

①

MODULES IN PYTHON

Dt 8/04/2020
WEDNESDAY

Organising python code

- ① functions
 - ② class, methods-
 - ③ OOP, functional programming
- linking the files in python

-- MODULES

main.py

utility.py

import utility

print(utility)

utility.py

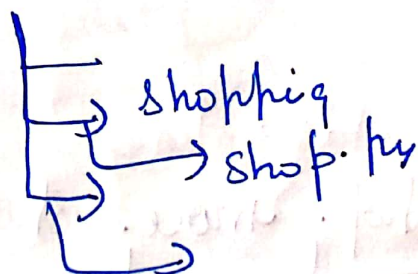
def multiply (num1, num2):

return num1 * num2

def my-own-add (num, num

return num1 + num2

Packages in python



main.py

import shopping-cart
import utility

shopping

import shopping-cart

shop is a folder created
for .py files

A package is simply a folder, Level up containing
Multiple Modules in it

Pycharm shop → under root of this folder you
need to have --init-- .py
coz the interpreter to know
or ~~can~~ read it as python package.

Rightclick → python package.

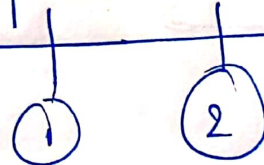
Different way of importing python

Eg: Package inside a package

Refactor

Accessing: import shop.module.shopCart

• goes recursively



instead of import we can say.

3

from shopping . more . shopping . Cart import
↓ ↓ ↓
Packagename Packagename Packagename

import <function name>

print (buy ('apple'))

from utility import multiple, divide
importing multiple files

name collisions

if the same function name is given

Eg: def max(c)

return "oops"

inbuilt max and the max defined by

of completely diff

from utility import *

-- name --

-- main --

if -- name == 'main'


```
if __name__ == '__main__':
```

```
    print(__name__)
```

when you import module in python it finds the module runs code & adds the code to memory & goes on to next line.

```
    print(__name__)
```

↓
prints the file name & gives the name specifically for main file.

if you print of add __name__ in main file even you rename it it prints __main__

```
if __name__ == '__main__':
```

```
    print("Please run");
```

↓
i.e the file name is main

```
class student1:
```

```
    pass
```

st1 = student1() → Instantiating

```
print(type(st1))
```

-- main -- this was the class which is being created in the Main file

if -- name == 'main' --:
Then do something

Built-in Modules

Python Module Index

random

print (random)

Libraries - 3.7

Built-in

help (random) → Prints documentation

dir (random) → Path

Shows all the methods in that package.

print (random.random())

randint(1, 10)

start

end

Between

Between

choice ([1, 2, 3, 4, 5])

picks one element from the given

shuffle

random.shuffle

Quick documentation P 1⁶

import random as ran

ran.shuffle(my-list)

import sys

sys.argv

first = sys.argv[0] ①

last = sys.argv[1] ②

Local

print ('hi' + first + last)

Giving input via command line

Python main.py Alpha numbers

Parameters

24/10/18