

And the second s	find out		V Cash Control of the Control	
DA = X	9n + 9B	ti uh da	15 19	4.7
DB = (	1 P A (			
QX=E	opx+2	EPPE OF	, with the same	70-5
			1) You	
stepa s	tate table.			
			inter carry	
P.S		73/		
DA GB	X QA	QB 19	· ili	
00	0 0	0 1	16 (00)	
0 - 1		0		ofo
ŏ i	0	10	(1)	(0)
10		7 0	Colp. 1.10	
	00	0/1	10111	nore . Se
0	0.			
Digital	elk design.		12 / 19 /	
		i contrain	VIVALUE OF	
· digital	clk is	a type	of Ok +1	not display
the at -	ime digital	y 26	100 14	
		1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1 Warden	
· digital	clks or	e of ten	associate d	with
election	nic device	The de	gital de	scription
refers	only the	Hisplay	not the	40 me
mechani	sm: EH	set, (1 49.) 1		96311
Recou	le dipital	alk car	, ile	6 05 0
			16	Manage
- M 1217	pires device			

	Poblary Sushmita
	Python programing: Popiary Sushmita  HALI become - 368.
	4 图形的基础的形式,是是一种形式的一种形式的一种形式的一种形式的一种形式的一种形式的一种形式的一种形式的
	Object ovented programing
	Dops the way to organise the code
	Dops the way to organise
<u> </u>	Backena
	import Splite3
	clas Database
	definit-(self;
	(onn= Splite. conect (books. olb)
	CIVE COMM. CUNCECCO
	Cur. execute ("CREATE TABLE")
	Cam-Commit ()
	conn, close ()
•	front end:
	from thinter import in the little
	from tkinter import Database
	from tkinter import Database Database Database Database )
	from backend import Database database continued in port
	from backend import Database ()  database = Database ()  det get - selected - you (event):
	from backend import Database database = Database ()  det get_selected_row (event):  global selected_liple
	from backend import Database ()  database = Database ()  det get _ selected _ row (event);  global selected _ liple  index = ligt 1. (urreselection () [0]
	from backend import Database ()  database = Database ()  det get _ selected _ row (event):  global selected _ liple  index = list 1. (urreselection()[0]  selected_lipte = list 1.get (index)
	from backend import Database ()  database = Database ()  det get _ selected _ vow (event):  global selected _ liple  index = list 1. (urreselection () [o]  selected_lipte = list 1.get (index)  e 1. delete (o. END)
	from backend import Database ()  database = Database ()  det get _ selected _ row (event):  global selected _ liple  index = list 1. (urreselection()[0]  selected_lipte = list 1.get (index)
	from backend import Database  database = Database 1)  det get _ selected _ row (event):  global selected _ liple  index = list 1. (urreselection()(o)  selected_lipte = list 1.get (index)  e1. delete (o. END)  e1. insert (END. selected truple(1))
*	from backend import Database  database = Database ()  det get _ selected _ row (event);  global selected _ liple  index = list 1. (urreselection()[o]  selected_lipte = list 1.get (index)  e1. delete (o. END)  e1. insert (END. selected tuple(i))  Inheritence: Is the process of crease of
*	from backend import Database  database = Database ()  det get _ selected _ row (event);  global selected _ liple  index = list 1. (urreselection()[o]  selected_lipte = list 1.get (index)  e1. delete (o. END)  e1. insert (END. selected tuple(i))  Inheritence: Is the process of crease of
12	from backend import Database  database = Database ()  det get _ selected _ row (event):  global selected _ liple  index = list 1 (urreselection()(o)  selected lipte = list 1 get (index)  e1. delete (o. END)  e1. insert (END. selected truple(1))  Inheritence: Is the process of create a  ew class from a base class.
12	from backend import Database  database = Database ()  det get _ selected _ row (event);  global selected _ liple  index = list 1. (urreselection()[o]  selected_lipte = list 1.get (index)  e1. delete (o. END)  e1. insert (END. selected tuple(i))  Inheritence: Is the process of crease of
7	from backend import Database  database = Database ()  det get _ selected _ row (event):  global selected _ liple  index = list 1 (urreselection()(o)  selected lipte = list 1 get (index)  e1. delete (o. END)  e1. insert (END. selected truple(1))  Inheritence: Is the process of create a  ew class from a base class.

def init - (self, félépath): self, dilepath = félépath with open (filepath, '71) as file: self. balance = int (file. read (1) det without (self. amount): selt. balance = self. balance - amount det de posite (selt. amount): selt. balance = selt. balance + amount. det commit (self): with open (self. filepath, 'w') as file:
file write (str (self. balance)) dass deching (account) -> desired dass
det\_init\_ (self) fieleth filepath, fee): Account\_init\_ (self, file pathe) selt-fee = fee des transer (self amount): selt. balance = self. bakance= amount = self checking = checking ("account 11 balance. +x+") checking, transfer (100) print (checking balance) checking, commit().