JMETER Academy:

1. Protocolos:
   * HTTP: El protocolo de transferencia de hipertexto (HyperTextTranaferProtocol), es una serie de normas o reglas para que el cliente le envíe una petición al servidor y el servidor le devuelva una respuesta al cliente. Establece la forma en que debe estar la reqeust y la response.
     + Métodos:
     + StatusCode:
   * HTTPS. Es la versión de seguridad.
   * SOAP Simple (Object Access Protocol): Usa HTTP o HTTPS, USA XML.
   * REST Representacional State Transferer: Usa HTTP o HTTPS, Más rápido, puede usar JSON, XML, YAML.
   * JDBC (Java Database Connectivity): Sirve para conectar a la base de datos, hacer CRUD, por medio de SQL.
2. Elementos JMeter:
   * Plan de trabajo (TestPlan): El inicio de un proyecto.
   * Grupos de hilos (Thread Groups): Flujos de trabajo.
   * Peticiones (Request): Campos identificados con nombre y valores.
   * Correlación
     + Pre procesadores (Pre-Processors): Procesos que guardan variables antes de una ejecución.
     + Post Procesadores (Post-Processors): Procesos que guardan variables después de una ejecución.
   * Expresiones regulares (REgular EXpressions): Definen patrones de búsqueda.
   * Parametrización: Valores dinámicos para su ejecución.
   * Aserciones (Assertions): Comparación del resultado obtenido con el deseado.
   * Temporizadores (Timers): Tiempo de espera para seguir una acción.
3. Listeners:
   * Árbol de resultados (View results tree): Evidencia si las solicitudes ejecutadas fueron correctas.
   * Informe agregado (Aggregate report): Muestra de los percntiles, y la medida de tendencia central.
   * Informe de resumen (Summary report): Proporciona una visión general de los resultados de una prueba.
4. Acceso directo y ejecución desde consola:
   * Acceso directo:
     + En la ruta de la instalación abrir el CMD dentro de bin
     + Crear un documento de texto con la extensión .bat
     + Editar el documento con este texto: java –jar ApacheJmeter.jar, guardarlo como una
     + Crear el acceso directo.
   * Ejecutar desde la consola:
     + En la ruta de la instalación abrir el CMD dentro de bin
     + Copiar la siguiente línea de comandos: jmeter -n -t C:\Users\bumanag\Documents\Practica\_Performance\JMeter\1-Orange\_HRB\Trx\_Recruitement.jmx -l C:\Users\bumanag\Documents\Practica\_Performance\JMeter\1-Orange\_HRB\Log
     + Cambiar la ruta donde se encuentra el proyecto
     + Cambiar la ruta donde se registrarán los logs del proyecto
5. Generar reporte:
   * GIU:
     + Ir a Tools -> Generate report HTML
     + Seleccionar el archivo .jlt de salida
     + Seleccionar el user.properties de la carpeta jmeter bin
     + Seleccionar el archivo de salida
   * CMD:
     + Utilizar el comando: jmeter -n -t [jmeter file] -l [result file] -e –o [Report File]

JMETER Introducción:

1. (52) Introducción a JMeter
2. Contenido:
   1. Descripción de JMeter.
   2. Interface del JMeter.
   3. Creación de un Template.
   4. Grabar Request con JMeter.
3. Descripción de JMeter parte 1:
   1. Es una aplicación de escritorio de sofware libre. 100% Java.
   2. ¿Qué se puede hacer con la herramienta?: Se pueden hacer pruebas de rendimiento con recursos estáticos o dinámicos. (archivos, servlets, scripts, objeto Java, Bases de datos, Queries, servidores FTP).
   3. JMeter no es un explorador: No ejecuta loss Javascripts encontrados en algunas páginas HTML.
4. Descripción de JMeter parte 2:
   1. Características:
      1. Puede cargar y realizar pruebas en diferentes tipos de servidores, tales como:
         1. HTTP / HTTPS
         2. SOAP
         3. JDCB
         4. SMTP POP3 IMAP
      2. Completa portabilidad y 100% JAVA
      3. Interfaz de usuario que permite el diseño de las operaciones.
      4. Captura de peticiones, resultados de prueba
      5. Funciones que pueden ser usadas para proveer entradas dinámicas o extracción de datos.
   2. Prerrequisitos:
      1. Tener JAVA Versión 8 o superior
5. Descripción de JMeter parte 3:
   1. No se instala, solo se descomprime:
      1. Ir ala tura de la página oficial
      2. Descargar el reléase
      3. Descomprimir en la ruta más adecuada.
6. Interfaces de JMeter parte 1:
   1. Descripción de los componentes del sistema:
      1. Nuevo, Plantillas, Abrir, Guardar
      2. Undo, Redo (Modificando la configuración)
      3. Cortar, Copiar, Pegar
      4. Expandir, Reducir, Interrupto de habilitar y deshabilitar
      5. Validar los grupos de hilos (Con la librería)
      6. Comenzar, Comenzar sin pausa
      7. Parar, Cancelar
      8. Limpiar, Limpiar todo
      9. Buscar, Limpiar búsqueda
      10. Función de ayuda con código, Ayuda
      11. Tiempo de ejecución, Errores / Advertencias, Hilos usados
      12. Pluggins
7. Creación de template parte 1:
   1. Ir a plantillas
   2. Seleccionar en crear
8. Creación de template parte 2:
   1. Testplan: Describir el objetivo de testplan, autor, fecha de creación, tipos de pruebas.
   2. Grupos de hilos: La carga que se va a lanzar, cantidad de usuarios e información, tiempo de inyección y carga sostenida.
   3. Controlador de transacción: Describir que peticiones viajan en ese grupo y las cabeceras con sus datos.
   4. Peticiones: Describir la data y las variables que usa.
   5. Muestreadores: La ruta que escribe los resultados en un archivo plano.
   6. Nombre del archivo: .jmx nombre de la herramienta
9. Grabar Request con JMeter parte 1:
   1. Configurar el proxy de JMeter en el navegador web, con el fin de que la navegación sea capturada.
10. Grabar Request con JMeter parte 2:
    1. Configurar el equipo con el proxy identificado.
11. Grabar Request con JMeter parte 3:
    1. Instalar el certificado
    2. Configurar la IP Local
12. Grabar Request con JMeter parte 4:
    1. Seleccionar el Recording controller
13. Grabar Request con JMeter parte 5:
    1. Para pararlo se usa con el botón STOP
14. Grabar Request con JMeter parte 6:
    1. Observar el árbol de resultados
15. Gracias

Día 1 - Configuración:

* Descargar JMeter
* Descargar la librería pluggins-manager y colocarlo en la ruta: lib-> ext
* Ir a Jmeter.properties:
  + Quitar el comentario a: language:en
  + Colocar el comentario a: undo.history.size=5
  + Cambiar la estructura a: proxy.sampler\_format #(counter,number,000).#{name} - #{path}
* Ir a la parte derecha de las librerías y colocar los pluggins necesarios.
* Seleccionar el template: Recording
* Recomendación: Usar Firefox y borrar cookies.
* Subir el certificado RootCA.crt
* Tener el mismo puerto en el proxy del navegador y Test Script Reorder.
* Ir a Request Filtering grabar solo lo necesario.
* Cambiar el nombre por cada acción en pantalla.
* El proyecto debe tener 2 carpetas: Logs y Data.
* Para Logs del buscador, guardarlo en la ruta, con la extensión Log(nombre).jtl
* SI son varios servicios: so (servicio orquestador), s- (Solo un servicio) y Trx (Transacciones que utilicen front.
* Target Controller: Use Recording Controller
* Gouping: Do not group samplers
* Naming Scheme: Use format string
* Duplicar el script guardado
* En caso de que haya peticiones que no se utilicen, primero deshabilitar y luego eliminar, recomendación para saber, lo que no tengan una URL con el servicio válido.

Día 2 – Ejercicio:

* Correcciones:
  + Conceptos: Usuarios y clientes.
  + Cómo poner una fila de información
  + No dejar espacios en blanco con los nombres de variables
  + Data con ruta relativa
* Random variable: Cuántos usuarios quiero que se ejecuten
* Loop controller: Dentro de loop controller van las URL que se van a ejecutar. Colocar la variable de radom
* Debug processor: Me muestra paso a paso los datos, deshabilitar cuando sean ventanas, solo para creación de scripts.
* User defined variable: Crear variables globales: ${\_BeanShell…)
* Simple Data Writer: Crea logs con la ubicación y la fecha, yyyy-MM-dd\_HHmmss
* Sample Data Writer Error: La misma ruta con diferente nombre y con le checklist de errors
* Aggregate Report: Para ver cuántas veces se ejecuta

Día 3 - Corrección y Logs:

* Sentencias de control:
  + Sentencia de control If: Add -> Logic Controller -> Se coloca una variable a ser comparada, y dentro del If el script o la condición a realizar.
  + Sentencia de control while: Respaldo de la original. Mientras que no sea mayor de X número, seguir ejecutando
  + LOOP: No es necesario una condición
* Escribir logs personalizados:
  + JSR2333 -> Pasar solo las comillas
  + Groovy
* Procesamiento:
  + Pre:
  + Post:
* Registros serializados a nivel de logs:
  + Exitosos
  + Errores
  + Reporte