

# **Apache JMeter**

Caio Guilherme de Lira dos Santos  
Lucas Monteiro Duete

# Sobre Apache JMeter

O aplicativo **Apache JMeter** é um software de código aberto, uma aplicação totalmente escrita em Java, projetada para carregar o comportamento funcional do teste e medir o desempenho. Ele foi projetado originalmente para testar aplicativos da Web, mas desde então foi expandido para outras funções de testes.

# O que posso fazer com isso?

O Apache JMeter pode ser usado para testar o desempenho de recursos estáticos, dinâmicos e aplicativos dinâmicos da Web. Ele pode ser usado para simular uma carga pesada em um servidor, grupo de servidores, rede ou objeto para testar sua força ou para analisar o desempenho geral sob diferentes tipos de carga.

# Os recursos do Apache JMeter incluem

Capacidade de testes de carga em diferentes tipos de aplicativos / servidores / protocolos

- Web - HTTP, HTTPS (Java, NodeJS, PHP, Go ...)
- SOAP / REST Webservices
- Protocolos FTP, TCP
- Base de dados via JDBC
- SMTP (S), POP3 (S) e IMAP (S)
- Comandos nativos ou scripts de shell
- Objetos Java

# Os recursos do Apache JMeter incluem

- IDE de teste com características completas que permite uma rápida análise do Plano de teste
- Modo de linha de comando (Non GUI / headless mode)
- Um relatório HTML
- Correlação fácil através da capacidade de extrair dados dos formatos de resposta mais populares, HTML , JSON , XML ou qualquer formato textual
- **100% Java**

# Os recursos do Apache JMeter incluem

- O quadro **multi-threading** completo permite a amostragem simultânea por muitos threads e a amostragem simultânea de diferentes funções por grupos de threads separados.
- Cache e análise offline / reprodução de resultados de teste.

# Os recursos do Apache JMeter incluem

- **Extensível:**

- Os amostradores plugáveis permitem capacidades de teste ilimitadas.
- **Scriptable Samplers** (linguagens compatíveis com JSR223 como Groovy e BeanShell).
- Os **plugins de análise e visualização de dados** permitem grande extensibilidade, bem como personalização.
- As funções podem ser usadas para fornecer entrada dinâmica para um teste ou fornecer manipulação de dados.

# Download

Download Apache JMeter:

[http://jmeter.apache.org/download\\_jmeter.cgi](http://jmeter.apache.org/download_jmeter.cgi)

<https://github.com/apache/jmeter>



# Execução

- GUI

- Linux:

- `sh installPath/bin/jmeter.sh`

- `./installPath/bin/jmeter.sh`

- Windows:

- `installPath/bin/jmeter.bat`

- CLI:

- `jmeter -n -t [localização do script de testes] -l [localização do arquivo de resultados]`

# Apresentação

- <https://github.com/lucasduete/jmeter-aula>
- [https://docs.google.com/presentation/d/1\\_WxRtE5qeTW2MacrSdFOkYSDq9d0K\\_chnL4eFEulJDY/edit#slide=id.g357f890dcb\\_0\\_0](https://docs.google.com/presentation/d/1_WxRtE5qeTW2MacrSdFOkYSDq9d0K_chnL4eFEulJDY/edit#slide=id.g357f890dcb_0_0)

# Elementos

- Thread Groups -> Grupo de acesso na aplicação:
  - Number of Threads -> Equivalente ao numero de usuarios
  - Ramp up period -> Equivalente ao tempo de espera entre as interações dos usuários
  - Loop Count -> Quantidade de interações por usuário
    - Caso Marcado “Forever” Rodam Eternamente

# Elementos: Requests

- HTTP Request -> Requisição HTTP
  - Protocol: Definir protocolo (HTTP, HTTPS), Default: http
  - Server Name: Endereço do servidor ou ip
  - Port Number: Porta da Aplicação, Default:80
  - Method: Método de Acesso
  - Path: Endereço relativo, caso index ponha ‘/’
- HTTP Header Manager -> Gerenciador de Headers de uma HTTP Request

# Elementos: Requests

- FTP Request -> Requisição FTP
  - Server Name: Dns do servidor ou endereço ip
  - Remote File: Endereço do PC Remoto para o Arquivo que será baixado ou para onde será enviado
  - Local File: Endereço no PC local do Arquivo que será enviado ou para onde será baixado
  - get(RERT) -> Faz o Download do Arquivo
  - put(STOR) -> Faz o Upload do Arquivo
  - Password e Username: Dados de Conexão FTP

# Elementos: Requests

- JDBC Request -> Requisição JDBC (Banco de Dados)
  - Query Type: Define o tipo da query
  - Variable Name: Define o nome da Conexão
- JDBC Connect Configuration -> Configuração da Conexão
  - Variable Name: Define o nome da Conexão
  - Database URL: Url de Conexão com o BD
  - JDBC Driver Class: Seleciona o drive de Conexão (Baixar Driver antes e por em pasta LIB)
  - Username e Password: Dados de Autenticação com o BD

# Elementos: Listeners

- Listener Result Tree -> Lista os resultados das requisições em formato de árvore (Necessário muito processamento e memória)
- Listener Result Graph -> Lista os resultados das requisições em formato de grafo (Necessário muito processamento e memória)
- Listener Result table -> Formato Padrão, Lista as Requisições em Tabela com atributos como desempenho, tamanho, latencia e código de sucesso ou erro

# Elementos: Listeners

- Listener Result Summary -> Lista os resultados das requisições em formato de tabela informando dados estatísticos como média mínimo, máximo, taxa de erro, 99%, 95% e 90% do resultados(Análise dos 5 pontos)
- Listener Result Graph -> Lista os resultados das requisições em formato de tabela dando alguns dados estatísticos, Desvio padrão, média, min, max, bytes recebidos e enviados



# Elementos: Assertions

- Response Assertion -> Assert para cada resposta
  - Apply to: Escopo do Assert (Todas as requisições, para cada requisição)
  - Field to Test: Local onde a assertiva será testada
    - Text response
    - Response code
    - Response message
    - Request, Response Headers

# Response Assertion

- Pattern to Test -> O que será esperado no assertiva
- Pattern Matching Rules -> Qual o operador será usado na assertiva
  - Contains
  - Matches
  - Equals
  - Substring
    - Not, OR -> Opcionais e podem ser colocados juntos

# Atividade

Realize Testes nos Métodos:

- Login
- Upload
- Download

Na API UpSafe.