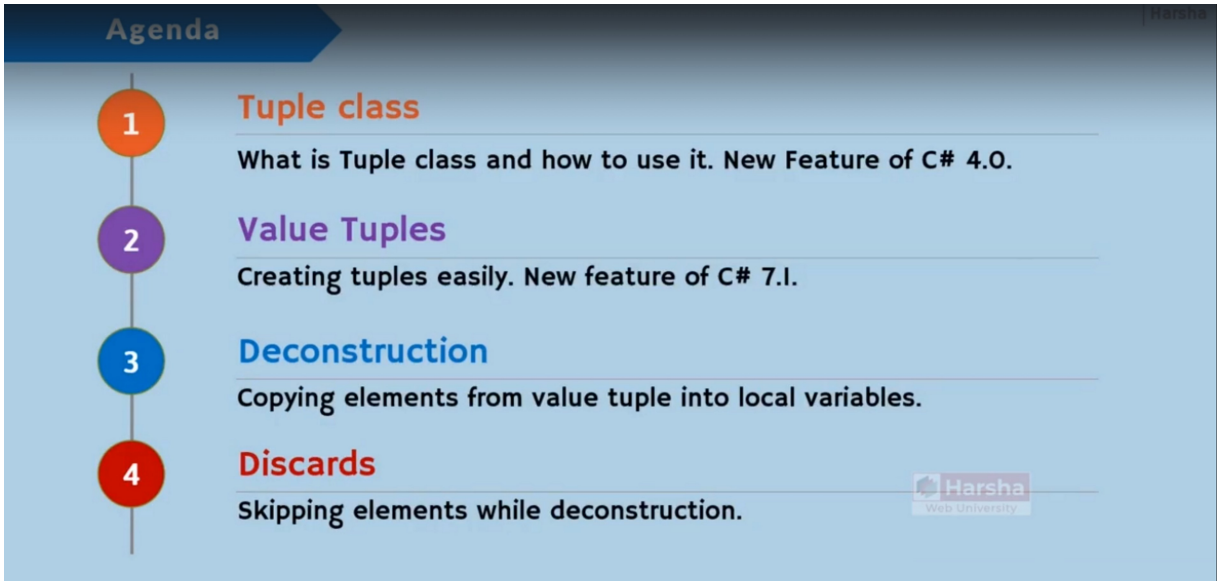
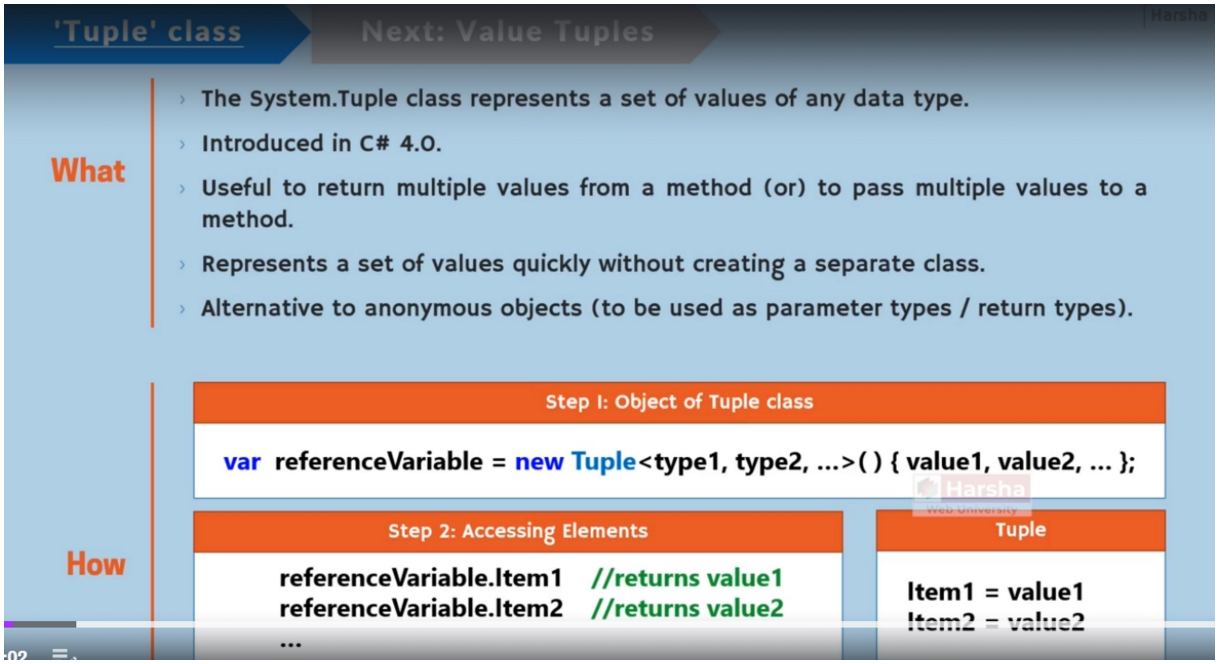
Section 24: Tuples

28 July 2024 18:21



The Tuple is an object that can contain multiple values at the same time
so it is an alternative to anonymous object but the benefit is that you can use the tuples as a parameter type or return type.



From a method you want to return multiple values in that cases one option is that you can
create a separate class with the specified properties and specified that class as the written
type of the method for example there is a method called get person details and from this method you want to return
person name and age so the option number one is that create a separate class called person details with two properties that is person name and age and for this get person details method specify the written type as person details and inside the method you create and return an object of person details class this is one option to return multiple values but it's a multi-step process and a bit lengthy process right.

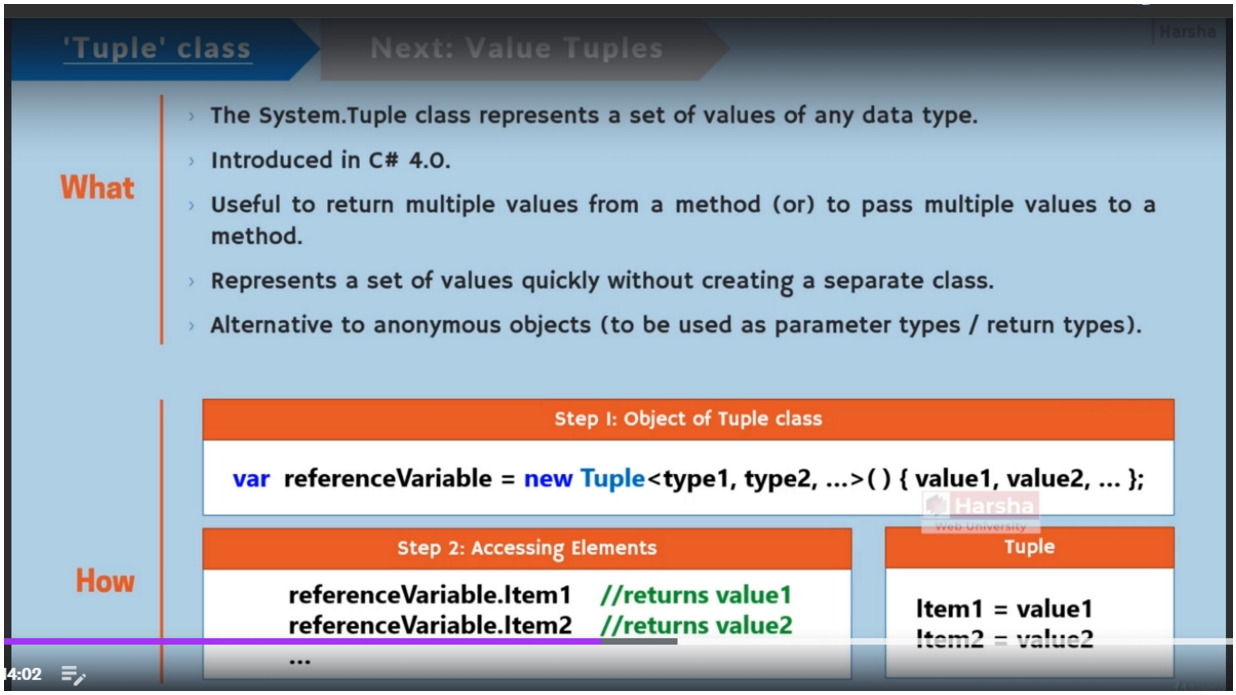
step one you should create a separate class called person details and specific that class as the written
type of the particular method and you should make a separate object for the person details class
and return that object from the get person details method so it's a lengthy process but quickly you want to return an object
that contains multiple values for example it should include person name and age here.

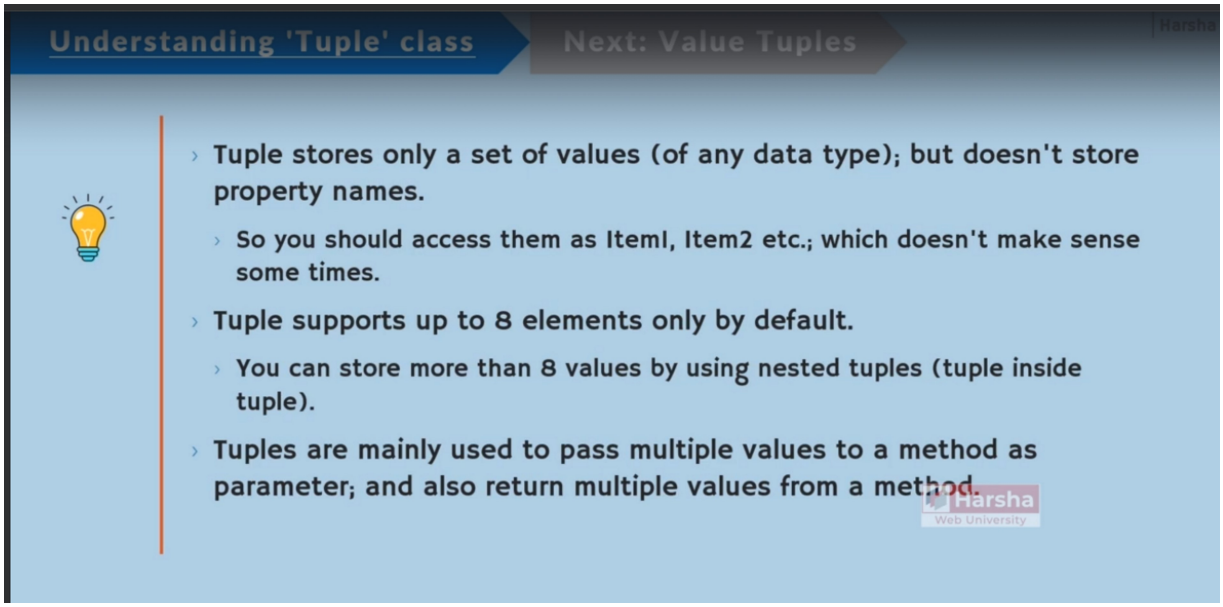
second option is that we can return an anonymous object and specify the return type as object that means system.object class
but the problem is after receiving the value in the calling portion it will be difficult to identify what

properties are exist in the object because the written type is system.object that is why it is not recommended to
written anonymous objects from any method you can use anonymous objects within the same method
but not across the methods that means should not be used as method parameter or written type of the particular method.

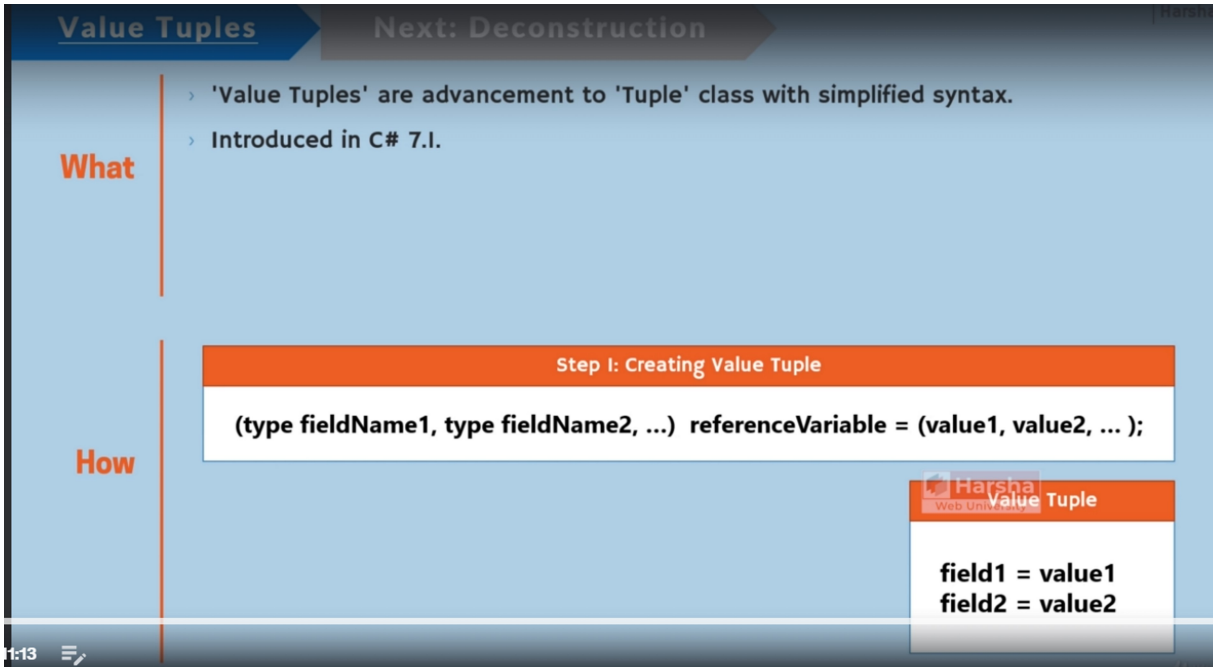
the third option is that you can use out keyword.

Fourth option, use tuple class



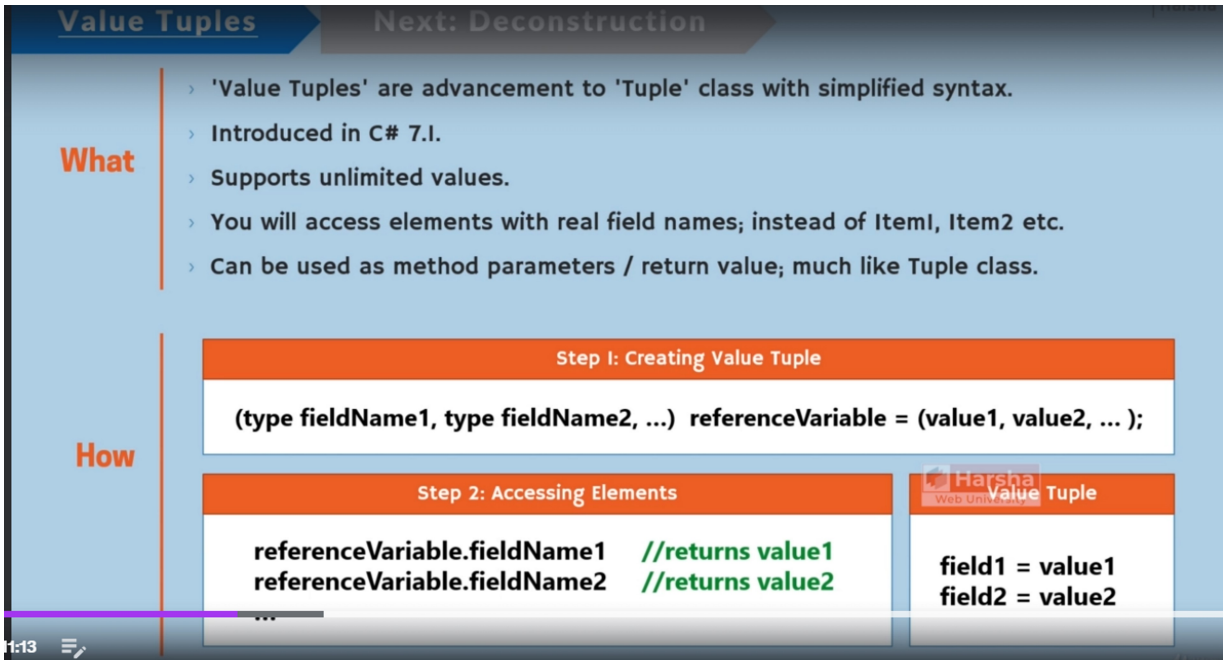


Value Tuples



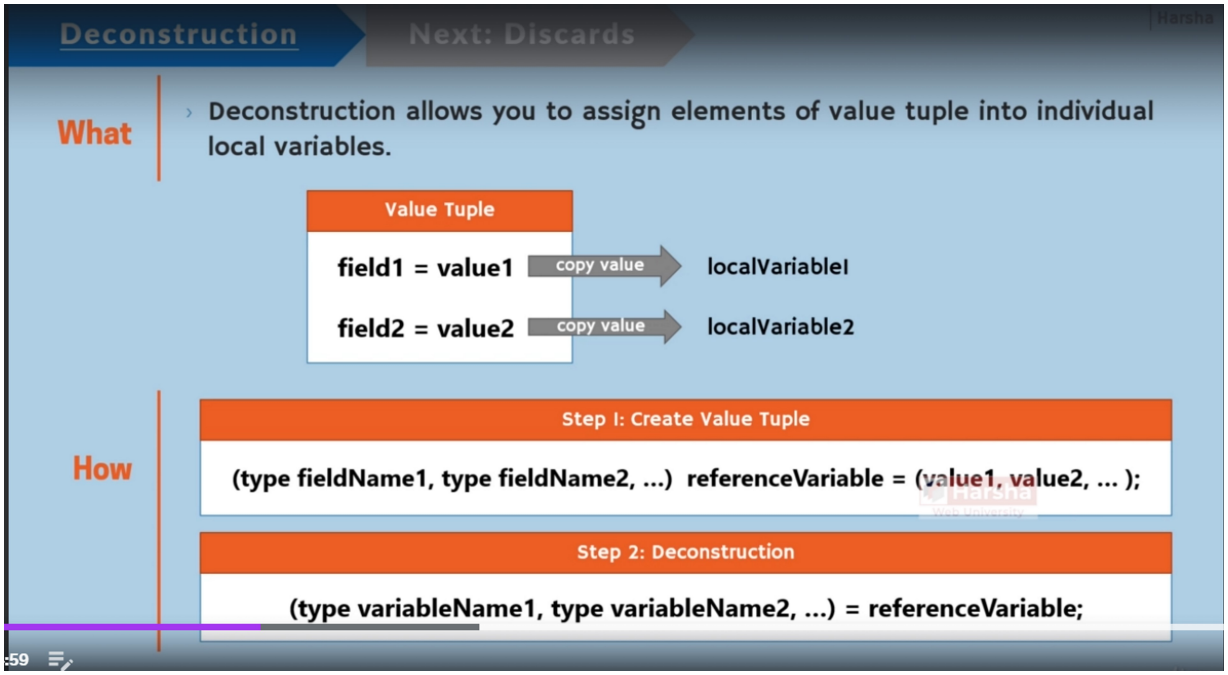
So what is the big complaint about the Tuple class? The process of creating the Tuple class's object and initializing the values into that object is a bit lengthy syntax.
And also each value is represented as item1, item2, item3 etc., properties; instead of representing actual field names such as customerName or phoneNumber. These two

problems are overcomed in C# 7.1 by introducing value tuples.

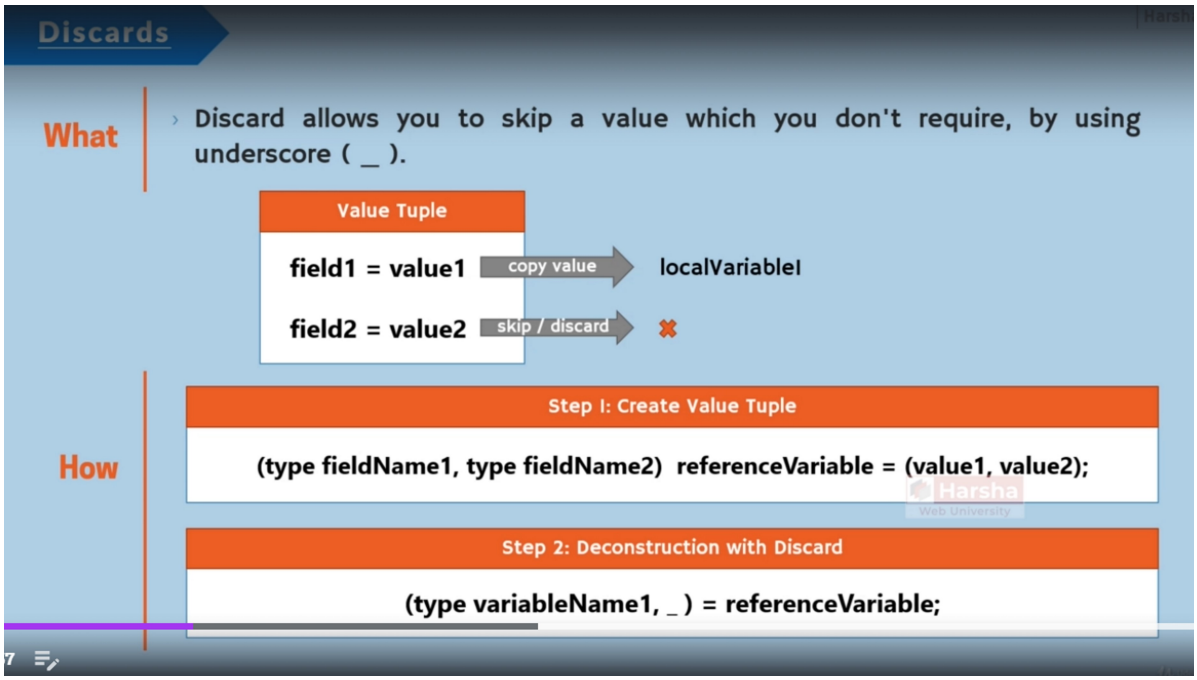


Value tuple is the best for returning multiple values at a time from a method.

Desconstruction


Deconstruction
Next: Discards
What
Deconstruction allows you to assign elements of value tuple into individual local variables.
Value Tuple
field1 = value1
copy value
localVariable1
field2 = value2
copy value
localVariable2
How
Step 1: Create Value Tuple
(type fieldName1, type fieldName2, ...) referenceVariable = (value1, value2, ...);
Step 2: Deconstruction
(type variableName1, type variableName2, ...) = referenceVariable;

Discards


Discards
What
Discard allows you to skip a value which you don't require, by using underscore (_).
Value Tuple
field1 = value1
copy value
localVariable1
field2 = value2
skip / discard
How
Step 1: Create Value Tuple
(type fieldName1, type fieldName2) referenceVariable = (value1, value2);
Step 2: Deconstruction with Discard
(type variableName1, _) = referenceVariable;