Apache HttpClient

Apache Commons HttpClient project

- Is the old project that produced HttpClient 3.1.

- Commons HttpClient library is now deprecated.

Apache HttpComponents project

- Is new proect that produces HttpClient 4.x.

* Apache Commons HttpClient -> Apache HttpComponents
* Apache HttpComponents
  + HttpCore
    - A set of low level HTTP transport components that can be used to build custom client and server side HTTP services with a minimal footprint
  + HttpClient
    - Is a HTTP/1.1 compliant HTTP agent implementation based on HttpCore.
  + HttpComponents AsyncClient
    - Is a HTTP/1.1 compliant HTTP agent implementation based on HttpCore NIO and HttpClient components.
* Methods
  + GET, HEAD, POST, PUT, DELETE, TRACE and OPTIONS
  + HttpGet, HttpHead, HttpPost, HttpPut, HttpDelete, HttpTrace, and HttpOptions.

httpbin(1): HTTP Request & Response Serviceuestions

1. <http://httpbin.org/>
2. Testing an HTTP Library can become difficult sometimes. RequestBin is fantastic for testing POST requests, but doesn't let you control the response. This exists to cover all kinds of HTTP scenarios. Additional endpoints are being considered.

Maven (pom.xml) setup of Apache HttpComponents

pom.xml

<dependencies>

<!-- https://mvnrepository.com/artifact/org.testng/testng -->

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>6.11</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/org.json/json -->

<dependency>

<groupId>org.json</groupId>

<artifactId>json</artifactId>

<version>20170516</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.httpcomponents/httpclient -->

<dependency>

<groupId>org.apache.httpcomponents</groupId>

<artifactId>httpclient</artifactId>

<version>4.5.3</version>

</dependency>

</dependencies>

<properties>

<maven.compiler.target>1.8</maven.compiler.target>

<maven.compiler.source>1.8</maven.compiler.source>

</properties>

**testing v6.11 wants**

- com.beust:jcommander:1.64

- org.yaml:snakeyaml:1.17

**json v20170516**

- org.json:json:20170516

**httpclient v4.5.3**

- org.apache.httpcomponents:httpclient:4.5.3

- org.apache.httpcomponents:httpcore:4.4.6

- commons-codec:commons-logging:1.2

- commons-codec:commons-codec:1.9

Eclipse setup of Apache/HttpClient 4.5.2 (GA)

<?xml version="1.0" encoding="UTF-8"?>

<projectDescription>

<name>http-requests</name>

<comment></comment>

<projects>

</projects>

<buildSpec>

<buildCommand>

<name>org.eclipse.jdt.core.javabuilder</name>

<arguments>

</arguments>

</buildCommand>

</buildSpec>

<natures>

<nature>org.eclipse.jdt.core.javanature</nature>

</natures>

<linkedResources>

<link>

<name>httpcomponents-client-4.5.2</name>

<type>2</type>

<location>/Users/ben/Documents/Software/Apache/HttpClient 4.5.2 (GA) Source/httpcomponents-client-4.5.2</location>

</link>

</linkedResources>

</projectDescription>

**How to url encode a string**

String q = "random word £500 bank $";

String url = "http://example.com/query?q=" + URLEncoder.encode(q, "UTF-8");

Questions

1. List and describe each supported http request method.
2. What are the different http request schemes?
3. What is HttpCore NIO?
4. SSL/TLS protocol is the most widely used technique to ensure HTTP transport security. What is it?

Java™ Secure Socket Extension (JSSE) Reference Guide http://docs.oracle.com/javase/6/docs/technotes/guides/security/jsse/JSSERefGuide.html

TestNG Assert (actual, expected)

Assert.assertEqual(String actual,String expected) : Ass