**Print (“mina/nhosam”) equal mina**

**Hosam**

**print(“mina”+’ ‘+”hosam”)equal mina hosam**

**Arthimetic Operators**

**1+2equal3**

**2-1eqal1**

**2\*2equal4**

**6/3equal2**

**7//3equal 2 flour devision**

**7%3equal 1 modules 7%4equal3**

**round(5/6)equal 1**

**Variables**

**Num1=5**

**Num2=6**

**Num3=Num1+Num2**

**print(Num3)**

**equal 11**

**name=”mina”**

**print (name)**

**equal mina**

**variable multiple assignment**

**num1,num2,num3=2,3,4**

**print (num1+num2+num3)**

**9**

**Data types**

**Numbers = int**

**Names =str**

**True , False=bool**

**Numbers as 1.5=float**

**print (type(num1))**

**class ‘int’**

**Input or read data**

**Name=Input(“enter your name”)**

**print (“name is”+Name)**

**name is mina if I enter minandd**

**Other types of data**

**ord() function in python**

**Given a string of length one, return an integer representing the Unicode code point of the character when the argument is a unicodeobject, or the value of the byte when the argument is an 8-bit string. For example, ord(‘a’) returns the integer 97, ord(‘€’) (Euro sign) returns 8364. This is the inverse of**[**chr()**](https://www.geeksforgeeks.org/chr-in-python/)**for 8-bit strings and of unichr() for unicode objects. If a unicode argument is given and Python was built with UCS2 Unicode, then the character’s code point must be in the range [0..65535] inclusive.**

**Convert between types**

**a=int(True) a=1**

**a=bool(1) a=True**

**a=ord(“a”) a=67**

**a=chr(67) a=a**

**Import random and randint**

**Import random**

**X=random.randint(اول رقم ,اخر رقم)**

**print(x)**

**or**

**from random import randint**

**Assignment operators**

**+= (تعنى ضيف رقم ع الموجود )or-=or\*=or/=**

**mean already add number to number recorded**

**Logical operators**

**And**

**True And True equal True**

**True And False equal False**

**Or function**

**True Or True equal True**

**True Or False equal True**

**False Or False equal False**

**Not function**

**Not True equal False**

**Not False equal True**

**Comprasion operators**

**== معناها التساوى فى القيمة**

**!= معناها لا يساوى**

**<= اصغر من او يساوى**

**>= اكبر من ااااو يساوى**

**Dictionary python**

**p1={key:value,key:value}**

**علشان اجيبvaluueجوهkey**

**print(p1[key])**

**علشان اغير value**

**p1[key]=new value**

**Tuple python**

**p1=(value,value,value)**

**print(p1[index])**

**index is the number of the value begin from 0**

**I cannot change any value in tuple**

**List python**

**p1=[value,value,value]**

**print(p1[index])**

**I can change any value in list**

**p1[index]=new value**

**Add and del in python**

**P1=[v1,v2,v3]**

**P1.append(v3) add one value**

**P1.extend([v4,v5,v6]) add more values**

**del(P1[index]) del one value**

**del(P1[index],P1[index]) del more values**

**Nested list python**

**print(p1[index of the list][index of the value of the list])**

**21-Length on python**

**len():determine how many value in list or tuble,in string how many character**

**22-hex,oct,dec,bin on python**

**hex(decimal no.)or format(decimal no.,x) = hex no.**

**oct(decimal no.)or format(decimal no.,o) = oct no.**

**bin(decimal no.)or format(decimal no.,b) = bin no.**

**int(“hex no.”,16)or int(“oct no.”,8)or int(“bin no.”,2) = decimal no.**

**23,24-If statement**

**if condition:**

**do that if condition is true**

**elif another condition:**

**do that if another condition is true**

**else:**

**do that if all condition is false**

**25-Range in python**

**X=range(first value,endvalue,amount of increase or decrease)**

**The output of range is list**

**26,27-for loop**

**for new variable in variable:**

**do that**

**28-29-while**

**while condition:**

**do that**

**else:**

**exit**

**30-do u want to use program again**

**Define variable as equal yes or no and make IF statement and the condition if variable is equal yes use program again by using input in the end of program if I want use program again**

**31-special loop for dictionary**

**for (two variables) in dictionary:**

**variable1=key,variable2=value**

**dictionary.items() : shows all items in dictionary (key and value)**

**ex: for x,y in dictionary.items()**

**32-infinite loop python**

**Increase without condition**

**In for loop**

**fromitertools import count()**

**for x in count():**

**print(x)**

**33-break infinite loop**

**If the condition approved we put break: the program will finish**

**34-continue in loop**

**Exchange break by continue**

**The difference between break and continue in break program will end but in continue the program won’t done the other operations but it will back to it’s start and continue loop**

**35-string litterals**

**\r any word write after \r wrote in the beginning**

**\n any word write after \n wrote in the next line**

**\a alarm as nodifications**

**\b any letter wrote after \b wrote instead of the end letter**

**\b\b two letters**

**\t make tab**

**\x**

**36-string multiline repeat**

**\n\*3 = ‘’’’’’**

**when--------------print(‘ahmed’\*3)-------------ahmedahmedahmed**

**37-string litterals**

**\N show signs in keyboard**

**print(‘\N{copyright sign}’)**

**\x 2-digits**

**\u 4-digits**

**\U 8-digits**

**38-substring**

**Substring with index**

**my=’welcome’**

**print(my[0:4])----------------welc**

**39-string format**

**%s:%(‘name’) or %number or %float**

**%d:%number or %float**

**%s,%s,%s:%( )**

**40-Detail string format**

**%c:%’character’or number equal charater or character equal number in ASC table**

**%i**

**%e**

**%E**

**%g**

**%G**

**%f**

**%u**

**%o**

**%b**

**41-upper&lower&isupper&islower**

**42-find string**

**New variable=variable which I want to search in it.find(‘search’)**

**43-check from string**

**print(‘abcABC’.isalpha())--------------------True**

**print(‘abcABC123’.isalnum())------------------------True**

**print(‘123’.isdigit())------------------------True**

**print(‘123’.isnumeric())------------------------True**

**print(‘ ’.isspace())------------------------True**

**44-replace string**

**Variable.replace(‘old thing’,’new thing’)**

**45-split string**

**New variable=old variable.split(, or or .)**

**the result of split is list**

**46-join string**

**New variable=(- or .).join(old variable)**

**if I make join to string it will be joined between each character**

**47-file w r open**

**Variable=open(‘name of file I want to create’,’w+’)**

**Plus indicates read and write**

**‘w’ for create and write,’r’for read , ‘a’ for append**

**48-write on file**

**After create in file**

**file.write(‘what I want to write’)**

**file.close()**

**or**

**file.writelines()-------------to write from list**

**49-read from file**

**file.read() or file.readlines() or file.readline()**

**file.readlines() shows the result in list**

**print(file.readline)----------print the first line in list**

**file.seek(0) to go to start**

**file.seek(1)**

**معناها الغى اول حرف**

**50-append file**

**Append=overwrite on the file not damage file**

**51-read write binary file**

**uses when we want to read or write audio file we use rb or wb**

**52-modes of open file**

 r for reading

 r+ opens for reading and writing (cannot truncate a file)

 w for writing

 w+ for writing and reading (can truncate a file)

 rb for reading a binary file. The file pointer is placed at the beginning of the file.

 rb+ reading or writing a binary file

 wb+ writing a binary file

 a+ opens for appending

 ab+ Opens a file for both appending and reading in binary. The file pointer is at the end of the file if the file exists. The file opens in the append mode.

 x  open for exclusive creation, failing if the file already exists (Python 3)

53-with……as python

the main uses of with function is instead of file.close()

with open(“”,””) as variable name:

what I want to write

54-**create folder or directory**

**import os**

**os.mkdir(‘folder name’) or os.makedirs(‘folder name’)**

**55-if exists file folder**

**if os.path.exists(‘file name’):**

**56-delete files and folders**

**os.remove(‘file name’) or os.rmdir(‘folder name’) or os.removedirs(path)**

**57-show files in dir**

**os.listdir(‘folder name’)**

**to get files not folders os.path.isfile(full path)=(‘folder name\\’+file)**

**58-system run files and folders**

**os.system(‘file name’)**

**if file name contain space os.system(‘\’file name\’’)**

**to create folder os.system(‘mkdir\’folder name\’’)**

**59-create list from for**

**New variable=[variable for variable in range()]**

**60-all list file from for**

**import os**

**s=[p for p in os.listdir('New folder') if os.path.isfile('New folder/'+p)]**

**print(s)`**

**isfile for files**

**isdir for folders**

**61-copy and paste files**

**Import shutil**

**Shutil.copy2(‘place where the file in’,’new name of the file’)**

**62-cut and rename**

**shutil.move(‘’,’’)**

**63-copy folders**

**Shutil.copytree(‘’,’’)**

**64-conditional operator**

**f1=”t1” if v1>v2 else ‘t2’**

**65-print date and time**

**Import datetime**

**datetime.datetime.now()**

**66,67-date and time**

**custom for date and time**

**day=now.day**

**month=now.month**

**year=now.year**

**hour=now.hour**

**minute=now.minute**

**format of datetime**

**print(now.strftime(‘%a’))**

**watch video 67 to know abbrevation**

**68-regular expression**

**Import re**

**Re.match(second,one)**

**69-math python**

**print(math.pi)----------22/7**

**print(math.sqrt(81))----------------- 9 sqrt**

**print(round(1.4))----------------1**

**print(round(1.5))---------------------2**

**print(math.ceil(1.1))--------------2**

**print(math.floor(1.99))---------------1**

**print(abs(-10))----------------10**

**print(math.fabs(-20))-----------20**

**print(pow (5,3))-------------125**

**print(math.pow(5,3))---------------125**

**print(math.factorial(5))----------------120**

**print(sum[1,2,3,4,5])-------------------15**

**70-error exception**

**try:**

**(method)**

**except ‘error name’ as y:**

**print(y)**

**71-set function**

**x=[‘a’,’s’,’d’,’w’,’e’]**

**y=set(x)**

**y >============= {a,s,d,w,d}**

**add on set function**

**x=set()**

**x.add(‘a’)**

**x.add(‘s’)**

**x.add(‘d’)**

**x={a,s,d}**

**not allow to repeat same values**

**72-set check**

**l1={5,4,3}**

**print(5 in l1)**

**>=================True**

**73- set(clear-remove-discard)**

**set clear() to remove all items in set function**

**set.remove() to remove selected items**

**set.discard() to remove selected items if found**

**74-set union and intersections**

**v1.union(v2)**

**v1.intersection(v2)**

**75-comma separated value(csv)**

**import csv**

**then open the file which I save as .csv**

**v=csv.reader(variable which open file.csv)**

**g1=next(v)#give the first line in list**

**76-create function**

**def f1():**

**what I want in function**

**f1()**

**77-create function with parameters**

**def f1(p1,p2,p3)**

**f1(‘p1’,’p2’,’p3’)**

**78- create function with many parameter**

**def f(\*p)**

**f(many of parameter)**

**79-create function has return method**

**when using return method u must put function in variable and then print variable**

**80-import my custom file**

**put some instruction and name as file.py then import file**

**82-excute code from file**

**exec(open(‘file contain code.py’).read())**

**83-excute function code**

**84-recursive function**

**video 84**

**85-pass for empty block**

**def f():**

**pass**

**give me empty block when run**

**86-help and documentation**

**to view how the function work**

**help(function what I want to know more)**

**87-dir function**

**show all operations done**

**print(dir())**

**The dir() function returns all properties and methods of the specified object, without the values.**

**This function will return all the properties and methods, even built-in properties which are default for all object.**

**88-object oriented programming**

**high level language :1-procedual oriented programming,2-object oriented programming**

**89-object oriented programming**

**class&object**

**92-create class and object**

**class (name of class):**

**(name of object)=(name of class)()**

**93-costractor initial in python**

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

* **when initial wrote in class any object create initial will be done when program run**

**94-constractor initial with parameter**

**def \_\_init\_\_(self , ’parameters’):**

**when create object u must enter parameters what u want**

**95-count**

**next():**Python file method **next()** is used when a file is used as an iterator, typically in a loop, the next() method is called repeatedly. This method returns the next input line, or raises *StopIteration* when EOF is hit.

**96-global variable inside class**

**to define variable created out off the class to change this variable in the class we must do that:**

**global (variable name)**

**98-del class**

**def \_\_del\_\_:**

**then delete object by : del object created**

**99-add new attribute in class**

**any attribute I want created automaticly**

**100-secret attribute in class or method**

**write 2 underscore before class name or method name**

**101-hasattr**

**it is importance to know if attribute found in class and the answer will be true or false**

**print(hasattr(object,’any attribute in class’))**

**102- add attribute with setattr**

**setattr(object name,’new attribute’,’ ‘)**

**103-setattr**

**to add attribute in class**

**setattr(object,’attribute name’,’’)**

**103-getattr**

**getattr(object , ‘atttribute’)**

**to get attribute created in class**

**104-delattar**

**delete attribute created not delete attaribute in class**

**delattr(object,’attribute created in class’)**

**105-inner or nested class**

**106-inhertence**

**107-inhertence in python**

**class employee(person)**

**person is class which I want to inheritance from**

**108-multi inheritance in python**

**you can inherte from 2 classes by add , between two classes**

**109-doc built in attribute**

**show documentation of class**

**print((class name).\_\_doc\_\_)**

**110-dictionary built in atterbute**

**print((class name).\_\_dict\_\_)**

**111-name built in attribute**

**print((class name).\_\_name\_\_)**

**print(type(object name.\_\_name\_\_)**

**112-module built in attribute**

**print((class name).\_\_module\_\_)**

**113-base built in attribute**

**print((class name).\_\_base\_\_)**

**base show if class inheritance from any class and if it inheritance from 2 classes base show first inheritance only**

**114-how to get all base attribute**

**import inspect**

**print (inspect.getmro(class which I want)**

**115-bases built in attribute**

**print((class name).\_\_bases\_\_)**

**116- overriding**

**نعيد كتابة نفس الدالة مع كتابة التغيير المراد**

**117- print the name of class**

**print(self.\_\_class\_\_.\_\_name\_\_)**

**118- html in python**

**file.write('<!DOCTYPE html>\n')**

**file.write('<html>\n')**

**file.write(' <head>\n')**

**file.write(' <title>mypage</title>\n')#mypage is the page name**

**file.write(' </head>\n')**

**file.write(' <body>\n')**

**file.write(' <h1>python world</h1>')**

**file.write(' </body>\n')**

**file.write('</html>\n')**

**file.close()**

**119- html class in python**

**make class and make html web**

**120-like 119**

**import webbrowser**

**webbrowser.open\_new\_tab(‘html namet’)**

**121-intro to tkinter**

**122-test of tkinter**

**import tkinter**

**tkinter.\_test()#to make test**

**frm=tkinter.Tk()**

**frm.mainloop()**

**123-change title**

**frm=tkinter.Tk()**

**frm.title(‘new title’)**

**frm.mainloop()**

**124-geometry tkinter**

**frm.geometry(‘500x600’)**

**125-screen width and height with tkinter**

**w=str(frm.winfo\_screenwidth())+’x’+str(frm.winfo\_screenheight())**

**126-location of tkinter**

**import tkinter as tk**

**frm=tk.Tk()**

**frm.geometry('500x600+5+5')**

**frm.mainloop()**

**127-size of geometry**

**frm.update()**

**w=frm.winfo\_width()**

**r=frm.winfo\_height()**

**128-tkinter form in center**

**129-avoid resizable**

**frm.resizable(True,False)**

**True mean u can extend width**

**False mean u can’t extend height**

**130- maxsize and minsize**

**frm.maxsize(اقصى طول , اقصى عرض)**

**frm.minsize(اقل طول تريده,اقل عرض تريده)**

**131-background form**

**frm.config(background=’red’) or**

**frm.config(background=’#ff0000’) #(color in decimal)**

**132-label form in tkinter**

**from tkinter import ttk**

**lbl=ttk.Label(frm اسم الفورم المراد التحزيم فيه ,text=’label’)**

**lbl.pack()**

**133-color label from tkinter**

**lbl1=ttk.Label(frm,text='label1',background='red',foreground='blue')**

**or**

**lbl1.config(background=’red’,foreground=’blue’)**

**# frm.geometry(العرض x الطول )**

**134-font of label**

**lbl1=ttk.Label(frm,text='alpha',background='red',foreground='blue',font='arial')**

**or**

**lbl1.config(font=’arial’)**

**135-font size of label**

**lbl1=ttk.Label(frm,text='header',background='blue',foreground='red',font=('arial',16))**

**or**

**lbl1.config(font=(‘arial’,16))**

**136-padding(dimension) in tkinter**

**lbl1=ttk.Label(frm,text='label 1',background='red',foreground='blue',font=('arial',20),padding=(20,10,10,10))**

**or**

**lbl1.config(padding=(20,10,10,10))**

**padding(شرق,شمال,غرب,جنوب)**

**137-text box entry**

**txt=ttk.Entry(frm)**

**138-style from tkinter**

**s=ttk.Style()**

**s.configure('TLabel',background='red',foreground='blue',font=('arial',20))**

**z=ttk.Label(f,text='enter ur name',style='TLabel')**

**139-button in tkinter**

**btn=ttk.Button(frm,text='click here')**

**140-command to button**

**btn=ttk.Button(frm,text=’click here’,command=اسم الدالة المراد تنفيذها)**

**141-connection between text and button**

**txt1.get() بيجيب البيانات اللى مدخلها فى التيكست**

**142-messagebox from tkinter**

**from tkinter import message box**

**def sum():**

**messagebox.showinfo(‘title name’,’eqal’+ناتج لجمع)**

**143-error message**

**messagebox.showerror(‘title name’,’نوع الخطأ’)**

**144-message box from tkinter**

**145-control from tkinter**

**146-نص افتراضى للتيكست عند بداية البرنامج او نص توضيحى للمطلوب ادخاله**

**svname=tkinter.StringVar()**

**txt=ttk.Entry(frm,variablename=svname**

**svname.set(‘what I want to write’)**

**147-lamda**

**y=lambda x:x print(y(x)) بمعنى ان القيمتتين يساوو بعض**

**y=lambda x:’hi’**

**y=’hi’**

**148-combobox in pythonقائمة منسدلة**

**cbx=ttk.Combobox(frm,values=(‘’,’’,’’)الاشياء المراد ادخالها,state=’readonly’عشان عدم السماح بالكتابة)**

**cbx.current(index no. of value) بمعنى قيمة افتراضية بتكون موجودة**

**149-list box in python**

**lbx=tkinter.Listbox(frm)**

**lbx.insert(index no.,’what I want to insrt’)**

**lbx.get(ACTIVE)>>>>>>>>>>>>>> to view which I select**

**150-radio button in tkinter**

**v=IntVar() #لو دخلت القيم 0و1**

**v=StringVar()# لو دخلت القيم اسماء**

**v.set(0)**

**rdbm=tkinter.Radiobutton(frm,text='male',value='male',variable=v)**

**rdbf=tkinter.Radiobutton(frm,text='female',value='female',variable=v)**

**def f():**

**print(v.get())**

**btn=tkinter.Button(frm,command=f)**

**151-check button in tkinter**

**v=IntVar()>>>>>>>>>>>>>>> answer 0 when not checked or 1 for checked**

**#v=StringVar()>>>>>>>>>>> answer 0for not checked or 1 for checked**

**#v.set(0)**

**v=BooleanVar()>>>>>>>>>>> answer True for checked or False for not checked**

**cb=tkinter.Checkbutton(frm,text='agree',variable=v)**

**cb.pack()**

**152-show image on tkinter**

**canvas=tkinter.Canvas(frm,width=500,height=500)**

**canvas.pack()**

**img=tkinter.PhotoImage(file='giphy.gif')**

**canvas.create\_image(0,0,image=img,anchor=NW)**

**عشان اخلىه يملا الشاشة كلها**

**canvas.pack(fill=tkinter.BOTH,expand=1)**

The **fill** option tells the manager that the widget wants fill the entire space assigned to it. The value controls how to fill the space; **BOTH** means that the widget should expand both horizontally and vertically, **X** means that it should expand only horizontally, and **Y** means that it should expand only vertically.

The **expand** option tells the manager to assign additional space to the widget box. If the parent widget is made larger than necessary to hold all packed widgets, any exceeding space will be distributed among all widgets that have the **expand** option set to a non-zero value.

**canvas.create\_line(x1,y1,x2,y2,fill=’red’)**

**153,154-grid system in tkinter**

**lbl.grid(row=0,column=0)**

**row كناية عن الصف وبيبدأ بالصف 0**

**column كناية عن العمود وبيبدأ بالعمود 0**

**155-frame grid system**

**ملخصها ان ممكن اعمل اكثر من فريم وكل فريم له جريد خاصة به اى له تنسيق مخصوص**

**frame=tkinter.Frame(frm)**

**اى حاجة عايز احزمهها بحزمها فى الفريم الجديد\**

**frame=(row=2,column=0,columnspan=3)**

**column span بيعتمد على عدد item اللى قبله**

**156,157,158,159-calc with tkinter**

**عشان ابدل الحاجة اللى مكتوبة فى label**

**label[‘text’]=(what I want)**

**padx uses to get space in x dimension**

**pady uses to get space in y dimension**

**160-170-employee data entry**

**to exit from app we put command=frm.destroyed**

**to be sure that the field isnot full by space**

**txt1.get().strip()==’’**

**171-change icon picture**

**frm.iconbitmap(‘place of icon’)**

**when I want to open program directly extension will be .pyw**

**172-import class tools**

**173-hide and show item**

**to hide label : lbl.pack\_forget()**

**to show label : lbl.pack()**

**174-get all controls in tkinter**

**show all things such as label,button,…….. in tkinter**

**controls=frm.winfo\_childern**

**175-difference between ttk and tkinter**

**label[‘text’]=’hello’ change text settings in label by hello**

**176,177- background for all items in tkinter**

**178,179-font for all items in tkinter**

**180,181- foreground for all items in tkinter**

**182,183,184-messagebox**

**185-show dialog tkinter**

**بمعنى يظهر فريم ولا يعمل شئ جديد الا عندما يتم الغاء الفريم الجديد**

**frm3=tkinter.Toplevel()**

**frm.grab\_set()**

**186-key press enter**

**بمعنى يعمل المطلوب حتى لو التيكست فاضى**

**txt.bind(‘<Return>’,lambda x:test())**

**187-generate input box**

**wait\_window :** seems to not return until the given widget passed as parameter is not destroyed.

**txt.focus when I want cursor to be in the field I want to write in it**

**when I want to change style of each button make main.TButton**

**190-key press entry**

**when I click word show all information about this word**

**def key(event):**

**print(event)**

**txt1.bind(<Key>,key)**

**191-accept numbers only**

<http://www.tcl.tk/man/tcl8.5/TkCmd/entry.htm#M-validate>

**def validate\_number(text):  
 if str.isdigit(text): return True  
 elif text == '':return True  
 else:return False  
reg\_text=frm.register(validate\_number)  
txt4.configure(justify='center',validate='key',validatecommand=(reg\_text,'%P'))**

**validate=’key’--------mean I validate when I tab key on keyboard**

**193-new easy form**

**make form in file and import the file**

**194-easy addition from tkinter**

**import buttons from file and then make command as**

**num = input(‘number 1’,True)**

**195,196-generate easy button**

**make button in the file and then import button from file created and it is possible to make command in the file**

**197-generate easy label**

**make label in the file then import it**

**198,199-generate text box**

**make text box in the file and then import it**

**200-generate easy variable from tkinter**

**make StringVar def() in file and then import it**

**in textbox=txt.config(textvariable=stringvar)**

**201-generate easy radio button**

**make radio button def() in file and then import it**

**202-generate easy checkbox**

**make checkbox def() in file and then import it**

**203,204-easy radiobutton and checkbox**

**205-generate textbox**

**206-generate easy combobox**

**make combobox def() in file and import it**

**207-generate easy style**

**208-generate easy style**

**209-generate easy listbox**

**make listbox def() in file and then import it**

**210,211-generate easy frame**

**make frame def() file and import it**

**212-generate title**

**213-generate easy toplevel**

**214,220-screen forums**

**221-menubar in tkinter**

**menu=tkinter.Menu(frm) علشان اعمل منيو فى التيكنتر  
filemenu=tkinter.Menu(menu,tearoff=0) بعرفه ان ديه قائمة منسدلة من المنيو الاساسى وبعل التيراوف=0 علشان ألغى الفواصل اللى فوق فى المنيو   
filemenu.add\_command(label='new',command=lambda :print('new')) بضيف ااوبشنات فى القائمة المنسدلة  
filemenu.add\_separator()بضيف خط فاصل بين الاوبشنات  
filemenu.add\_command(label='save',command=lambda :print('save'))  
menu.add\_cascade(label='file',menu=filemenu) بحزم القائمة المنسدلة فى القائمة الرئيسية  
frm.config(menu=menu)**

**233-connect with my sql windows**

**import mysql.connector  
conn=mysql.connector.connect(user='root',passwd='')  
print(conn)**

**mysql.connector to be sure that the program is connected with database**

**mysql.connector.Error-----function contain error**

**236-create user account on sql server**

**237-creat database with python**

**mycursor=conn.cursor()**  
**mycursor.execute('CREATE DATABASE db\_with\_python')**

**the name of database mustnot have spaces**

**238-create database support Arabic language**

**cur.execute('CREATE DATABASE mycompany DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8\_general\_ci')**

**239-show all my sql databases**

**cur.execute('SHOW DATABASES')**

**240-create table**

**cur.execute('CREATE TABLE employee(empno int primary key,empname varchar(80))')**

**241-insert data into table**

**cur.execute("insert into emp values(1,'mina')")**

**242-update data from table**

**cur.execute("update emp set empname='k' where empno=1")**

**243-delete data from table**

**cur.excute(“delete from emp where empno=1”)**

**244-select data from table**

**cur.excute(“SELECT \* FROM emp”)**

**245-select data from table**

**x=cur.fetchone()---------------result type is tuble contain one column**

**x2=cur.fetchall()---------------result type is list contain tubles and u can use more times than one**

**246-stored procedure in python**

**store procedure**

<https://www.youtube.com/watch?v=bVS4vWE6Fgw>

**247-call procedure**

**248-create Arabic user with stored procedure in python**

**249-create mysql db tools**

**make file and import \***

**250-mysql db tools**

**to check if the item available or not**

**if ‘what I want to check’ in locals():---------------if I want to check in def()**

**if ‘what I want to check’ in globals():---------------if I want to check in al program**

**251-get data from table in mysql db tools**

**252-MAX(),MIN() mysql db tools**

**The SQL MIN() and MAX() Functions**

**The MIN() function returns the smallest value of the selected column.**

**The MAX() function returns the largest value of the selected column.**

**253-mysql db tools**

**254-connect between database and python**

**255- connect between database and python(2)**

**when design table or database we do**

**create table/database IF NOT EXISTS**

**if not exists to prevent error when the program run and the table was created**

**in the program I created is\_created equal none**

**256- connect between database and python(3)**

**257- connect between database and python(3)**

**258- connect between database and python(4)**

**259- connect between database and python(5)**

**contain how to make password show by \* and how to make the text read only**

**to make text read only**

**txt.config(state=’readonly’)**

**to write password**

**txt.config(show=’\*’)**

**260- connect between database and python(6)**

**261- connect between database and python(7)**

**262- connect between database and python(8)**

**263- connect between database and python(9)**

**264- connect between database and python(10)**

**emp\_var.set(dbautonum(‘employee’,’employeeid’)**

**emp\_var---------string variable in text box,dbautonum------------function or def contain table,employee-----------name of table,employeeid---------column which I want to make relation with emp\_var**

**265- connect between database and python(11)**

**add button**

**is\_add=dbrun ”insert into employee values(%d,’%s’,’%s”,%d) %(empno.get(),empname.get(),address.get(),salary.get())**

**if is\_add:print (‘added’)**

**dbrun ---------function or def to make insert to table,then get values entered with using string var**

**or**

**is\_add=dbrun(insert into employee from values(“+empno.get()+”,”+empname.get()+”+”+address.get()+”+salary.get()+”)**

**266- connect between database and python(12)**

**clear button**

**empno.set(dbautonum(‘employee’,’employee\_id’)**

**empname.set(‘’)**

**address.set(‘’)**

**salary.set(‘’)**

**empnametext.focus()**

**dbautonum ---------it is function is count how many people added in table**

**267- connect between database and python(13)**

**find button**

**enum=input(‘enter number:’,True)**

**row=dbget(‘select \* from emp where emp\_id like ”%’+enum+’%”)[0]**

**empno\_var.get(row[0])**

**empname\_var.get(row[1])**

**salary\_var.get(row[2])**

**address\_var.get(row[3])**

**enum----input function to enter empno,row--------------get list of tuple of selected empno then add [0] to get in tuple without list ,empno\_var ---------------get the first value in tuple**

**268- connect between database and python(14)**

**make button disable or enable**

**frm.winfo\_children()[no of element I want enable or disable].config(state=’enable or disable’)**

**269- connect between database and python(15)**

**edit button**

**is\_edit=dbrun(“update employee set employeename’”+empname\_var.get()+”’,address ‘”address\_var.gett()+”’,salary ”+str(empsalary\_var.get)+”where employee\_id =”+enum+”**

**if is\_edit: print(‘edited’)**

**employeenas,address,salary------------column in employee table**

**270- connect between database and python(16)**

**improve in find button**

**271- connect between database and python(17)**

**delete button**

**if msgask(‘do u want to delete’):**

**is\_del=dbrun(‘delete from employee where employee\_id=”+empno\_var.get())**

**if is\_del: print(‘deleted’)**

**272- connect between database and python(17)**

**check valdition**

**if empname\_var.get().strip()==’’:**

**return False**

**273-**