Asynchronous Programming:

Performing different tasks simultaneously/parallely instead of sequentially is called as asynchronous.

Ex:

Let say we have a repository class interacting with a database class which is an expensive task, so if one task is interacting with database and you want other task to interact as well.

Here you can make an asynchronous call from repository to database so that other task do not have to wait for the completion.

**Syntax Ex:**

*Task<Model> MethodNameAsync(Attributes)*

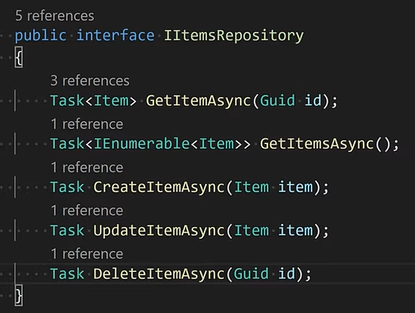
***Task<Item> GetItemAsync(Guid Id)***

**Async**: Indicates that it’s an asynchronous method.

**Task**: Indicates that it’s an asynchronous operation, so you will get the result as task that represents an asynchronous operation that eventually is going to return items whenever we have finished retrieving from the database.

**Await**: When the asynchronous operation completes, the await operator returns the result.

Repository > IItemsRepository:



Repository > DatabaseRepository:

