# What is Framework:

* Provides QA professionals with different blends of libraries
* Set of guidelines,
* Coding standards

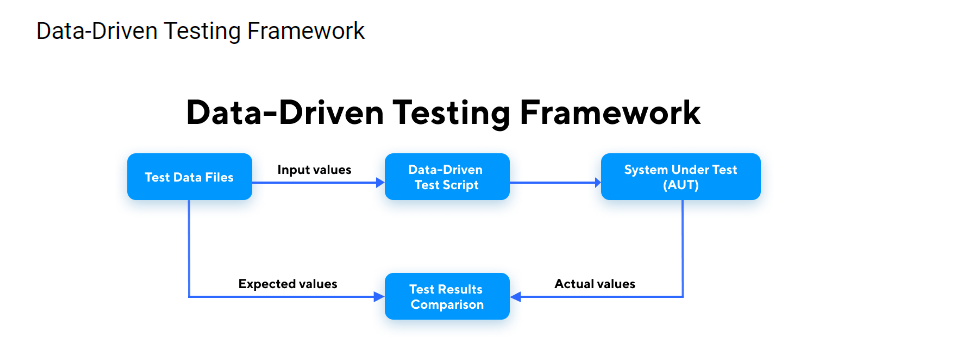
# Whys is framework required 🡪Purpose:

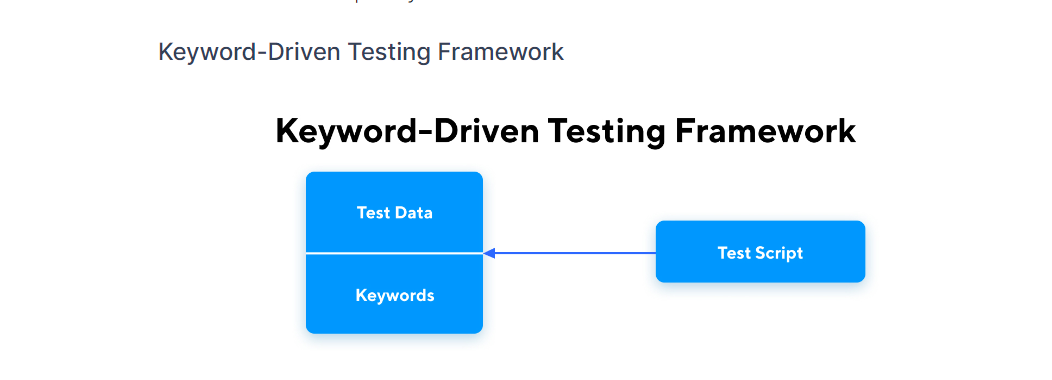
1. Maintenance : Keeping code changes up to date would be easy with structured artifacts
2. Reusability: Reusable functions and modules across different functions
3. Scalability : Tests need to be scaled up and down depending on demand.

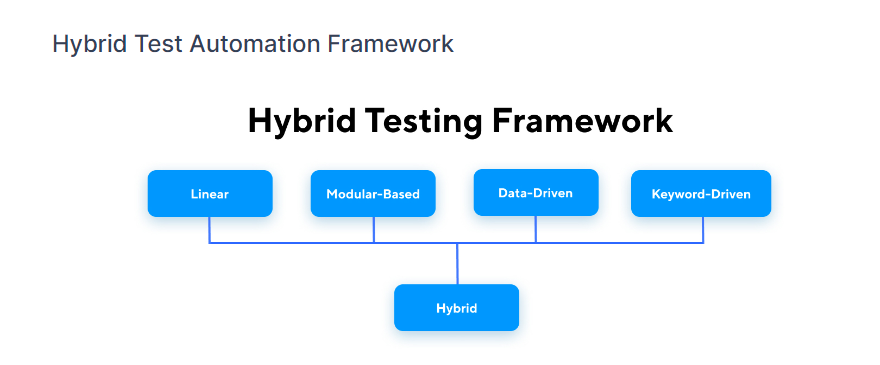
Ex: Regression, functional, load testing, smoke testing

# General Frameworks

# Data Driven testing Framework







Reference: <https://www.testingxperts.com/blog/test-automation-frameworks>

# Pytest (Python Unit test framework) :

* Unit Test framework for Python (Unittest and Nose)
* Conditionally execute the test cases
* Can set up pre-requiste and post setup (Fixtures)
* Assertions adding
* Generate report

## Conditions:

* Test case file start with test string
* Test case method should start with test
* Execute test cases using pytest
* Execute from pycharm or terminal

## Controlling and skipping

* Write multiple test cases in single file
* Skip execution for specific case
* Conditionally skip execution (skip if)
* Execute any specific test case
* Display detailed results

## Group Test cases together

* Execute specific grp - Progression,Regression,Smoke,Sanity,shakedown

Using @pytest.mark.Regression

* Execute all others other than Specific grp

## Fixtures: (CONFTEST.py)

* Execute something before test cases

Pre-requiste like Env Check, DB connection, Url opening etc

* Execute after test cases

Environment cleaning, cache deleting, DB disconnecting

* EXECUTE ONLY ONCE (At modle level / test case level)

## Pytest Assertions:

* Assertions to compare actual to expected result

# Project Structure:

Base – Save common data to import and use

* Initiate Browser

Driver –

* Keep the driver exe files for easy usage (chromedriver,geckodriver files)

## Test cases

TC: 01 Import “Initiate Browser”

* + Register User
  + Driver Close

TC 02 : Initiate Browser

* + Invalid Registration
  + Driver Close

## Config

* + .cfg file for Application url,Browser details
  + .cfg file for Element locators

## Library:

Import ConfigParser and write a python file (.py file) for reading the config file

## Pages:

## Class designed to represent one or more webpages

* Methods: Actions on web page

## Assertions

**POM (Page Object Model) – Design :**

\*\* create a class for each page

* Methods (Actions that can be performed in the page)

Advantages:

1. Minimize duplicity
2. Reduce development efforts
3. Reduce maintenance Effort and time
4. Increase reusability

**Commands:**

* Execute whole test suite or folder

Pytest –v –s --disable-warnings

* Execute Specific Test case and print statements in console

Pytest –v –s --disable-warnings **<FILE NAME>** : **<Test Case Name>**

* **Execute with a tag or marker**

Pytest –v –s -m **<Mark Name>** --disable-warnings **<FILE NAME.py>**