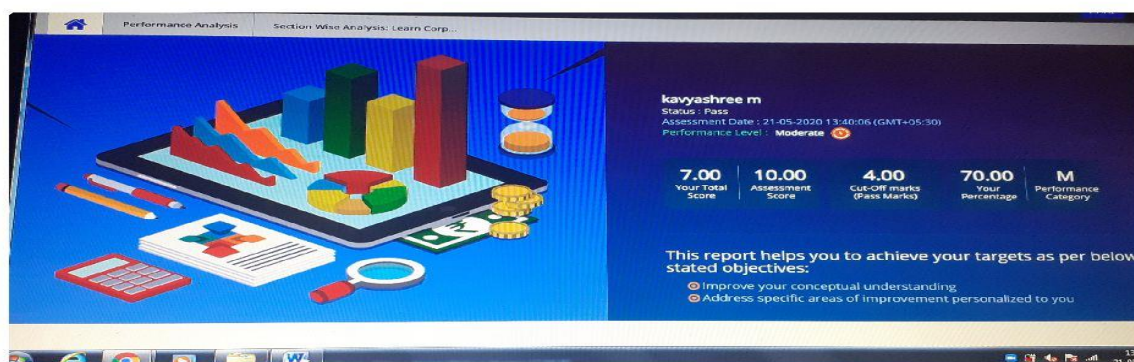


DAILY ASSESSMENT FORMAT

Date:	21-5-2020	Name:	Kavyashree m
Course:	TCSION	USN:	4a115ec036
Topic:	Write an effective mail,learn corporate telephone etiquette,basics of account	Semester & Section:	8 th A
Github Repository:	kavya		

FORENOON SESSION DETAILS

Image of session



Write an effective mail

❖ Objective

- Describe the structure of the e-mail
- develop an effective subject line and text
- utilize a few opening and closing phrases

- state the dos and don'ts of email writing
- draft an email using the pointers taught in the session

❖ **what is email?**

Email is the short form for electronic mail. It is the information sent electronically between two or more people over the network. It involves a sender and a receiver

❖ **Why use email ?**

An email is communication that happens in real time and can get. Imported data across to people in various geographies. An email is a record of the communication that has happened and is stored in the server of the organisation. One has to be very cautious while typing out a mail

❖ **Tone of voice while typing a mail**

- Formal
- Informal
- Direct
- Indirect

❖ **Dos of email etiquette**

- Use a strong subject line
- Type the correct email id
- Keep your email short
- Be polite and courteous in tone
- Use correct spelling and grammar
- Reply within a reasonable time

❖ **Don't of the email etiquette**

- Don't use email for gossip
- Don't use bright and gaudy colours
- Don't write an email when angry
- Don't reply with attachment all the time

Learn corporate telephone etiquette

❖ **Objectives**

- Essential guidelines about telephone etiquette
- The difference between the formal and informal communication
- Learn to take a call
- Learn to place calls on hold
- Learn to take messages

❖ **How to create a first impression**

- Be alert be pleasant
- Be expressive
- Be natural
- Be expressive
- Be distinctive

❖ **Dos of telephone etiquette**

- Ask the purpose of the call
- Give due importance to the call
- Acknowledge the callers queries
- Transfer the call if required

- Take permission before placing the call on hold

❖ Phrases for making phone calls

Introduction phrases

- 1) Formal phrases
- 2) Leaving message for an unavailable person
- 3) Dealing with connection errors
- 4) Closing the call

❖ Phrases for receiving the calls

- Answering the call
- Asking the name of the caller
- Asking the caller to hold on line
- Responding to the caller
- Closing the call

❖ Taking messages

- How a telephone message should be taken
- Take notes on paper
- Keep your stationery ready
- Jot down important information
- Write neatly

❖ Dos of placing the call on hold

- Seek permission before placing the call on hold
- Give them a valid reason for placing the call on hold
- Tell them how much Time you will take to return to their call
- If required, ask for more time to look up for information
- On returning to the call thank the caller for waiting

❖ **Voicemail etiquette**

Voicemails are digital recordings of incoming and outgoing voice messages. The voice mail system allows the caller to convey a message even in the absence of the called person.

❖ **Features of voicemail**

- Messages can be taken from multiple callers
- Message can be stored for a long time
- Messages can be stored extremely, locally or network or cloud storage.

❖ **Voicemail greetings**

It is very important to create an appropriate voicemail greeting.

Caller information

- Mention your name along with the organization's name
- Your contact details
- State the purpose of the call
- Mention if a return call is needed

❖ **Do's of voicemail**

- Leave a polite and professional message
- Leave a clear and understandable message
- Check your mails twice a day
- Call back at the given address as soon as possible

❖ **Don'ts of voicemail**

- Don't mention a deadline for the call back.

- Don't forget to leave your contact details
- Don't talk to others while recording/ leaving a message
- Don't make/create any sort of noise/background music.

❖ **Essential guidelines for telephone etiquette**

- Answer the phone promptly.
- Identify your organization and then yourself
- Show your genuine interest in the caller
- Address their problems positively

❖ **Basic telephone courtesies**

- Speak with enthusiasm
- Be soft and polite
- Avoid chewing gum while talking
- Hang up on the caller gently
- End the conversation with a positive note

Basics of accounting

- Accounting is a system which collects and processes financial information of a business.
- Accounting is called as language of business for recording transactions in systematic manner has given rise to book keeping. Only transactions related to business expressible in money terms are recorded.
- The process of accounting is below.

❖ **Accounting cycle**

- Transaction
- Journal

- Ledger
- Trail balance
- Trading account
- Profit and loss account
- Balance sheet
- Balance sheet opening

❖ **Double entry system of accounting**

- Each amount recorded in at least TWO accounts
- Accounting equation approach
- Expense is a loss to the company and reduces capital
- Income is a profit to the company and hence increases capital

❖ **Accounts**

- Personal
- Impersonal

❖ **Journals and ledgers**

Journals

1. Process of making entries in the books
2. Gives snapshot of an account with all the separately with double entry system

Ledges

1. It is the principal book containing all the accounts
2. Gives snapshot of an account with all the transactions

❖ **Accounting assumptions**

1. Accounting entity assumption
2. money measurement assumption
3. Accounting period principle
4. Going concern assumptions

❖ **Accounting principles and concepts**

- Dual aspect principle
- Historical cost concept
- Matching concept
- Full disclosure concept
- Verifiable and objective evidence concept

❖ **Modifying principles**

- Cost benefit principle
- Materiality principle
- Consistency principle
- Prudence principle

❖ **Final accounts**

- It shows the profitability and financial soundness of the business
- Prepared at the end
- For external reporting and internal needs of the management like planning etc.

❖ **Income statement**

Income statement will show profitability during the given period.

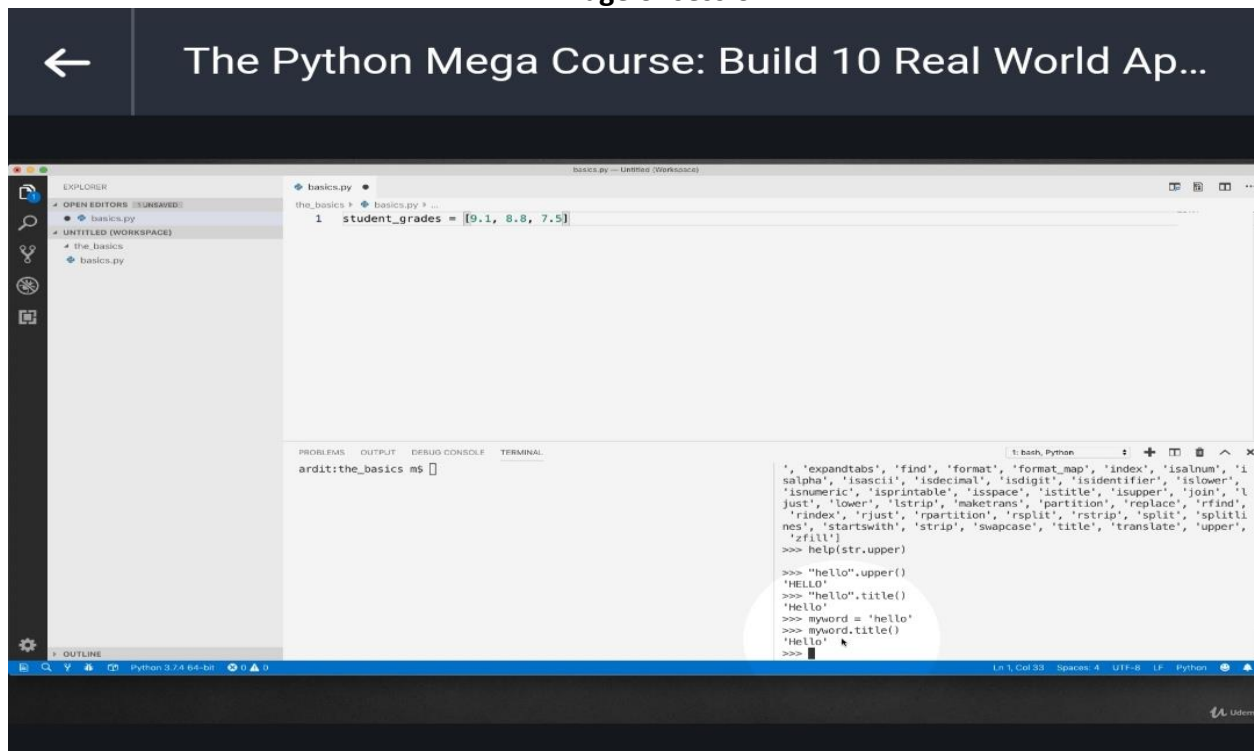
❖ **Revenue and expenses**

- ❖ Assests
- ❖ Liabilities
- ❖ Debit and credit rules

AFTERNOON SESSION DETAILS

Date:	21-5-2020	Name:	Kavyashree m
Course:	Python programming	USN:	4al15ec036
Topic:	Class and instance attribute in pythonand tuples in python	Semester & Section:	8 th A
Github Repository:	kavya		

Image of session



Class and Instance Attributes in Python

- ❖ Class attributes

It belong to the class itself they will be shared by all the instances. Such attributes are defined in the class body parts usually at the top, for legibility.

filter_none

edit

play_arrow

brightness_4

Python code

```
class sampleclass:
```

```
    count = 0    # class attribute
```

```
    def increase(self):
```

```
        sampleclass.count += 1
```

Calling increase() on an object

```
s1 = sampleclass()
```

```
s1.increase()
```

```
print(s1.count)
```

Calling increase on one more

object

```
s2 = sampleclass()
```

```
s2.increase()
```

```
print(s2.count)
```

```
print(sampleclass.count)
```

Output:

1

2

2

❖ Instance Attributes

Unlike class attributes, instance attributes are not shared by objects. Every object has its own copy of the instance attribute (In case of class attributes all objects refer to single copy).

To list the attributes of an instance/object, we have two functions:-

1. vars()– This function displays the attribute of an instance in the form of a dictionary.

2. dir()– This function displays more attributes than vars function, as it is not limited to instance. It displays the class attributes as well. It also displays the attributes of its ancestor classes.

filter_none

edit

play_arrow

brightness_4

Python program to demonstrate

instance attributes.

class emp:

def __init__(self):

self.name = 'xyz'

self.salary = 4000

def show(self):

print(self.name)

print(self.salary)

e1 = emp()

print("Dictionary form :", vars(e1))

print(dir(e1))

Output :

Dictionary form : {'salary': 4000, 'name': 'xyz'}

['__doc__', '__init__', '__module__', 'name', 'salary', 'show']

Tuples in python

A tuple is an immutable sequence of Python objects. Tuples are sequences, just like lists. The differences between tuples and lists are, the tuples cannot be changed unlike lists and tuples use parentheses, whereas lists use square brackets.

Creating a tuple is as simple as putting different comma-separated values. Optionally you can put these comma-separated values between parentheses also. For example –

```
tup1 = ('physics', 'chemistry', 1997, 2000);
```

```
tup2 = (1, 2, 3, 4, 5 );
```

```
tup3 = "a", "b", "c", "d";
```

The empty tuple is written as two parentheses containing nothing –

```
tup1 = ();
```

To write a tuple containing a single value you have to include a comma, even though there is only one value –

```
tup1 = (50,);
```

Like string indices, tuple indices start at 0, and they can be sliced, concatenated, and so on.

Accessing Values in Tuples

To access values in tuple, use the square brackets for slicing along with the index or indices to obtain value available at that index.

For example –

```
tup1 = ('physics', 'chemistry', 1997, 2000);
```

```
tup2 = (1, 2, 3, 4, 5, 6, 7 );
```

```
print "tup1[0]: ", tup1[0];
```

```
print "tup2[1:5]: ", tup2[1:5];
```

When the above code is executed, it produces the following result –

```
tup1[0]: physics
```

```
tup2[1:5]: [2, 3, 4, 5]
```

Updating Tuples

Tuples are immutable which means you cannot update or change the values of tuple elements. You are able to take portions of existing tuples to create new tuples as the following example demonstrates –

Example :

```
tup1 = (12, 34.56);
```

```
tup2 = ('abc', 'xyz');
```

Following action is not valid for tuples

```
# tup1[0] = 100;
```

So let's create a new tuple as follows

```
tup3 = tup1 + tup2;
```

```
print tup3;
```

When the above code is executed, it produces the following result –

```
(12, 34.56, 'abc', 'xyz')
```

Delete Tuple Elements

Removing individual tuple elements is not possible. There is, of course, nothing wrong with putting together another tuple with the undesired elements discarded.

To explicitly remove an entire tuple, just use the **del** statement.

Example :

```
tup = ('physics', 'chemistry', 1997, 2000);
```

```
print tup;
```

```
del tup;
```

```
print "After deleting tup : ";
```

```
print tup;
```

This produces the following result. Note an exception raised, this is because after **del**

tup tuple does not exist any more –

```
('physics', 'chemistry', 1997, 2000)
```

After deleting tup :

Traceback (most recent call last):

```
File "test.py", line 9, in <module>
```

```
print tup;
```

```
NameError: name 'tup' is not defined
```