DAILY ASSESSMENT FORMAT

Date:	19/05/2020	Name:	Lavanya B
Course:	TCSION	USN:	4al17ec043
Topic:	Carrier guidence frame work, resume & cover sheet , group discussion	Semester & Section:	6th A
Github Repository:	Lavanya-B		

FORENOON SESSION DETAILS
Image of session
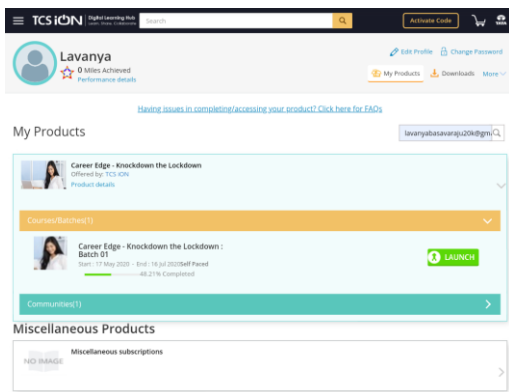
Report

Carrier guidence framework

- **Intense competition**
- **Talent acquisition**
- **Employable skills**
- **Changing job roles**
- **Employment outlook positive**

6 key pillars to get a headstart
1. **Clarity of thoughts**
2. **Access and visibility**
3. **Early preparation**
4. **Acquire relevant skills**
5. **Compelling resume**
6. **Cracking the interview**

Prepare Resume & Cover letter

Structure of resume

    Contact details
    Objective/summary
    Education
    Skills
    Personal details

3 types in Resume

1. **Chronological resume**
2. **Functional resume**
3. **Combinational resume**

Do's	Don'ts
Customize your resume to a applying sector First page should catch the interviewer Use bulletin Maintain std. Space between the lines Use keywords Font size 10-12	Don't exceed 2 pages 3-5 bullets Don't make spelling & grammar mistakes Don't use jargon/slag words Don't lie about achivement Don't copy objective from the internet

Group Discussion

GD is a positive exchange of views on a particular topic
It is a time bond & open to all the present in the group
It is offen used as mass grading process

Why GD is conducted
To check communication skills
Ability to work in a group
Ability to express your point of view
Listening skills
Knowledge of the subject
Overall personality

Points to they assessed
Activitie listening

Body language
Clarity
Tone of voice
Courtesy
Conciseness
Appropriate language
Confidence
Correctness

Do's	Don'ts
Dress formally	Don't let small group to dominate the discussion
Maintain eye contact	Don't let the discussion to drift away from the give topic
Have a neutral tone of voice	Don't interrupt mid in the discussion
Try to initiate the GD	Don't take any topic personally
Express your point of view	Don't agree with all points of views
Be open minded about the topic	Don't be biased
Keep track of time	

Date:	19/05/2020	Name:	Lavanya B
Course:	Python	USN:	4al17ec043
Topic:	Comprehension, Functions, File processing, Modules	Semester & Section:	6th A

AFTERNOON SESSION DETAILS

Image of session
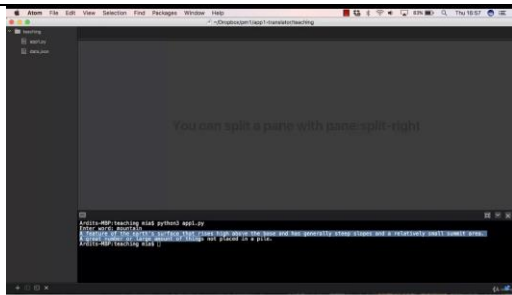
Lectures   More   ⊘

Section 12 - Imported Modules

79	✓ Builtin Modules
	CC Video - 05:53 mins
80	✓ Standard Python Modules
	CC Video - 08:34 mins
81	✓ Third-Party Modules
	CC Video - 05:49 mins
82	✓ Third-Party Module Example
	CC Video - 02:45 mins - Resources (1)
83	✓ Summary: Imported Modules
	Article

Section 13 - Application 1: Build an Interactive English Dictionary   ⊘

84	✓ Note
	Article
85	Interactive English Dictionary - How The Output ...
	CC Video - 04:10 mins - Resources (2) ⊘

Report –

Comprehension

A list comprehension is an expression that creates a list by iterating over another container.

A basic list comprehension:

```
[i*2 for i in [1, 5, 10]]
Output: [2, 10, 20]
```

List comprehension with if condition:

```
[i*2 for i in [1, -2, 10] if i>0]
Output: [2, 20]
```

List comprehension with an if and else condition:

```
[i*2 if i>0 else 0 for i in [1, -2, 10]]
Output: [2, 0, 20]
```

Functions with parameters

- Functions can have more than one parameter:

  eg.   ```
         def volume(a, b, c):
             return a * b * c
         ```
- Functions can have default parameters (e.g. coefficient):

  eg.    ```
         def converter(feet, coefficient = 3.2808):
             meters = feet / coefficient
             return meters
             print(converter(10))
         ```

Output: 3.0480370641306997

Arguments can be passed as non-keyword (positional) arguments (e.g. a) or keyword arguments (e.g. b=2 and c=10):

```
def volume(a, b, c):
    return a * b * c
    print(volume(1, b=2, c=10))
```

1. An *args parameter allows the function to be called with an arbitrary number of non-keyword arguments:

```
        eg.   def find_max(*args):
                 return max(args)
                 print(find_max(3, 99, 1001, 2, 8))
```
Output: 1001

2. An **kwargs parameter allows the function to be called with an arbitrary number of keyword arguments:

```
        eg.   def find_winner(**kwargs):
                 return max(kwargs, key = kwargs.get)
                 print(find_winner(Andy = 17, Marry = 19, Sim = 45, Kae = 34))
```
Output: Sim

File processing

We can read a existing file in python also we can create a new file and write some text in python

```
    eg.   open("file.txt") :
           content = file.read()

           open("file.txt", "w") :
          content = file.write("Sample text")
```

Modules

__Builtin objects__ are all objects that are written inside the Python interpreter in C language.

__Builtin modules__ contain builtins objects.

Some builtin objects are not immediately available in the global namespace. They are parts of a builtin module. To use those objects the module needs to be imported first.

```
        eg.   import time
              time.sleep(5)
```

A list of all builtin modules can be printed out with:

```
        import sys
        sys.builtin_module_names
```

Standard libraries written in Python reside in the Python installation directory as .py files. You can find their directory path with sys.prefix.

Packages are a collection of .py modules.

Third-party libraries are packages or modules written by third-party person.

Third-party libraries can be installed from the terminal/command line: