

Subject.

python arg ~~for jsl~~

Date.

Django  $\Rightarrow$  جملہ تکمیل

Single comment  $\Rightarrow$  #

#! python is a command line tool \*

#!usr/bin/python interpreter will run  
unlike windows binary

''' ... ''' Multi-line comment \*

Python 2  $\rightarrow$  print statement \*

Python 3  $\rightarrow$  print()

raw\_input  $\rightarrow$  Python 2 \*

input  $\rightarrow$  Python 3

indentation  $\rightarrow$  یہ کیا کیا کیا \*

Space [4]

Sublime  $\rightarrow$  اپنے editor \*

Salam.lower()  $\rightarrow$  "Salam" lowercased \*

ctrl + space  $\rightarrow$  tab word \*

PAPCO

Subject.

\* Python

Date.

\* Containers : List  
Tuple  
Dictionary

\* List → students = [ ' ', ' ', ' ' ]  
students.append( ' ' )

'ali' in students → true  
False \*

'dog'.lower() in students

العنصر موجود في القائمة  
ما اسم العنصر الذي يوجد في القائمة \*

students.index('hashem')  
→ 0

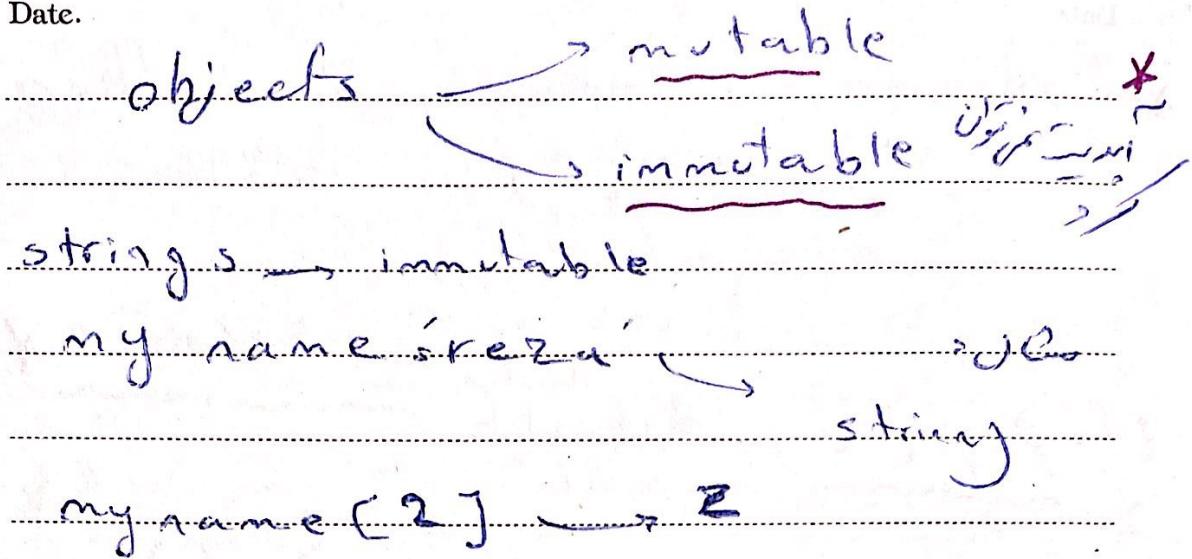
جديد  
ctrl + n → new file \*

if not in file:  
exit()

P4PCO

Subject.

Date.



\* Tuple → names = (" ", " ", " ")

→ ~~list~~ list with option

→ immutable, integers

\* for in → PHP's foreach loop

C# net gave → memory safe

>= operator <= Syntax \*

جایزه ای که پس از اینجا می بینید

Subject.

Date.

\* For  $i$  in names

print( $i$ )

= end joins \*

the pop(0) default

fruits . remove('orange')  
, join =

the reversed)

fruits . insert(0, 'apple')

fruits . sort()

to 'A' => abracadabra  
string concatenation

fruits [1 : 3] mission  
1, 2, 3 1:3

len (fruits)

fruits [0 : 7 : 2] reverse

if len(fruits) >= 1  
, an

Subject.

Date.

For i in reversed-name

print(i, end='')

← carriage → CR(↑) → ↑↑↑↑  
↑↑↑↑ → گھوٹا کھوٹا کھوٹا کھوٹا  
line feed → ↗

\* list(name) → string list(string) \*

\* del / len list ~ over string

\* cast → if list it is del just \*

(\* original reversed-name) \*

at \* e \* r \* i \* l \* a \* c \* o \*

'' join (reversed-name)

↳ join just

\* copy → cloning \*

name2 = name → copy \*

object = object → copy \*

Subject.

Date.

## Dictionary

students = { }

dictionary \*

Students = { <sup>key, value</sup>   
 "names": "mohammad",   
 "age": 18,   
 "id": "15" } }

Keys = <sup>list</sup> of keys, values \*  
Value = <sup>ctrl + d</sup> / <sup>لیٹری</sup>   
<sup>inv</sup> <sup>لیٹری</sup> <sup>لیٹری</sup>

word wrap = view <sup>for</sup> \*  
scrolling <sup>is</sup> in position

word map, Document map \*

students.keys() \*

for ja in students.keys()  
key \*

ja + <sup>→</sup> students[ja])

for ja in students

key result

PAPCO

Subject.

Date.

students[get('daniel')]  $\rightarrow$  دستیت حاصل  
= students['daniel']  $\rightarrow$  اندسادنیم  
انساندارم

DIC  $\rightarrow$  دیکشنری \*

students[''] =  $\rightarrow$  اگر = فارغ  
popitem()  $\rightarrow$  اخر  
(key, value)  $\rightarrow$  پیشیزه

pop('')  $\rightarrow$  اولین رکورد  
dic.pop('')  $\rightarrow$  دلخواهی داشتن  
(key, value)  $\rightarrow$  key, value کو  
name1, name2 = 'ali', 'reza' \*

name, scores = students.popitem() \*

tuple((),)  $\rightarrow$  جفت از این دو

PAPCO

Subject.

Date.

Set (colors) set  $\infty$ \*  
→ trivial work

List (set(colors))

• colors = حروف رنگی پردازی شده اند

• day = دن هایی داشتیم می خواهد

O-daniel 12

I-reza 14

For i in range(1, 4) \*

    print("Hello")

also printable by print " " \* print

• case ← Python \*

Sensitive

Subject.

Date.

= For i in students.keys()\*  
= for j in students  
j, values Default, so ->  
right

my\_keys = list(students.keys())\*  
print(my\_keys[st\_num]) \*

off by one error \*

print(students[my\_keys[st\_num]])\*  
cause

Escape characters \*

Escaping \ to do escape

مراجع  
www = Python oreilly

learn python the hard way

automating the boring stuff

Subject.

Date.

no scratch

think like a programmer

@Sina-kh

: Sina

Sina.Kh.se@gmail.com

{ Linux → ln \*

{ win → tr

mac → tr ln

windows →

tr ln

shift + backspace ↪ \*

shell ↪ pull, end press

jadi github → pengalih \*

download

git bash → terminal base

shell ↪

□ lb → nub back ↪ \*

Space

P4PCO

Subject.

Date.

□ a

{ " " "

" " "

private حفظ > \*

< public

" "

" "

" "

public \*

\* def → define

\* parentheses

\* [optional] arguments

def some\_name():  
function body

some\_name()

نویسیدن یک تابع با نام some\_name \*

{ parameter → یکم  
argument → هفتم فراخوانی \*

PAPCO

Subject.

Date.

كفرن و كفرن

Variables Scope: مفهوم \*

global name

local name /

WV

مفهوم و المفهوم : عرض

local, global

Business logical Plan \*

بروتوكول - (بروتوكول) بروتوكول  
لعمودي جامد

عن فتح

php type juggling

True偽، لورا و booleans  
ستكون مثل هذا

PAPCO

Subject.

\* : بحث

Date.

Save the notes on the language

1 - حارم از اول درجه می باشد  
2 - => علمند می باشد  
10 - پسندیده می باشد

ultra class

File I/O → Input output \*

write → ایجاد  
read → نمایش  
append

: cmd ↗

echo "salam" > salam.txt  
→ حذف  
append -

type

PAPCO

Subject.

Date.

open ( ) => append -> it's read \*

my\_file  
f = open ('password.txt', {w, 'r'}) \*

with - give file

- original

my\_file.read('password') \*

print(f) print(f) it's read \*

my\_file.readlines(), dir()

write

+ " \r\n " )

on - go back

" a + " is \*

it's write, it's append

will close file & read it

Subject.

Date.

one line del = off  
per line lot 19 per sec

one  
line  
lot

replace ('\\r\\n', ''')

strip off quote  
strip() - 2.

CSV Reader \*

ali, reza, maryam

Save as CSV (option)

Sublime Text Editor

C# → editor → VI mode

process explorer → Windows \*

Windows

→ Windows

task

manager

PAPCO

Subject.

Date.

permissions  $\Rightarrow$  a(b) ?

file mode - s1

original is protected by \*

del  $\Rightarrow$  delete

del ~~\*~~

import sayHello  
from sayHello import \*

$\rightarrow$  SayHello.salam("reza")  
vs Salam ("reza")

import os

os del \*

os.getcwd()

os.mkdir("Session-01")

PAPCO.....

Subject.

Date.

• file listdir os  
• file remove os  
• file update os

□ os.listdir ("Seasons")

• file remove os

□ os.listdir ('.')

('CS 111')

('' '' '')

for i in os.listdir ('.') :

if "Season" in i :

→ remove os.rmdir(i)

→ print i

→ print os.listdir(i)

import requests as req

target\_url = " " "

Subject.

Date.

□ \*os.listdir('..')

□ \*os.system("dir")

□ \*os.system("remove ('ll')")

□ \*os.name

'nt'

os.system("ls")

□ \*os.walk('..')

↳ listdir return isdir for  
      in os walk

P4PCO

Subject.

Date.

\* For my\_folders my\_folders Folders  
my\_folders files in os.walk(''):

print my\_folder <sup>folders</sup>  
+ [for my\_folder content,

\* os.path.abspath(my\_folder)  
c:\file.txt file exist

module

ext. \*

- organize code
- Code management
- clean coding

--init\_\_.py → \_\_init\_\_.py  
init, add the package \_\_init\_\_.py  
from \_\_init\_\_.py import \*  
not from import services  
discovery → discover  
os

From os import \*  
python\discovery\os\

PAPCO

Subject.

Date.

package design

os.walk . object O

file size , file size O  
test - 1

for sign in file O

new file type

file size : python main \*

- Desktop, web develop, AI,
- Data mining, bitcoir, web interact
- Image processing, Game

file size = 1000 L

send sms

call

download

security

network

web Develop

Subject.

Date.

.NET (C#) → test.exe \*

Dependency → <sup>DotNet Core</sup> <sub>Framework</sub> <sup>Libraries</sup>

Output → test → test.exe

→ gives platform independent binary

→ works on all operating systems \*

(Windows, Linux, Mac) crossplatform \*

(ARM, GPU) until

(ARM v7) yes

→ can install

python editor ← idle \*

PyCharm

3.2 ← Python

mac → terminal → brew install python

linux → apt-get install python

Codecademy Python 3 <sup>111</sup> working

Coursera

edX

PAPCO

Subject.

Date.

IDE = integrated Development Environment

IDEs → interpreter or compiler based  
→ compiler is like  
translator which converts  
language into machine language

CIS

PAPCO

Subject.

\* python

Date.

for + tab  
def + x

<= sublime > \*

vs code <= visual studio \*

PyCharm <= download link

Python Editor

Font => source code pro semi bold

: PyCharm, is / run file \*

add config => python.exe

interpreter => Python 3.6.4

background color = Run Title \*

dark config & = Desktop app \*

text, form &  
box

material theme for pycharm \*

Presentation mode <= view title \*

idea, ideas config file <= idea \*

idea project settings => project

python

P4PCO

Subject.

Date.

لـ wheel طریقہ کوئی PyCharm \*

setting, ~~setting~~

setting,

wheel  $\Rightarrow$  change font size

Alt + enter  $\Rightarrow$  ~~no error~~ ~~fix~~ \*

int - ing, ~~int~~

Alt + enter  $\Rightarrow$  ignore error like  
~~this~~ ~~no big deal if it is wrong~~

- in case  $\leftarrow$  PHP \*

sensitive

Python

ctrl + alt + L  $\leftarrow$  reformat = pretty

refactor  $\Rightarrow$  rename ~~say~~ \*

no code break

Scroll  $\Rightarrow$  pin scroll file over \*

add - pin over

P4PCO.....

Subject.

Date.

نحوه فی المنهجیات  
- pass page

Error Handling \*

C7 debugging

Struct int  $\Rightarrow$  struct struct \*

نحوه

WAF  $\rightarrow$  web Application Firewall \*

نحوه

break point  $\Rightarrow$  توقف \*

نحوه

Step over = next character \*

Step into  $\Rightarrow$  دخول Debug

On the fly  $\rightarrow$  نسخة

On fly  $\rightarrow$  نسخة

Debug

Subject.

Date.

try s

overwriting  
polish together

except:

Exception Handling

else s

finally

S E H (structured Exception Handling) ~~is a exp~~

using try - catch  
else if - else if - else  
finally try - except

ctrl + w  $\Rightarrow$  ~~ctrl~~  $\rightarrow$  ~~ctrl~~  $\rightarrow$  def ~~\*~~  
getbrainwashed

import colorama No grisebleu ~~\*~~  
colorama . init()

PAPCO print( colorama. Forenti GIT(YAN  
EX)

Subject.

Date.

shift + tab → integrate with code \*

if you have some function  
which does not  
have return value  
then ->

except Exception as khata \*  
print(khata)

JAD : ~~exception~~ \*  
except

python Scripts and tools \*

Tools

scripts = \scripts

physical Environment or path  
variable

1. /usr/local/bin/ = pip \*

2. /bin/ (23)

Pip install module-name

PEP ⇒ guidelines

PEPCO

Subject.

Date.

پیلے دلپیں جو اسی کا \*

جسے تباہ کرنے

Say-my-name <sup>def</sup> سے نام

کہا تو نہیں لیں @

ویں کوپی اسی کا سچان دے کر جو اسی کا \*

(a-z, A-Z, -)  $\Rightarrow$  پیلے دلپیں جو اسی کا

built in method  $\Rightarrow$  - - - \*

implist, \_\_str\_\_ صولت

پیلے دلپیں

(a-z, A-Z, -) (a-z, A-Z, -, 0-9) \*

گلیکسیز

PYPI.org  $\Rightarrow$  instapy : دلپیں

لے - لے - لے

fortiweb  $\Rightarrow$  web ڈرولیک \*

Hackerone

bug crawl

web ڈرولیک

bug رکھی

PAFCO Facebook bug

رولک

Subject.

Date.

CVSS → Score from 0 to 10 based on ~~to~~ \*

(First.org/CVSS) →

OWASP → Application security project  
(open web application security project)

RegEx → Regular Expression \*

exact = , . , ,

decimal (added, int, digits)

word (string)

symbol

parentheses

slash / backslash

RegEx101.com → RegEx tester \*

Id +

int

variable → value in regular expression \*

Id → one decimal

PAPCO

Subject.

Date.

8.26.2011

\w  $\Rightarrow$  (a-z) (A-Z) letter

\s  $\Rightarrow$  space

\n  $\Rightarrow$  enter

\\*  $\Rightarrow$  any

@radia-feed  $\Rightarrow$  insta podcast

radiant  $\Rightarrow$  ✓

radi regex  $\Rightarrow$  one

pattern = re.compile(' \w+')

pattern.findall(texts)

Subject.

Date.

mat final (text) \*

प्रधान संस्कृत

वाल " ए ज्ञा शिव विष्णु "

मूर्ति विष्णु

→ विष्णु

लक्ष्मी विष्णु

108

(1) + (1) = लक्ष्मी विष्णु

{ groups  
plays }  $\Rightarrow$

ग्रामल → sina-har-sec@gmail.com \*

= SinaHarsec@gmail.com

organised

P4PCO

Subject.

Date.

.htaccess  $\xrightarrow{\text{Rewrite}}$  regular expression \*

↳ regular-expressions.info/reRewrite.html

↳ REGEX ] regular expression \*

(ir | com | biz | xyz) \*  
private or business website  
= .xyz .com .biz .info .net

domain suffix

mail@gmail | yahoo | hotmail

mail @ domain \*

(w+) . (w+) \* @ domain \*

regular expression list separator

tuple type is a list

[ (pi, robot) , (pi, robot) ]

PCPQ list tuple

Subject.

Date.

java(d?) لهم جوړو \*  
→ word spaced  
pattern = java(d?)  
word java spaced

Flags : ignore case بُلْدَه \*

Multi line

case sensitive

flags →  
ctrl + D multiline

flags → درست  
ctrl + click → فکر  
Alt + backspace → پاک

pattern = re.compile('reza', re.IGNORECASE)

re.IGNORECASE → جواز  
پیغام

PAPCO

Subject.

Date.

RE Flag  $\xrightarrow{\text{odd/even}}$  ascii \*

~~~~~~~  $\xrightarrow{\text{unicode}}$

~~~~~~~~~  $\xrightarrow{\text{java}}$

■ flag  $\Rightarrow$  global & local

• global is position specific

• always match whole line

$\hookrightarrow$  Global, Shalt-such \*

■ flag  $\Rightarrow$  re.locale  $\Rightarrow$  ignores

■ flag  $\Rightarrow$  extended ignore space

■ flag  $\Rightarrow$  unicode  $\Rightarrow$  ignores

■ flag  $\Rightarrow$  VERBOSE  $\Rightarrow$  warning  
is printed out for debugging

■ flag  $\Rightarrow$  regex  $\Rightarrow$  use = RegEx  
Detailed comment

PAPCO.....

Subject.

Date.

अंतर्गत लेन्स प्रकृति \*

assertion  $\Rightarrow$  ४६१\*

? =

ध्रुव / ध्रुवी / ध्रुवी \*

सांसार / किराश / करुष / मतदाता / वास्तव

\* look ahead, अंतर्गत लेन्स

\* look behind

(?) = १०० \*

(?) = १०० सांसार \*

Subject.

: فیصلہ

Date.

def beg.: Soltan(names'test): \*

وَيُؤْتَى بِهِ الْمُؤْمِنُونَ إِذَا قَاتَلُوكُمْ فَلَا يُؤْتُوهُمْ مَا أَنْهَا كُفَّارُهُمْ إِنَّمَا يُؤْتُوهُمْ مَا سَعَوا

أَرْجُواهُمْ لِيَقْبَلَ مِنْ رَبِّهِمْ وَلَا يُؤْتُوهُمْ مَا سَعَوا

أَرْجُواهُمْ لِيَقْبَلَ مِنْ رَبِّهِمْ وَلَا يُؤْتُوهُمْ مَا سَعَوا

def multiply(\*args)

multiply(1,2)

" (1,2,3)

وَإِذَا رَأَيْتُمْ مُّكَافَّةً مِّنَ الْمُؤْمِنِينَ فَلَا يَنْهَاكُمْ عَنِ الْمُحْسَنَاتِ إِنَّمَا يُنْهَاكُمْ عَنِ الْمُنْكَارِ إِنَّمَا يُنْهَاكُمْ عَنِ الْمُنْكَارِ

Kwargs

def showMyDictionary(\*args)

الآن نحن نعرض دیکشنری

showMyDictionary(names'sinat, lastnames'

PAPCO

kheirkhah1

Subject.

Date.

## Format strings : \*

print('my name is %s' % name)

www.dill-lawyer.com

realpython.com => introduction

■ %s → string

■ %d → digit

■ %x → hex

■ %i → integer

## anonymous functions : \*

integers, strings, lists, tuples, sets

= Expression

statement, Y/A, V, code

## Lambda Functions : \*

x=lambda arg1,arg2,arg3: expressions

it's a code block  
is, piece of code

Subject. \*

Date.

Condition -   
 plus right triangle

def myfunc(jabar),

return lambda alias: jabar

real s myfunc()

rati

• recursive function  
network

format string using f-strings \*

('Salam' + ' ' + 'az' + '\n')

but type error

{ } { } { } are .format(n, age)

{ } { } { }

{- continue} → ignore the rest of the code

{- break} → exit from loop

{- pass} → do nothing

PAPCO

Subject.

Date.

I/O with Sql (SQLite): \*

SQL (structured query language):

mssql, mysql, mongoDB, Hadoop, couchDB, (node.js)

SQLite (Relational Database) \*

myDatabase <sup>SQLite</sup> is creating, writing, reading

C1 built-in lib - SQLite \*

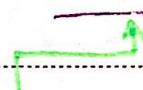
import sqlite3

conn = sqlite3.connect('pythondb.  
SQLite')

Prim oparation is to connect  
function is called to do it

Cursor => object \*

c = conn.cursor



c.execute('CREATE TABLE

students\_table ()')

**PAPCO**

Subject.

Date.

❑ int  $\Rightarrow$  print length \*  
and the next get this is also goes  
❑ id  $\Rightarrow$  unsigned int

db browser windows  $\Rightarrow$  \*

execute jobs

conn.commit()  $\Rightarrow$  yes

conn.close  $\Rightarrow$  kifas

INSERT INTO table name ( , ) \*

values ( , , , , )

string { }  
string { }

❑ cursor.fetchall

❑ cursor.fetchmany()  $\Rightarrow$  last  
use cursor size

❑ cursor.arraysize  
execute.execute

for i in cursor.fetchall()

for j in i

print j

PAPCO

سون

Subject.

Date.

int → type  
Statement → Attribute

update users\_tbl <sup>SET</sup> password = "123" \*  
                   where id = 1

Conn. commit)

(Delete From users\_tbl \*  
                   where ID = 2 )

Sql is now case-sensitive sql حروف \*

first letter of query is deleted

Drop TABLE users\_tbl \*

TRUNCATE → removes \*

OS remove ← DB حذف \*  
↓  
↓

Subject.

پڑھ دو

Date.

Geometric Plotting Software  
parties

University DB  
student\_tbl

Name / Score

b2s = 90\*

not  
and  
or

4.5 / 12 => 37.5 \*

Pypi جو پاکیزہ  
↳ mysql : mysql \*

\* Pip install mysql-connector-python

\* import mysql.connector  
\* my\_connection = mysql.connector.  
connect(user=" ", password=" ", host=" "  
PAPCO ..... database = " ")

male female ↪

Subject: Erum MySQL → engine → ~~for web, db, file~~  
Date:

User, password, host, database \*

→ connection string

Play localhost - 4 cmd  
qisad app

print host or \*

TABLES = { }

TABLES [ 'mt-pthon' ] = 0

per bar 2,000 = 46 72

)

\* my\_cursor = my\_connection.cursor()

my\_cursor.execute

web interaction with python \*

reporter and subscriber \*

سے تو

PAPCO

Subject.

Date.

class no 440

web (in urdu)

Terminology

وہب

html → hyper text markup language

web →

http → hypertext transfer protocol

request →  
response

tutorialspoint.com/http

کامپیوٹر سے ملکے \*

requests پر کرنے والے \*

\* pip install requests

نام نہیں لے سکتے اس کا دل کا دل نہیں رکھ سکتے  
کام کرنے والے کام کرنے والے

import requests

my\_site = ' url '

PAPCO

Subject.

Date.

my\_result = requests.get(my-site)

→ https://c://\*

- my\_result.text

- a = open('result.html', 'w', encoding='utf8')
- a.write(my\_result.text)
- a.close()

{beautifulsoup → ~~w3c~~ ~~css~~ ~~html~~ \*  
python request → real python}

2. python-requests - org/len/master/

Pip install beautifulsoup4: cmd> \*

```
From bs4 import BeautifulSoup  
import requests  
my-site = 'http://...'  
my_result = requests.get(my-site)  
my_result.status_code
```

PAPCO.....

Subject.

Date.

= status codes \*

- 404 → suggestions not found
- 200 → ~~in~~ ok
- 301, 302 → Moved permanently, moved temp
- 500 → Internal server error
- 400 → Bad request

if my\_result.status\_code == 200:  
    print("initial page")  
    print(my\_result.text)  
Soup = BeautifulSoup(my\_result.content)

Soup.findAll('a')

beautifulsoup tutorial  
python for beginners.com

my\_result.text.title()

Soup.findAll('a')[5].get('href')  
        ↓  
        for loop in range(5)  
            Find('a')      in range(5)

PAPCO

Subject.

Date.

: pip ~~version~~

{ python -m pip install --upgrade pip  
  cd scripts  
  pip install requests

Pip 3 - version cmd \*

help(i.getText) \*

x = script / idle shell \*  
script's primary user interface

import os

os.system('explorer') \*

Chrome user agent switcher \*\*

Subject.

: pip ~~version~~

Date.

{ python -m pip install --upgrade pip  
  cd scripts  
  pip install requests

PIP version

Pip 3 - version and \*

help(i.getText) \*

X = ~~script / idle shell~~ \*  
~~script file main.py / idle~~

import os

os.system('explorer') \*

Chrome user agent switcher \*

1. windows + F12

Subject.

Date.

result, request, headers \*  
my\_headers = {'User-Agent': 'fishif'}  
result, request, headers

get( headers )  
verify=False)

print(result.text)  
- stripping

with open('native2.html', 'w',  
encoding='utf-8') as

my\_new\_file  
my\_new\_file.write(native

text.strip())

Subject.

Date.

(Date = 06/06/2018) Topic \*

\* One Time password \*

\* Ramz Down → OTP

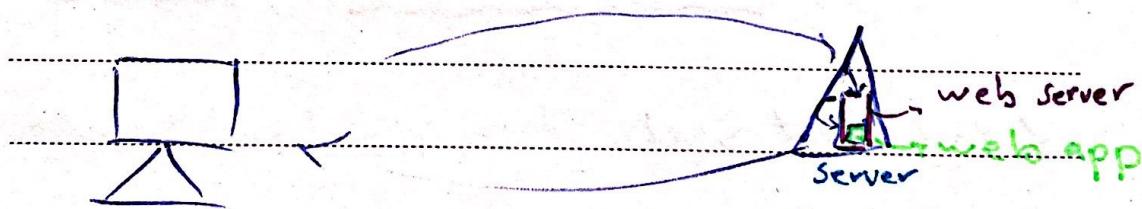
\* web application → php, ruby on rails,

\* web Service →

\* web Server →

\*\* Server →

\* web Server



Server host:

windows, linux(ubuntu,

mint, debian...)

web app (leaflet)

web server →

http, apache, Tomcat

, nginx, lighttpd,

bite speed

P4PCO

Subject.

Date.

web service → https, url \*

using api \*

api.karengarai.in/sendsmfphonenumber  
/message

req.get('api.karengarai.in/sendsmf  
109195152501/salam!')

web server → API (Application \*  
programming Interface)

API → RESTFUL(Json), SOP(XML)

<xml>text</xml> api

PWA (progressive web apps) \*

API key → time limit \*

of API key

API limit

SDK → Software Development \*

API library, module, plugin built

Subject.

Date.

web service → http://https, xml \*

JSON, API \*

API Kavengar.ir / send sms / phone number  
message

req.get('api.kavengar.ir/send sms  
109195152501/salam')

web server → API (Application \*  
programming Interface)

API → RESTFUL (Json), SOAP(xml)

<xml>text</xml> api

PWA (progressive web apps) \*

API key → token \*

at API token

API token

SDK → Software Development \*

API library, module, plugin

Subject.

Date.

pip install names \*

dir(names) \*

ctrl + spacebar

projectever.net

IPython: Interactive shell \*

anaconda → ipython \*

examples

arvancloud.com

s cmd \*

ipython

ctrl + d, exit

ipython notebook:  
Jupyter notebook

PAPCO

Subject.

Date.

shift entered ٢٠١٠

mark down: مراجعة  
الدروس

back gate مدخل

mark down sheet

ipab

filino برج مراقبة سقوط: في

الإذن بفتح المدارس

P4PCO

Subject.

Date.

- 1. SOAP → XML \*
- 2. RESTful → key-value → dictionary (JSON)

vajeyab.com → web service

www.vajeyab.com / dev

1. in mind long \*

aven cloud \*

2. in MySQL, Firebase is Backstage

3. in Django is engine

4. in is remote interface

document-oriented cross-platform MongoDB

Apache CouchDB

remote config → config

sohanak.com is

check.host

JSON \*

```
{  
    res.json()  
}  
result.keys()  
    , values()
```

PAPCO

Subject.

Date.

↳ JSON & JSON \*

{ "names": [

'ali', 'reza', 'hossein'

configurable in visual studio \*

private json is

Config.json

{

"user": "admin" ①

"pass": " "

"myName": " "

}

json.loadc

with open('config.json', 'r')

as myFile,

myConfigJsonLoad(myFile)

P4PCO

Subject.

Date.

for varhe in my result of some  
['data'][ ]  
print(varhe['text'])  
it prints on my while, write (varhe['text']) +  
'\n'

OOP براموزیس\*  
Spaghetti code لینجین  
Function base  
کسی نہیں میں {  
- } Hmc

QT - 5

class وہ (object) ایسا

code tutspoint.com

class khodro,  
pass

Subject.

Date.

for varze in my\_result['jason']  
[ 'data' ][ ]  
print(varze['text'])  
# prints all my file, while [varze['text']] +  
'\n'

OOP

Spaghetti code ~~function~~  
Function base

براموز سریع \*

ساده ساخت هست  
Hmc

QT - 5

کلاس کوئل (object) ہے

code.tutsplus.com

class khodro,

pass

P4PCO

Subject.

Date.

instance



پریس کھڈروں)

method ← مетод

property ← موصیہ

attribute ← موصیہ

init => initialize object (مکمل) \*  
it call fun

(assigning) property  
class khodras

def \_\_init\_\_(self):

print("khodra sehde shud  
abhi")

car\_1 = khodro() call گھومند

پریس self پریس پر  
کھڈروں کا ایسا ایسا  
object پر ← موصیہ

car\_1 = SayHello(Car\_1)

PAPCO بول کر حفظ کر لے

Subject.

Date.

Keyboard Shortcuts \*

def init(jakar):

print

jakar.rangs 'blue'

jakar.speeds '300'

jakar.brand = 'benz'

jakar.sankths 'gas'

def baro\_jal(jakar):

print('jakar is going to')

def farmaz(jakar):

print('jakar farmazed')

def info(jakar):

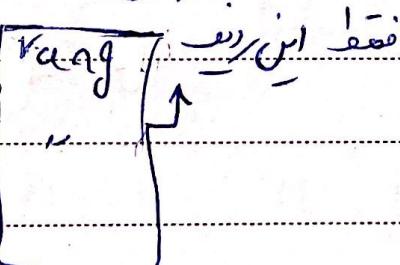
print(jakar.rang)

alt +

زن

Keyboard Shortcut \*

jakar.



PAPCO

Subject.

Date.

def info(jahat):

print(jatarang)

Colorama mit( ) : ~~import~~ \*  
import jcolorama as c  
c.init(colorama)

def Vaziat(jahat):

import time : Time JSL \*

time.time( )

import jdatetime  
jdatetime.datetime.now( )

time.strftime( "%Y" )

sim, id

time.sleep(0.2)

Subject.

Date.

## basic concepts in oop: \*

Abstraction

Poly morphism

Encapsulation

Inheritance

: Inheritance \*

class Humans:

def \_\_init\_\_(self):  
 self.gens = 11

class baba(Human):

pass

class Babo(baba):

pass

~~also (j) override~~

Abstract => abstract class / interface \*

@abstractmethod

def rah\_valta\_n(yahan):

PAFCO..... pass

Subject.

Date.

from abc import ABC, abstractmethod

class Human(ABC):

@abstractmethod  
def \_\_init\_\_(self):

def eat(self):  
 pass  
 instance("Human")  
 self.eat()  
 print("Force eat")

DevOps → Development and  
operations

: polymorphism \*

a=1      a=ali  
b=2      b=      print

a.add(b)

(ali print ali is his own name)

name

getter

: Encapsulation \*

Setter

PAPCO

Subject.

import random

Date.

class Library

def \_\_init\_\_(mayan):

pass

def showBooks(mayan):

class moshbari:

def \_\_init\_\_(karesh):

pass

import random

random.randint(1, 10)

لیستی از ده عدد

بازیگرها

Subject.

Date.

## multithreading

import time

import threading

f

multithreading : possible

for thread to b done \*

my\_kangar = threading.Thread(target=job)

my\_kangar.setDaemon(True)

my\_kangar.start()

import time, threading, random

time.time() \*

it will be

Subject.

Date.

archive < 1 to 50 > \*

for i in range(archive)

time.sleep(1) / wait

print(i+2) / print

50/10 = 5 \*

push queue new archive \*

import queue

public class

get()

queue.get() and print

threading lock \*

push to resource ready

wait for lock to be free

lock.acquire

n.release

lock and n pillars to prevent hijacking

PAPCO

onto pillars

Subject.

Date.

~~my boss / let's do it's boss \*~~

my-baghaan-empty() = True \*

False

def boss():

while not baghaan.empty(): \*

job()

when que ve job = task done) \*

task will be done and job()

(initialization)

push object in notification

empty

all pins = baghaan.join() \*

jobs = task which

application to : setDaemon \*

so crash in main

in the thread

Subject.

Date.

parent task done ✓ \*

parent empty ✓  
child join ✓

list of weight long heap - get() \*

parent job is output of child job  
↳ queues ↳ threads \*

Tree search while nodes

How many threads should we use? \*

site

login

Bruteforce

Username & password

Network, application, network, net  
in - internet - cloud

wAF → Firewall  
P4PCO  
→ Arvan cloud  
cloudflare  
cloudflare  
cloudflare

Subject.

insight hack

Date.

privacelis network →  
→ location is visible  
← copy ←  
raw headers → copy → paste →  
Edit and

1.  
120  
login → URL → new request  
data { "email": "j@gmail.com",  
"password": "123456",  
"register": "register",  
"login": "login",  
"index": "index" }

Subject.

Date.

import requests as req  
import threading

SEC 573

python for penetration tester \*

multi process ~~چیزی کو~~  
~~ریکارڈ، دسکریپٹنگ~~  
Gpu ~~کو اس کو GPU کو استعمال~~  
~~کریں، کوئی~~

Pyqt = ~~بیرونی کو کہا جائے~~ \*  
~~جسے کوئی~~

Qt Designer = ~~pyqt~~ ~~کوئی~~ \*

Qt Designer = ~~crossplatform~~ \*

- pip install pyqt5

- pip install pyqt5-tools

PAPCO

Subject.

Date.

pixels <= main window \*

Pyvics -> 01.vi - o alipy \*

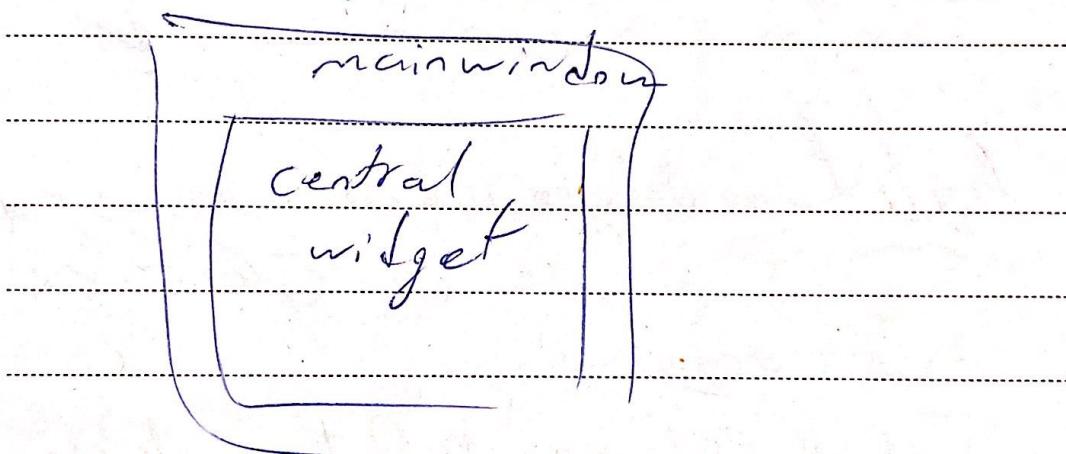
PyCharm

diff object \*

<-- int, float, str

---

tutorialspoint



15-minutes-apps

no 1 : circ

Subject.

Date.

pyinstaller → python click +  
→ exe

python installed with fbs

pp install pyinstaller

pyinstaller -o2.py → onefile → onefile o2.py → click

pyinstaller → onefile o2.py

Subject.

Date.

VS code  $\Rightarrow$  جیکس ایڈٹر

AJS - made in Iran

ny\_soup = bs(result, content, 'xml')

نگوچه را میں جس کی وجہ سے  
پیش کیا تھا جس کی

: Data science tools

cs.djgo.io/data

matplotlib.org  $\Rightarrow$  charts کو جیسے جیسے

panda  $\Rightarrow$  جیسے جیسے تکمیل

Conda install panda

pip install pandas  $\Rightarrow$  جیسے جیسے  
تکمیل

from matplotlib import pyplot as pl

X = [10, 20, 30]

Y = [5, 2, 17]

Subject.

Date.

```
plt.plot(x)
plt.plot(y)
plt.plot(x,y)
plt.ylabel()
x.xlabel()
plt.legend(["a","b"])
```

```
import pandas as pd
import os
my_data = pd.read_csv("file")
my_data.columns
print(my_data.columns[0])
print(my_data[0])
```

```
my_data[my_data.county == "united states"]
```

chart

Subject.

Date.

python data science  $\rightarrow$  learning  
 $\rightarrow$  youtube

ai with python

cs dojos

youtube  $\rightarrow$  tech with Tim \*

Youtube fabrits clips  
machine learning

john hammond

memory leak in ip

Image processing:

pip install numpy  $\rightarrow$  pixel wise  
 $\rightarrow$  and  $\rightarrow$  2d 3d

alltechbooks.org

automating boring stuff book  
codersplay

git.it  $\rightarrow$  cordova  $\rightarrow$  css html

client

PAPCO

Subject.

Date.

```
import cv2  
import numpy as np  
from matplotlib import pyplot  
  
# is it a  
(flag) lens  
my_picture = cv2.imread(''), cv2.IMREAD  
image                GREYSCALE  
dict  
  
cv2.imshow('my window', my_picture)  
  
cv2.waitKey(0)           # infinite  
cv2.destroyAllWindows()  
cv2.waitKey(1)  
  
my_webcam = cv2.VideoCapture(0)
```

Subject.

Date.

while True:

    ret, frame = my\_webcam.read()

    cv2.imshow('test', frame)

    cv2.waitKey(1)

    if cv2.waitKey(1) & 0xFF == ord('q'):  
        break

    cv2.destroyAllWindows()

cv2.waitKey(0)

cv2.destroyAllWindows()

openCV  $\rightarrow$  RGB - BGR

    cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)

    cv2.cvtColor(frame, cv2.COLOR\_RGB2BGR)

my\_frame = cv2.cvtColor(frame,

    cv2.COLOR\_RGB2BGR)

to import image

fourcc = cv2.VideoWriter\_fourcc(\*'XVID')

my\_webcam = cv2.VideoWriter('output.mp4', fourcc, 20.0, (640, 480))

P4PCO

Subject.

Date.

my\_webcam = cv2.VideoCapture(0)

outwriter = cv2.VideoWriter('output.mp4', cv2.VideoWriter\_fourcc(\*'XVID'), 20.0, frame\_size)

for frame in frames:

cv2.line(frame, (100, 200), (400, 300),

(255, 255, 255), 3)

cv2.rectangle(frame, (200, 300),

(400, 500), (0, 255), 4)

open cv face

cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml')

Easy Face Detection

my\_font = cv2.FONT\_HERSHEY\_COMPLEX

cv2.putText(frame, 'Salam', (200, 300),

my\_font, 5, (255, 0, 0))

jul

PAPCO

Subject.

Date.

✓

git clone → download

python pyinstaller a package  
virtual environment

pip freeze

pip install requirements.txt

pip virtual environment → venv

python3 -m venv

size reduction in size of

pip virtual env is possible