	LLO1
-	
	Chapter 9 - Structures
_	laser with me del ships till hand and " was
	Arrays and strings => Similar data (int, floot, cher)
_	a constant and the state of the
1	Structures can hold => dissimilar data
	The state of the s
	Syntax for creating Structures A C Structure can be created as follows:
	1) C Gleve time can be used to follows:
	All al year blooming the idea westered an
	struct employee &
	int code; => This declares a new
	int code; => This declares a new float salary; user defined data - type!
	float salary .
4	char name (10];
11	semicolon is important
	3cm/colon 15 important
	We can use this user defined data type as follows:
	We wan use this user metica wind type as follows.
	Struct employee e1; => creating a structure variable
	State complete C1; — Clarity a state warm
	Stropy (e1. name, "Harry");
	e1. code = 100;
_	e1. Salary = 71.22;
_	6- A clevalue in C is A calledian at variables
_	So a structure in C is a collection of variables of different types under a single name
_	of sufferent types where he single name
_	O. b. O Islaile a program to a love the I taile
_	Quick Quiz: Write a program to store the details
_	of 3 employees from user defined data. Use the structure declared above.
_	
	The street life to the street of a street of the street of
_	

structures?
Structures? Create the data types in the employee separately but when the number of in a structure increases, it becomes for us to create data variables without s. In a nut shell:
separately but when the number of
in a skucture increases it become
Par 111 to Create data Variables 1111
To mut chill:
s. In a run spen
C 17 17 MILE TO THE REAL OF THE STATE OF THE
make data mangement easy for the
make wata mangement thay for the
er. Employer trucks
Skructures
e on array of integers, an array of floats
n array of characters, we can create
e on array of integers, an array of floats in array of characters, we can create ray of structures.
nployce facebook [100]: => An array of
Structures
Lastruck Comployer Cold
access the data using
? [0]. Code = 100;
k[1]. Code = 10 1; 1991 = wald 19
000 & 50 on
I So a structure in C is a scalart
Structures (an also be initialized as fo) lows:
(an also be initialized as follows:
THAT AND
mployce harry = \(\frac{1}{2} \) 100, 71.22, "Harry" \(\frac{3}{2} \);
TARLES V. S. I. M. B. S. M.
employee shubb = {0}; => All clements set to 0

	Structures in memory
	Structures are stored in contiguous memory locations For the structure e1 of type struct employee, memory layout looks like this:
	For the structure en al libe struct employee memory
	layout looks like this.
74	You stow (Since employed): are further for
	100 (71.22 Haven)
	Address > 78810 78814 78818
	In an array of structures, these employee instances are stored adjacent to each other.
3	are stored adjacent to each other.
	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1	Pointer to structures
	Pointer to structures can be created as follows:
	t volema) Lines
	Struct employee * ptr;
	ptr= &ei
	· · · · · · · · · · · · · · · · · · ·
	Now we can print structure elements using:
	Look to Charley 5
	printf ("% d" *(ptr) (ode);
\perp	(*ptr).code
	Arrow Operator
	Instead of writing * (ptr). Code we can use arrow
	operator to access + structure properties as follows
	* (ptr)·lade Or ptr -> Code
	(*ptr).code
	Here -> is known as the arrow operator.

	6043	
/	Passing Skycture to a function	
-ure	Passing Structure to a function A structure can be passed to a function like any other data type.	fust
/	Void Show (Struct employee e); => Function	prototype
	Quick Quiz: Complete this show function to the Content of employee.	display
	Typedef keyword We can use the typedef keyword to commonly used with structions Expedef is more commonly used with structions	clate C. ures.
	Struct complex & Struct complex C_1, C float ing; for defining complex 3;	numbers
	typedef Struct Complex & floor real; float iny; => Complex No C1, C2	;
	3 Complex No: for Jefining Comple	number
	tere -> 15 known at the award of the rates.	
-		~