

O que aprendemos?

Nessa aula você:

- vivenciou alguma das diferenças entre UI Tests e Unit Tests: a necessidade da aplicação estar publicada antes do teste ser executado, o tempo de execução maior nos testes de UI
- continuou utilizando como nomenclatura dos testes as três seções do triplo AAA
- verificou que cada teste abre um navegador para sua execução, e que esse recurso pode ser melhor gerenciado aprendeu que a prática de isolar o código de inicialização dos testes é denominada **Setup** e que no xUnit isso é feito via construtor
- também aprendeu que a prática complementar de liberação dos recursos de um teste é chamada de **TearDown** e no xUnit é feito através da implementação da interface `IDisposable` pela classe de teste
- verificou que o xUnit cria uma instância da classe de teste para cada teste a ser executado, portanto a criação do WebDriver no construtor não soluciona o problema de gerenciamento
- por fim conheceu as duas estratégias do xUnit para compartilhamento de recursos entre testes: **IClassFixture**, que compartilha o recurso entre os testes da **mesma classe**, e **ICollectionFixture**, que compartilha por **várias classes**