**Pytest:**

Pytest is a testing framework you to make and write test simple and even it is use for to do selenium testing. writing the script is not just about testing user actions but also getting a certain value from it like when we perform test for verifying with certain values etc.

Pytest is a framework that makes building simple and scalable test easy.

We have to use **pip install pytest**  to install the package

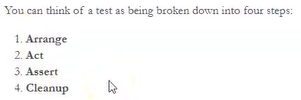


# file name can be start with or end with test where as method should start with test and to execute enter pytest  
# in terminal then all the test method with all test files will execute  
# to execute particular file use pytest filename.py in terminal  
# pytest -v where v is verbose will give o/p with fuctions with status like passed or failed  
# pytest -q gives just how many test case passed and sec for executing  
# pytest -h for help to get all commands  
# pytest -v -s , -v will give all the details filename, method name, status and s will give print statements  
# pytest -vsk keyword, will give same as above but only method name have k keyword

**Markers:**

# in pytest there is markers concept where we can do custom marking and also we have built in markers  
# these markers are used as decorators on function which helps to execute certain set of function which  
# having decorator with a same expression  
# we can use pytest -m expresion where expression is in decorator like ex: @pytest.mark.sanity  
# so we use pytest -m sanity  
# pytest --markers will give all builtin markers like skip and xfail  
# we use like @pytest.mark.skip will skip the execuion of this method when you use pytest to execute  
# we use like @pytest.mark.xfail will not execute but not show error as test failed. if the fuctionality of  
# ay function is not completed at that point we use this xfail

**Fixtures:**



So arrage is like open browser , login and cleanup is like logoff, closing browser

Act is like behaviour of application and asser is checking the condition

So for handling arrange and cleanup we use fixture where we want to execute for all methods written once using in setup method with decorator @pytest.fixture like

@pytest.fixture()  
def setUp():  
 print("open browser")  
 print("login")  
 print("browse product")  
 yield  
 print("logoff")  
 print("close browser")  
 # here before yield code will execute before every function and at last the remaining code will get execute  
 # to execute like that we have inherit this method to another where we want to use