	Graphs in Ryshon
χ.	Graphs in Python
	It is a Non-linear data structure. It does not non in sequence
	not non in sequence
	8 4 5
5	2 > Vouhices
	Edges
	3 Edges
	impout print
10	class Vertra:
	def-juit- (self, data) -> None:
	set sey, data = data
	Self. connections 2 [1]
that who is	who have the same
15	def add_edge (self, obj):
	def add_edge (self, obj): self. connections. append (obj)
	- 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
	class Folge:
	alf - lut - (self).
20	self: connections = []
	100 100 100 100 100 100 100 100 100 100
	det add-edge (self, from-venten, to-verten)
	Self. connection. append (prom. ventra. data)
05	self. connection, append (fine - venture data)
25	tlas Graph: (001) 2504 - 3160
. 10	
	self-graph 2 }}
30	def adel verbier (self, obj)
	self.graph.update (& obj. Data 9: Eobj. comeria)
-	V1 = Vertex (4,4)
	V\$ 2 Verban (n2h)
	V) 2 Verchex (ugu)
35	VI = Velicha ("4")
	VI - WMVI (7)
11	

			Camlin Page			
		Date	1	1		
,			,			
	012 80001)					
	el 2 Edge () el. add-edge (VI, V2)			1	_	
				<u> </u>	_	
5	er= Edgel)				_	
	22. add-edge (VI, V3)			<u>)</u>	_	
					_	
	e3 = Edge ()					
	e3 · adul-age (v3, v4)			9		
10	04 2 Flac ()					
	e4. adl-edje (V2, V3)					
	er unitige (_	
	e52 Edge()					
15	es.ad-edge (V4,V1)			<u>lut</u>	_	
	1/2.4				_	
	VI. add-edge (&1. connections) VI. add-edges (&2. commentum)					
	VI. adel_edges (2. compresse)					
	V2. add-edge (e3. connutrin)					
20	VZ. water and y					
	V3. add-edge (e4. connetton)				_	
				-	_	
	V4. add-edge (e S. connection)			1	_	
25				4	_	
	9:12 yeaph ()			ă		
					_	

91. add - vertices (VI)
91. add - vertices (V2)
91. ad - vertices (V3)
92. add - vertices (V3) pprint. pprint (g1-graph)