

XUnit Style Setup and Teardown

In this lecture I'm going to give you an overview of PyTest's support for the XUnit style of setup and teardown functions.

XUnit Style Setup and Teardown

```
def setup_module():  
def teardown_module():  
def setup_function():  
def teardown_function():  
def setup_class():  
def teardown_class():  
def setup_method():  
def teardown_method():
```

- XUnit style setup/teardown functions will execute code before and after:

- Test Modules
- Test Functions
- Test Classes
- Test Methods in Test Classes

- One key feature of all unit test frameworks is providing the ability to execute setup code before and after the tests. Pytest provides this capability with both XUnit style setup/teardown functions and with Pytest fixtures.
- The XUnit style setup and teardown functions allow you to execute code before and after: Test modules <click>, Test Functions <click>, Test Classes <click>, and Test Methods in Test Classes.
- Using these setup and teardown functions can help reduce code duplication by letting you specify setup and teardown code once at each of the different levels as necessary rather than repeating the code in each individual unit test. This can help keep your code clean and manageable.
- Lets look at some examples to see how this works.