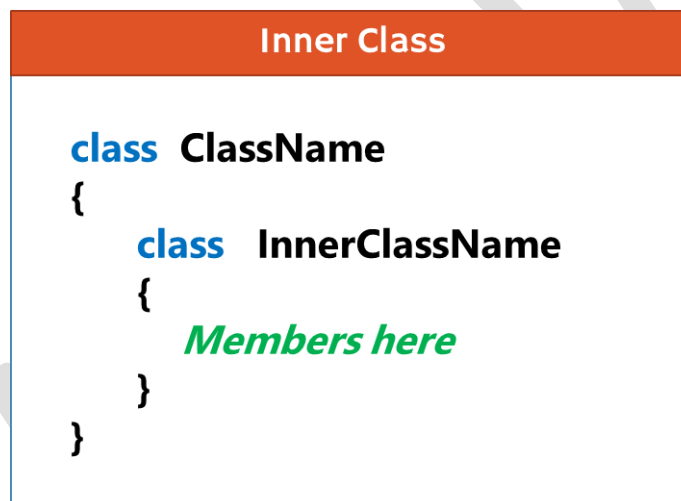


31

Inner Classes

Introducing Inner Classes

- › "Inner Class" (a.k.a. Nested Class) is a class, which is created in another class (outer-class or containing-class).



The diagram shows a code block titled "Inner Class" with an orange header. Inside the block, the C# syntax for a nested class is shown: an outer class named "ClassName" containing an inner class named "InnerClassName". The inner class contains the text "*Members here*".

```
class ClassName
{
    class InnerClassName
    {
        Members here
    }
}
```

- › We can create all inter-related classes of a class, "inner classes".

Syntax to access inner classes:

- › OuterClassName.InnerClassName
- › By default, inner class is "private"; so it is accessible within the same outer class.
 - › To make it available to outside of the outer class, you can use other access modifiers such as "protected", "private protected", "internal", "protected internal" or "public".
- › A nested class can be declared as a private (default), public, protected, internal, protected internal, or private protected.

- › Outer class can't access the members of inner class directly, without object.
- › Inner class can't access the members of outer class directly, without object.
- › You are allowed to create objects of inner class in outer class; and vice versa; but you can't do both; if you create objects vice-versa, it causes StackOverflowException.
- › You can create a child class for the inner class, outside the outer class.

HARSHA