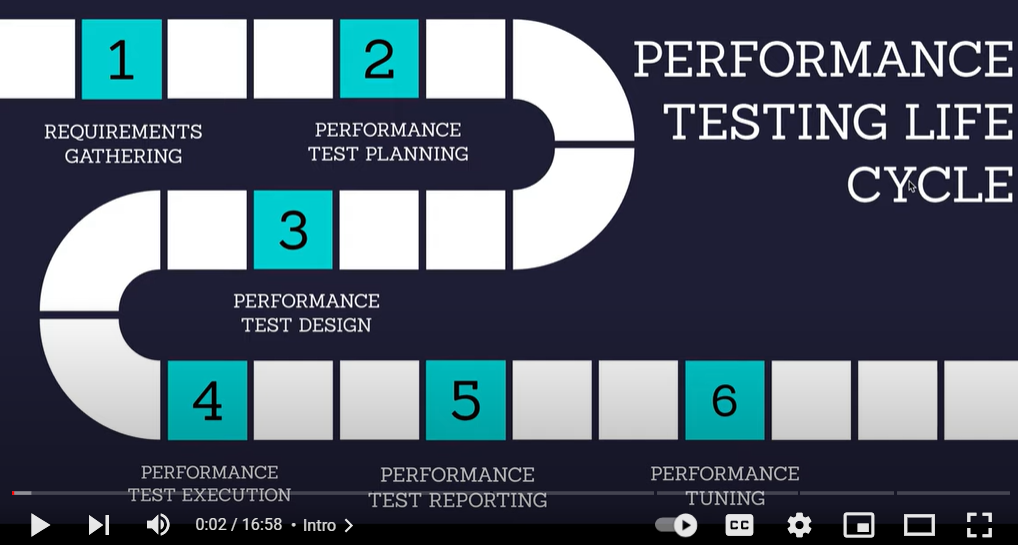
Performance Testing Life Cycle

<https://youtu.be/7di60jrzQLQ>



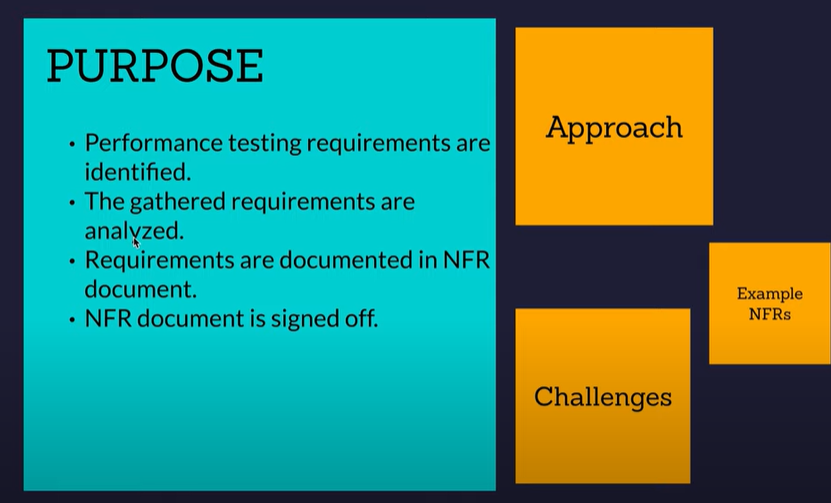
1. **Requirement Gathering Phase**
2. **Performance Test Planning Phase**
3. **Performance Test Design Phase**
4. **Performance Test Execution Phase**
5. **Performance Test Reporting Phase**
6. **Performance Test Tuning phase**

We will discuss about every phase one by one

1. **Requirement Gathering Phase** = in Requirement Gathering phase, we get the requirement from client, Analyze the requirement and capture that requirement in out NFR Document

**What type of requirement?**

* **How many concurrent users would be?**
* **What response time they are expecting?**
* **Any major module in application?**
* **Any Peak time in application that user come in?**
* **What is Error rate that we are targeting**
* **CPU and memory usage should not more than 70% while running the application**

****

**Example of NFR document**

****

In the End NFR Document, should be Signed Off from the client and Business people.

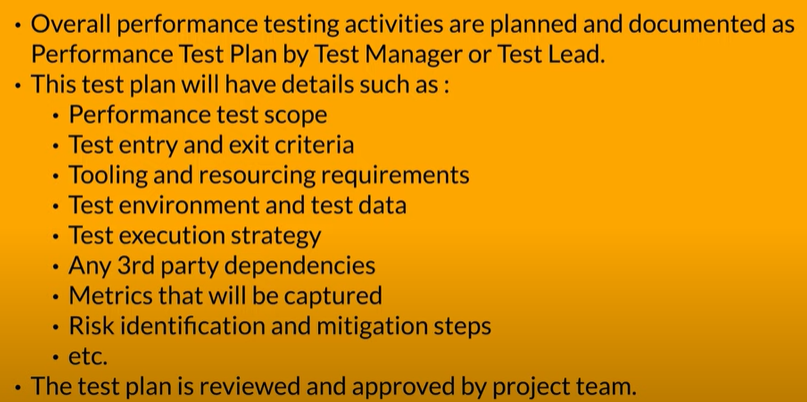
If Client don’t know about performance testing, or don’t know about the requirement so what would be our approach

You will have to ask relevant question to client

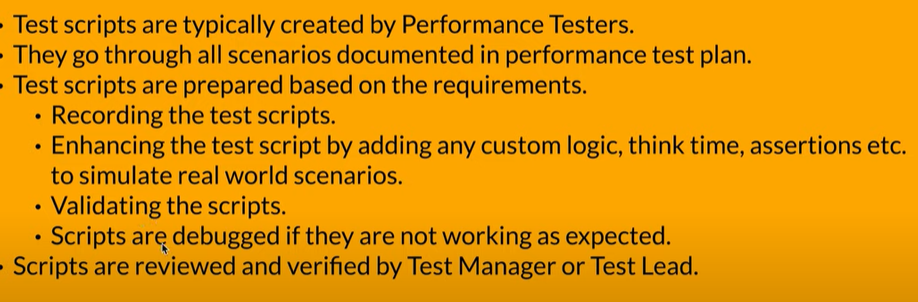
If project is new than what would be your approach?

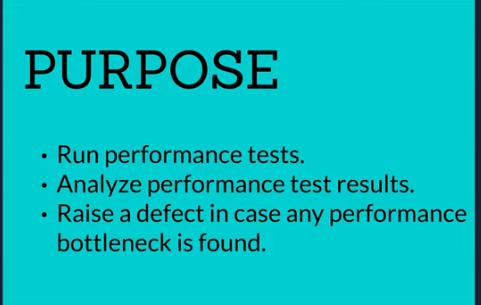
2) **Performance Test Planning Phase =** Based on the Requirement we have received we will create a Performance test planning phase; in performance test plan document we captured this information in details

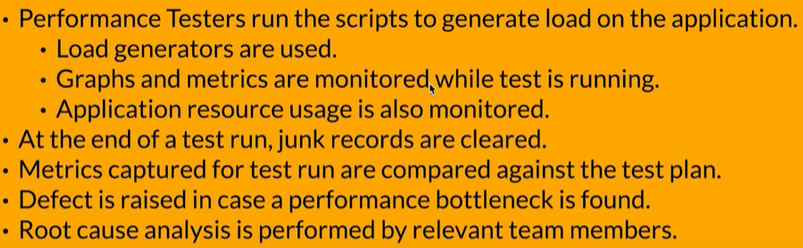
* Performance test scope
* Entry and Exit Criteria
* What tool we required like JMeter, LoadRunner
* Test environment
* Test data
* Any 3rd party dependency
* Resourcing Requirement
* Risk identification



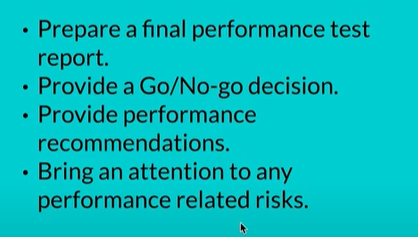
3) **Performance test design phase** = we create a performance test script as per test plan and validate that script are working as expected



4) **Performance Test Execution Phase =** in performance test execution phase we run the script and analyze the performance test result and raise a defect in case of any performance bottleneck 



5) **Performance Test Reporting Phase =** After running the scrip, we will create Performance Test Report, in that report we will mention all details like **any Observation, Suggestion, any performance bottleneck, Go/No Go decision**



6) **Performance Test Tuning phase** = In Performance tuning phase we need to tune our application.

While execution our JMeter script than we noticed our CPU and memory took more than 70 -80% usage than we need to tune our application. Performance tuning can be on hardware and software side.

In hardware side – we need to work on our infrastructure level like need to add more CPU in our infrastructure, if we have memory usage than we need to increase our RAM as well for same instance.

In Software side – if any coding issue noticed, while running script than developer need to work on it.

**After doing the performance tuning, we need to run again the script to check the performance of application and observe the CPU, MEMORY and bottleneck**

**Performance Bottleneck**

<https://youtu.be/GZN6b69rfwg>

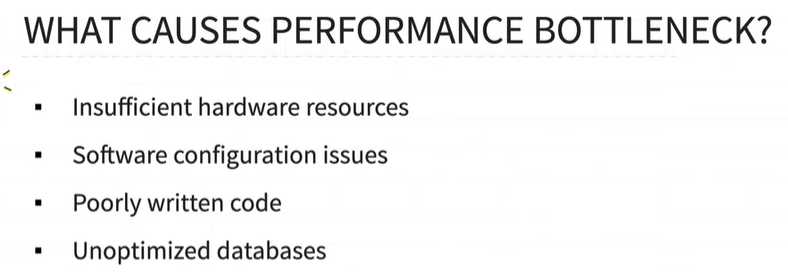
Performance bottleneck is **when your application performance starting to degrade**, it can be cause in many way

Various Types of Performance Bottlenecks, you can find all bottleneck information and can identify from APM Tool like in Dynatrace, New Relic etc. Also, You can check the performance bottleneck in CONSOLE , SERVER LOG and Report JTL File that will generate after result. While execution of script you have to monitor it.

Performance Bottleneck Types:

* **Memory Bottlenecks:** This type of performance bottleneck typically occurs when your system has insufficient RAM which could cause excessive paging or swapping.

java.lang.OutOfMemoryError: A prevalent sign is the java.lang.OutOfMemoryError error. This error has several detailed messages that would allow you to determine if there is a memory leak or not.

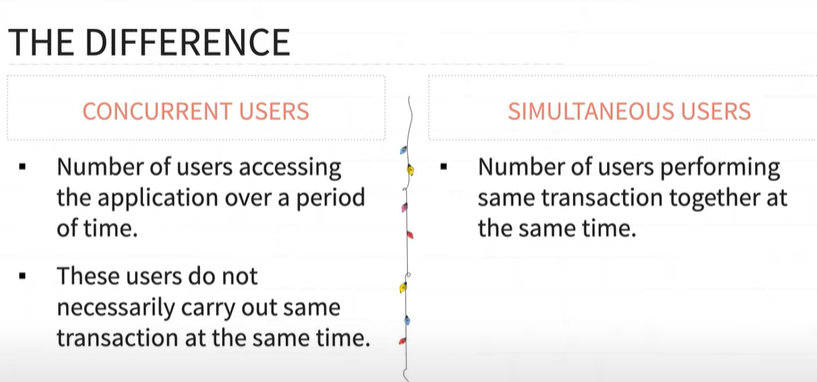
* **CPU Bottlenecks:** This happens when your processor is overburdened which leads to high CPU utilization and causes slow processing speeds.
* **Disk I/O Bottlenecks:** This is a result of slow disk operations which affects your system’s data retrieval and storage speeds.
* **Network Bottlenecks:** This happens due to limited bandwidth or high latency in network communications.
* **Database Bottlenecks:** This type of bottleneck comes from inefficient database queries, lack of indexing, or inadequate database optimizations. From Database Side, you can find out the **SLOW Query**
* 

**Concurrent and Simultaneous Users in Performance Testing**

Concurrent user means multiple users accessing the application at a same point of time and doing different different type of activity whereas Simultaneous users mean multiple users doing same activity at same time

Concurrent user means multiple users accessing the application at same time and doing different transaction/activity

Simultaneous users mean multiple users accessing the application at same time and doing same transaction/activity



**Difference between Synchronization and Asynchronization**

Synchronization – Synchronization means **one request at a time when one request is complete after that second request will start to work**, its work one by one means sequentially.

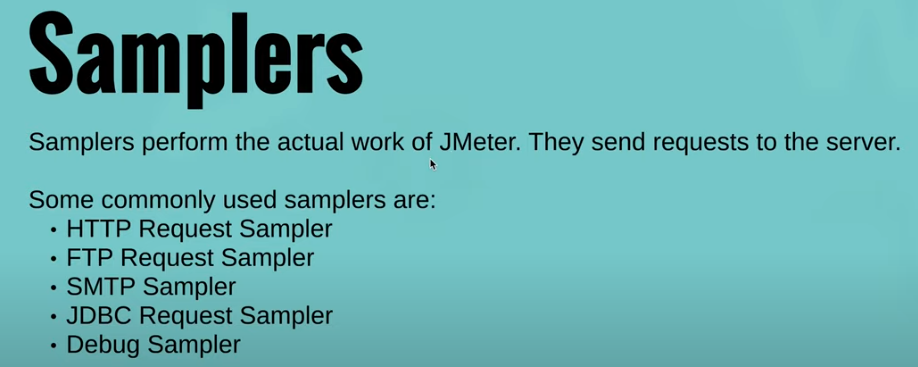
Asynchronization – Asynchronization means all request will work in application **parallelly**



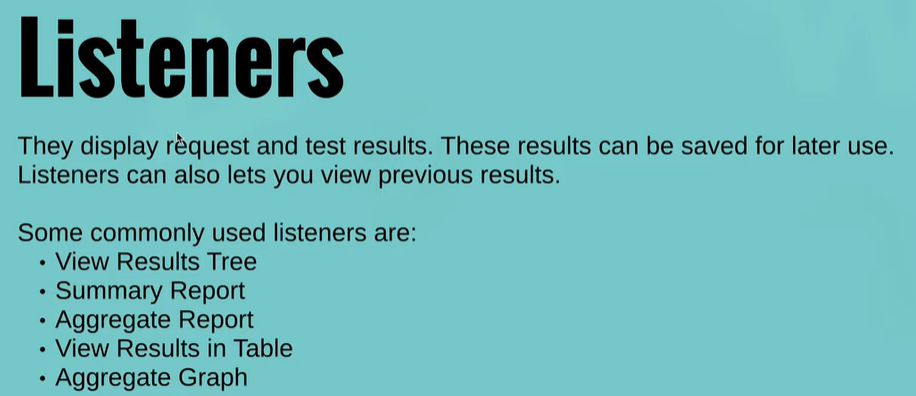
Understanding JMETER Component

<https://youtu.be/SiS1UA9-SLc>

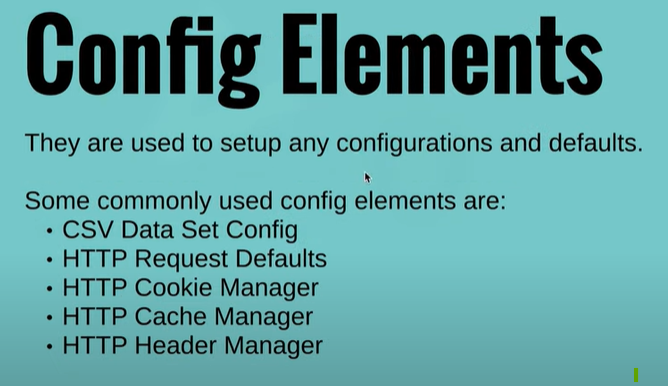
1. Sampler -its perform actual work of JMeter, its send a request to server



1. Listener = its just for a result



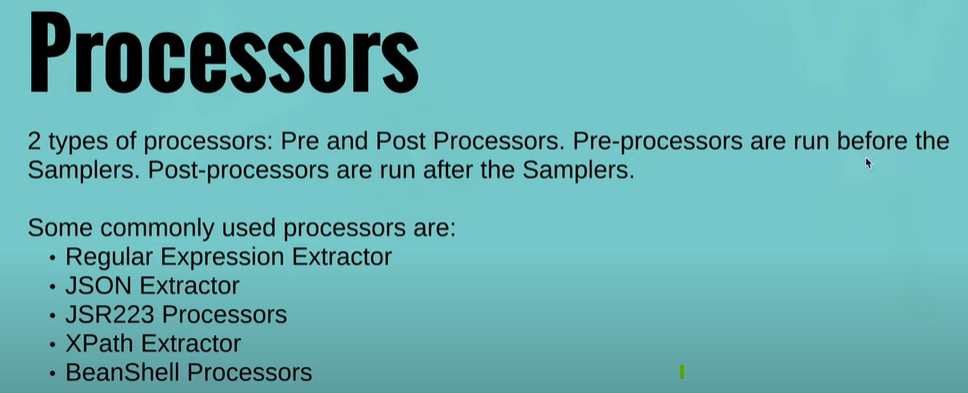
1. Config Element – its just for configuration purpose



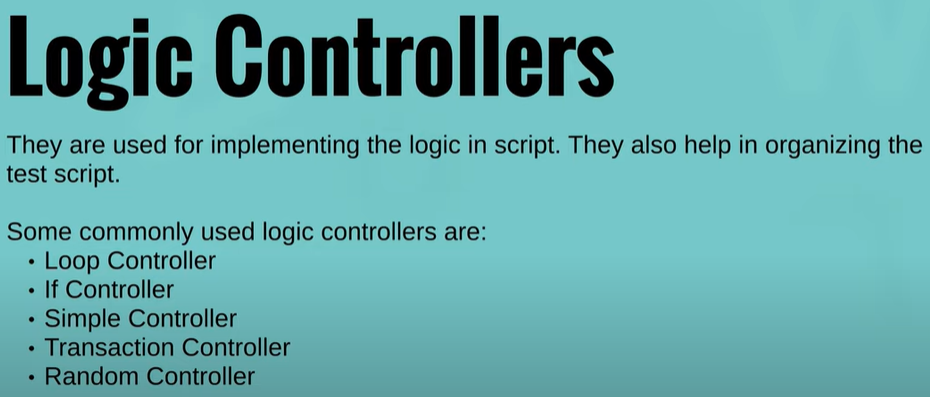
1. Processor – two types of processors, one is **pre processor** and other one is **post preprocessor**

Pre processor –preprocessor are run before the samplers

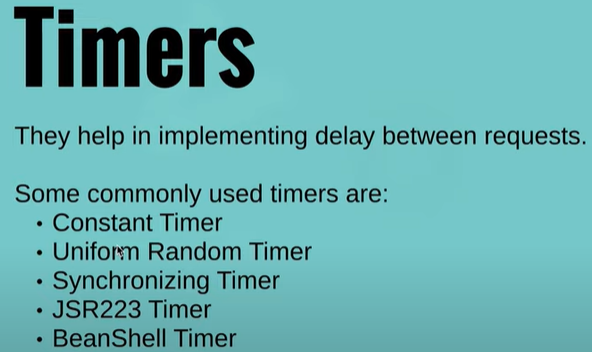
Post preprocessor – post processor are run after the samplers



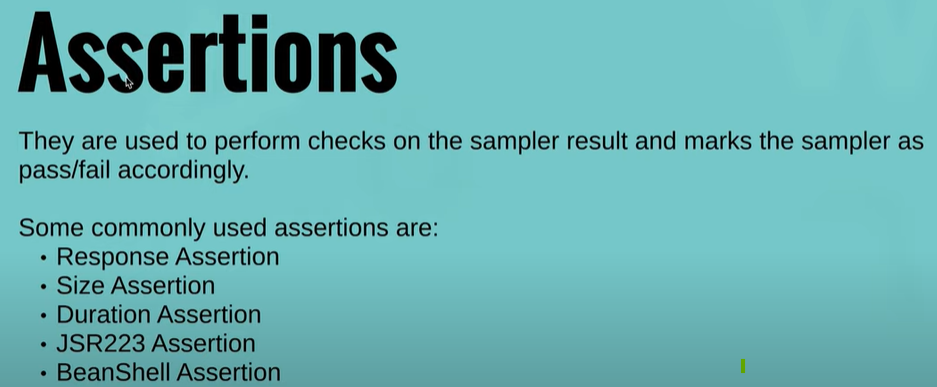
1. Logic controller – its define the logic in script



1. Timers – its for delay between request



1. Assertion – its just for checking the result



Performance Testing Type

1. Stress Testing = Testing the Application performance under extreme workload and check the **breaking point of an application**
2. Load Testing = Testing the Application Performance is used to determine the **how system perform under a given workload.**
3. Volume Testing = volume means data, Testing the application performance with **huge volume of data.**
4. Endurance Testing/Soak Testing = Testing the application performance under the **expected load over a long period of time**

OR

Endurance testing is done to make sure the software application can handle the expected load over a long period of time

1. Scalability Testing = To check the **maximum user load** can handle the software application
2. Spike Testing = **Suddenly large number of users comes into application** and observe the system behaviour of application ex- UPTU website at the time of result coming

**why need of Performance testing?**

Loss of Revenue, Loss of Customer, Loss of Productivity, Damage of brand

difference between authentication and authorization in performance testing

Authentication and authorization are two vital information security processes, Authentication **verifies the identity of a user or service**, and authorization determines their **access rights**.

Diff b/w Think Time and paceing

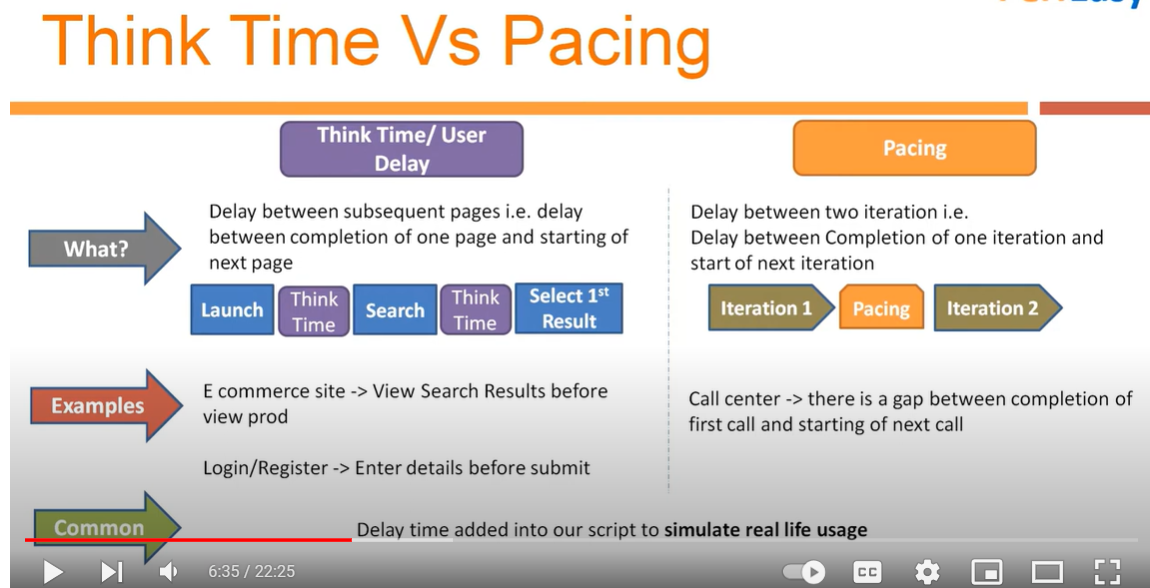
Think Time = delay between two transactions

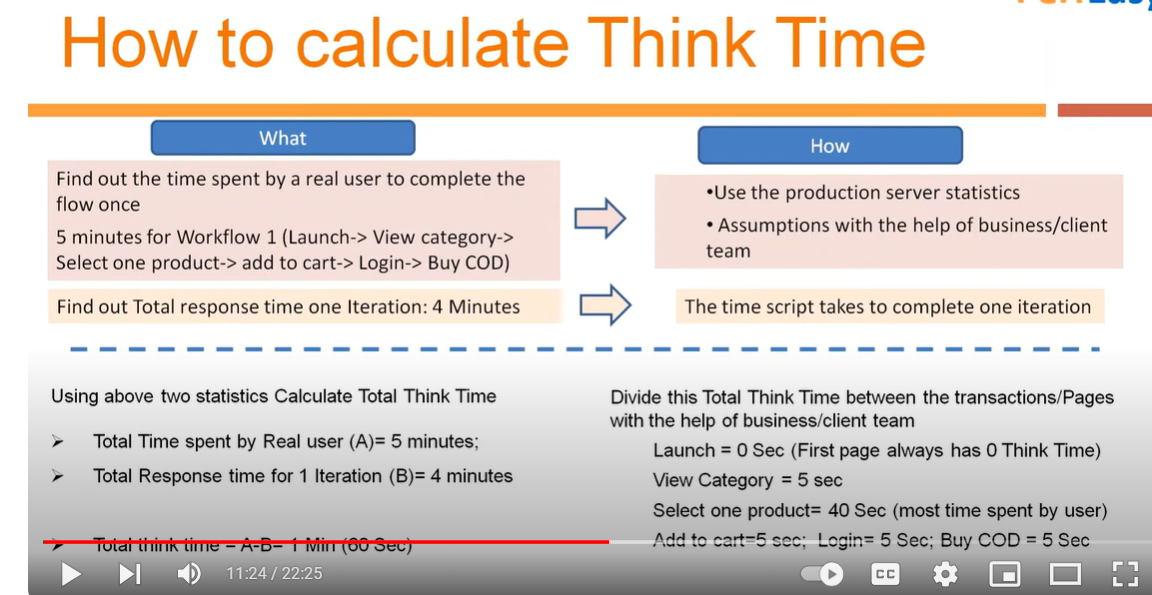
Delay b/w completion of one page and starting of next page

paceing = delay between two iterations

<https://youtu.be/tvxUmrQoHBY>

Delay b/w completion of one iteration and start of next iteration





Aggregate Report

<https://youtu.be/aJUeJzKNwPc>

Samples - sample is number of users what we defined on thread group

average - average time of these sample, JMeter takes the average time of these samples which is average time

median - 50% of the request time

90% - 90 % of request time at how many time 90% of these request got executed successfully