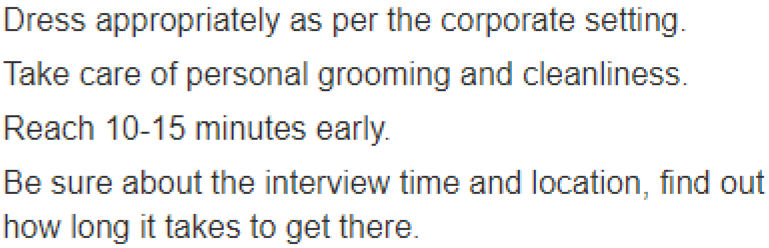
Researching the organization

Updating your resume

Preparing a list of commonly asked questions

Understanding the venue details

Dos before an Interview –



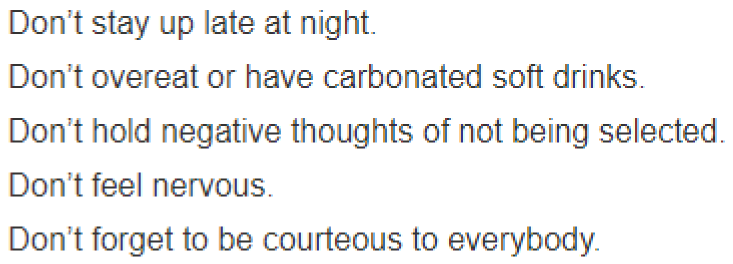
Dress appropriately as per the corporate setting.

Take care of personal grooming and cleanliness.

Reach 10-15 minutes early.

Be sure about the interview time and location, find out
how long it takes to get there.

Don’ts before an Interview –



Don’t stay up late at night.

Don’t overeat or have carbonated soft drinks.

Don’t hold negative thoughts of not being selected.

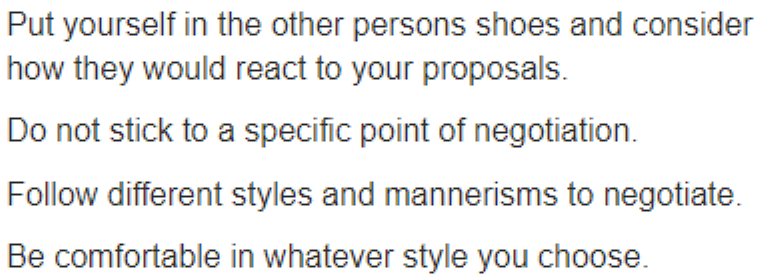
Don’t feel nervous.

Don’t forget to be courteous to everybody.

Negotiation Skills

Negotiation skills are qualities that allow two or more parties to reach a compromise. These are often soft skills and include abilities such as communication, persuasion, planning, strategizing and cooperating. Understanding these skills is the first step to becoming a stronger negotiator.

Techniques required to be a good negotiator –



Put yourself in the other persons shoes and consider how they would react to your proposals.

Do not stick to a specific point of negotiation.

Follow different styles and mannerisms to negotiate.

Be comfortable in whatever style you choose.

Learn Corporate Etiquette

This online course helped me to Learn Corporate Etiquette that teaches how a smile, being punctual, maintaining a neat cubicle and good relationships with co-workers will help us in progressing to higher levels in your profession. This module will helped in applying corporate etiquette in our day-to-day work environment.

The basis of business etiquette is about building strong relationships in your field by fostering better communication. This can only happen when those you work with feel secure and comfortable. Although basic business etiquette may vary from country to country, some principles stand the test of time and geography.

In the business world, it is best to observe the old rule, “Five minutes early is late.” Allow yourself enough time to arrive promptly, take off your coat, and settle in a bit. Arriving at a meeting exactly at the appointed time can make you feel rushed, and you will look it. Time is a commodity; by being punctual, you show you respect others.

Email is a valuable tool, it creates some challenges for writers. Miscommunication can easily occur when people have different expectations about the messages that they send and receive. Email is used for many different purposes, including contacting friends, communicating with professors and supervisors, requesting information, and applying for jobs, internships, and scholarships. Depending on your purposes, the messages you send will differ in their formality, intended audience, and desired outcomes.

Effective Emails

This online course, write an Effective Email, helped me to understand how writing an effective email is very important in making the right impact in the professional world. I learned how important it is to write an appropriate subject line and follow a proper email structure while writing an email.

Email is a valuable tool, it creates some challenges for writers. Miscommunication can easily occur when people have different expectations about the messages that they send and receive. Email is used for many different purposes, including contacting friends, communicating with professors and supervisors, requesting information, and applying for jobs, internships, and scholarships. Depending on your purposes, the messages you send will differ in their formality, intended audience, and desired outcomes.

Do’s of email Etiquette

Use a strong subject line.

Type the correct email Id.

Keep your email short.

Write about only one topic in one email.

Be polite and courteous in tone.

Use correct spelling and grammar.

Proofread all your emails before sending them.

Reply within a reasonable time.

Keep the message formal as your message could be read by many.

Don’ts of email Etiquette

Don't use all UPPER case or all lower case.

Don't use unnecessary humor.

Don't use too many short forms, SMS language.

Don't use one word responses.

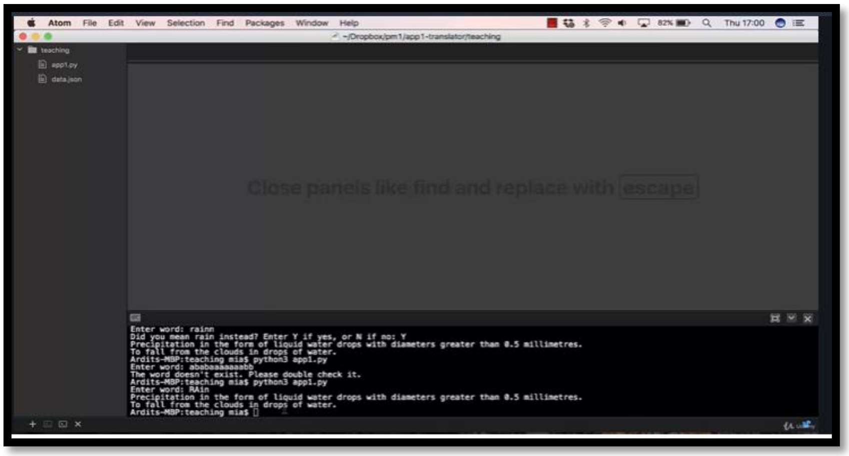
Don't call as soon as you send the message.

DAILY ASSESSMENT FORMAT

Date:	**20/05/2020**	**Name:**	**Davis S. Patel**
Course:	**Python Course**	**USN:**	**4AL16EC045**
Topic:	**Application 1: Build an Interactive English Dictionary**	**Semester & Section:**	**8th - A**
GitHub Repository:	**Davis**		

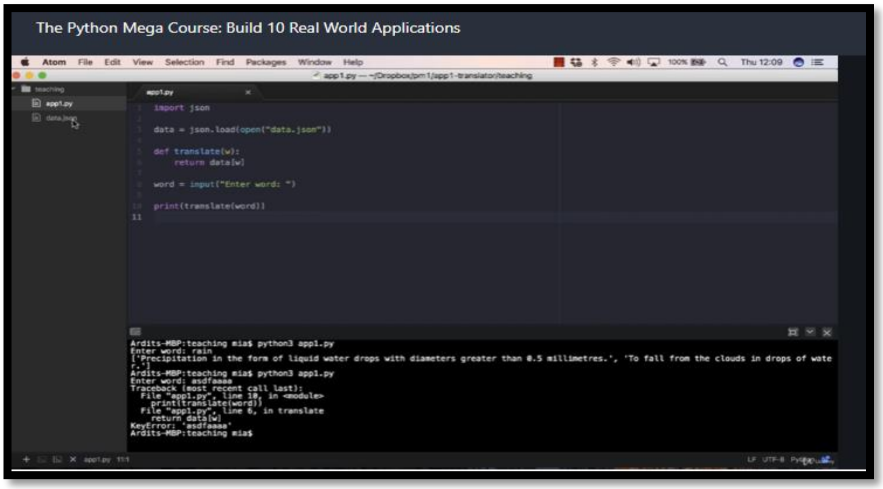
AFTERNOON SESSION DETAILS

Image of session



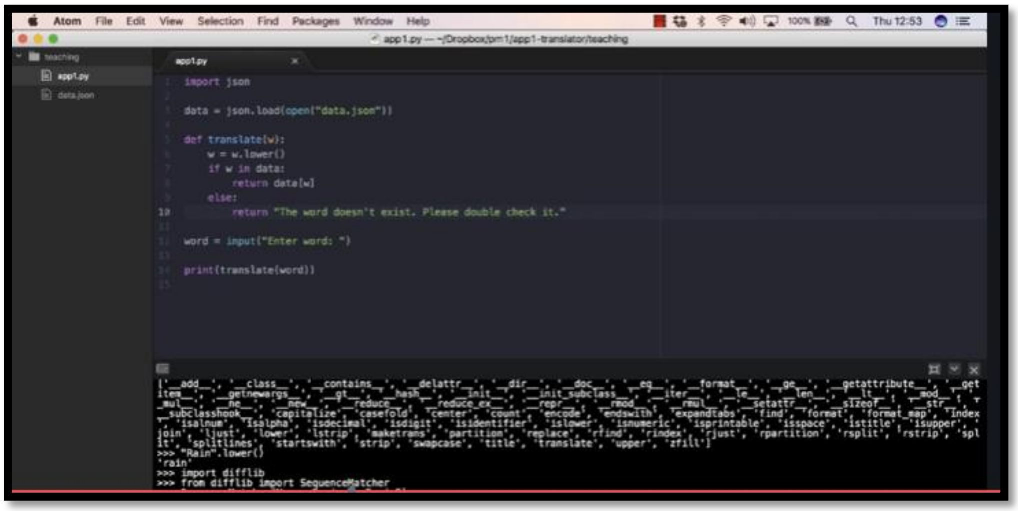
Close panels like find and replace with escape

```
Enter word: rainn
Did you mean rain instead? Enter Y if yes, or N if no: Y
Precipitation in the form of liquid water drops with diameters greater than 0.5 millimetres.
To fall from the clouds in drops of water.
Ardits-MBP:teaching mia$ python3 app1.py
Enter word: abababaaaaabb
The word doesn't exist. Please double check it.
Ardits-MBP:teaching mia$ python3 app1.py
Enter word: RAin
Precipitation in the form of liquid water drops with diameters greater than 0.5 millimetres.
To fall from the clouds in drops of water.
Ardits-MBP:teaching mia$ 
```


The Python Mega Course: Build 10 Real World Applications

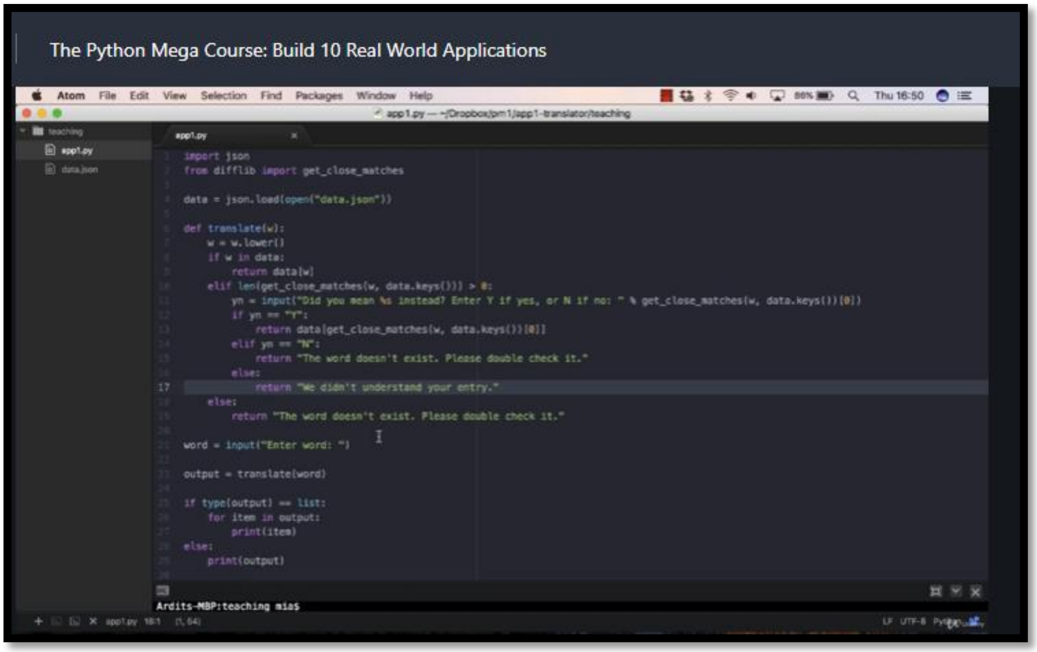
```python
import json

data = json.load(open("data.json"))

def translate(w):
    return data[w]

word = input("Enter word: ")

print(translate(word))
```

```
Ardits-MBP:teaching mia$ python3 app1.py
Enter word: rain
['Precipitation in the form of liquid water drops with diameters greater than 0.5 millimetres.', 'To fall from the clouds in drops of wate
r.']
Ardits-MBP:teaching mia$ python3 app1.py
Enter word: asdfaaaa
Traceback (most recent call last):
  File "app1.py", line 10, in <module>
    print(translate(word))
  File "app1.py", line 6, in translate
    return data[w]
KeyError: 'asdfaaaa'
Ardits-MBP:teaching mia$
```



```python
import json

data = json.load(open("data.json"))

def translate(w):
    w = w.lower()
    if w in data:
        return data[w]
    else:
        return "The word doesn't exist. Please double check it."

word = input("Enter word: ")

print(translate(word))
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattribute__', '__get
item__', '__getnewargs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__iter__', '__le__', '__len__', '__lt__', '__mod__', '_
_mul__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__setattr__', '__sizeof__', '__str__', '
__subclasshook__', 'capitalize', 'casefold', 'center', 'count', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format_map', 'index
', 'isalnum', 'isalpha', 'isdecimal', 'isdigit', 'isidentifier', 'islower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', '
join', 'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'spl
it', 'splitlines', 'startswith', 'strip', 'swapcase', 'title', 'translate', 'upper', 'zfill']
>>> "Rain".lower()
'rain'
>>> import difflib
>>> from difflib import SequenceMatcher
```


The Python Mega Course: Build 10 Real World Applications

```python
import json
from difflib import get_close_matches

data = json.load(open("data.json"))

def translate(w):
    w = w.lower()
    if w in data:
        return data[w]
    elif len(get_close_matches(w, data.keys())) > 0:
        yn = input("Did you mean %s instead? Enter Y if yes, or N if no: " % get_close_matches(w, data.keys())[0])
        if yn == "Y":
            return data[get_close_matches(w, data.keys())[0]]
        elif yn == "N":
            return "The word doesn't exist. Please double check it."
        else:
            return "We didn't understand your entry."
    else:
        return "The word doesn't exist. Please double check it."

word = input("Enter word: ")

output = translate(word)

if type(output) == list:
    for item in output:
        print(item)
else:
    print(output)
```

```
Ardits-MBP:teaching mia$
```

Report –

Application 1: Build an Interactive English Dictionary

Dictionary in Python is an unordered collection of data values, used to store data values like a map, which unlike other Data Types that hold only single value as an element, Dictionary holds key: value pair. Key value is provided in the dictionary to make it more optimized. Each key-value pair in a Dictionary is separated by a colon: whereas each key is separated by a 'comma'.

A Dictionary in Python works similar to the Dictionary in a real world. Keys of a Dictionary must be unique and of immutable data type such as Strings, Integers, and tuples, but the key-values can be repeated and be of any type.

JSON: It comes built-in with python, so there is no need to install it externally. **JSON** or **J**ava**S**cript **O**bject **N**otation is a format for structuring data.

Characteristics of JSON

- It is Human-readable and writable.
- It is light weight text based data interchange format which means, it is simpler to read and write when compared to XML.
- Though it is derived from a subset of JavaScript, yet it is Language independent. Thus, the code for generating and parsing JSON data can be written in any other programming language.

Programming e.g. –

```
# Python program showing
# use of json package

import json

# {key:value mapping}
a ={"name":"John",
   "age":31,
    "Salary":25000}

# conversion to JSON done by dumps() function
 b = json.dumps(a)

# printing the output
print(b)
```

Output –

```
{"age": 31, "Salary": 25000, "name": "John"}
```

We can say that dictionaries are the Python implementation of an abstract data type, known in computer science as an associative array. Associative arrays consist - like dictionaries of (key, value) pairs, such that each possible key appears at most once in the collection. Any key of the dictionary is associated (or mapped) to a value. The values of a dictionary can be any type of Python data. So, dictionaries are unordered key-value-pairs. Dictionaries are implemented as hash tables, and that is the reason why they are known as "Hashes" in the programming language Perl.

Dictionaries don't support the sequence operation of the sequence data types like strings, tuples and lists.

Version 1.1 (E) – Code

```
import json

from difflib import get_close_matches

data = json.load(open("data.json"))

def translate(w):

w = w.lower()

if w in data:

return data[w]

elif w.title() in data: #if user entered "texas" this will check for "Texas" as well.
```

```
return data[w.title()]
elif len(get_close_matches(w, data.keys())) > 0:
yn = input("Did you mean %s instead? Enter Y if yes, or N if no: " % get_close_matches (w, data.keys ()) [0])
if yn == "Y":
return data[get_close_matches(w, data.keys())[0]]
elif yn == "N":
return "The word doesn't exist. Please double check it."
else:
return "We didn't understand your entry."
else:
return "The word doesn't exist. Please double check it."
word = input("Enter word: ")
output = translate(word)
if type(output) == list:
for item in output:
print(item)
else:
print(output)
```

Version 1.2 (E) – Code

```
import json
from difflib import get_close_matches
data = json.load(open("data.json"))
def translate(w):
  w = w.lower()
```

```
    if w in data:
        return data[w]
    elif w.title() in data:
        return data[w.title()]
    elif w.upper() in data: #in case user enters words like USA or NATO
        return data[w.upper()]
    elif len(get_close_matches(w, data.keys())) > 0:
     yn = input("Did you mean %s instead? Enter Y if yes, or N if no: " % get_close_matches(w, data.keys())[0])
        if yn == "Y":
            return data[get_close_matches(w, data.keys())[0]]
        elif yn == "N":
            return "The word doesn't exist. Please double check it."
        else:
            return "We didn't understand your entry."
    else:
        return "The word doesn't exist. Please double check it."
word = input("Enter word: ")
output = translate(word)
if type(output) == list:
    for item in output:
        print(item)
else:
    print(output)
```