

```

38099@MILA MINGW64 ~/Desktop
$ cat Stresstest.jmx
<?xml version="1.0" encoding="UTF-8"?>
<jmeterTestPlan version="1.2" properties="5.0" jmeter="5.5">
  <hashTree>
    <TestPlan guiclass="TestPlanGui" testclass="TestPlan" testname="Test Plan" e
nabled="true">

```

```

38099@MILA MINGW64 ~/Desktop
$ cat Stresstest.jmx
<?xml version="1.0" encoding="UTF-8"?>
<jmeterTestPlan version="1.2" properties="5.0" jmeter="5.5">
  <hashTree>
    <TestPlan guiclass="TestPlanGui" testclass="TestPlan" testname="Test Plan"
enabled="true">
      <stringProp name="TestPlan.comments"></stringProp>
      <boolProp name="TestPlan.functional_mode">false</boolProp>
      <boolProp name="TestPlan.tearDown_on_shutdown">true</boolProp>
      <boolProp name="TestPlan.serialize_threadgroups">false</boolProp>
      <elementProp name="TestPlan.user_defined_variables" elementType="Arguments"
guiclass="ArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
        <collectionProp name="Arguments.arguments"/>
      </elementProp>
      <stringProp name="TestPlan.user_define_classpath"></stringProp>
    </TestPlan>
  <hashTree>
    <ResultCollector guiclass="ViewResultsFullVisualizer" testclass="ResultCollector"
testname="View Results Tree" enabled="true">
      <boolProp name="ResultCollector.error_logging">false</boolProp>
      <objProp>
        <name>saveConfig</name>
        <value class="SampleSaveConfiguration">
          <time>true</time>
          <latency>true</latency>
          <timestamp>true</timestamp>
          <success>true</success>
          <label>true</label>
          <code>true</code>
          <message>true</message>
          <threadName>true</threadName>
          <dataType>true</dataType>
          <encoding>false</encoding>
          <assertions>true</assertions>
          <subresults>true</subresults>
          <responseData>false</responseData>

```

```

    <samplerData>false</samplerData>
    <xml>false</xml>
    <fieldNames>true</fieldNames>
    <responseHeaders>false</responseHeaders>
    <requestHeaders>false</requestHeaders>
    <responseDataOnError>false</responseDataOnError>
    <saveAssertionResultsFailureMessage>true</saveAssertionResultsFailureMessage>
    <assertionsResultsToSave>0</assertionsResultsToSave>
    <bytes>true</bytes>
    <sentBytes>true</sentBytes>
    <url>true</url>
    <threadCounts>true</threadCounts>
    <idleTime>true</idleTime>
    <connectTime>true</connectTime>
  </value>
</objProp>
<stringProp name="filename"></stringProp>
</ResultCollector>
<hashTree/>
<ResultCollector guiclass="SummaryReport" testclass="ResultCollector"
testname="Summary Report" enabled="true">
  <boolProp name="ResultCollector.error_logging">false</boolProp>
  <objProp>
    <name>saveConfig</name>
    <value class="SampleSaveConfiguration">
      <time>true</time>
      <latency>true</latency>
      <timestamp>true</timestamp>
      <success>true</success>
      <label>true</label>
      <code>true</code>
      <message>true</message>
      <threadName>true</threadName>
      <dataType>true</dataType>
      <encoding>false</encoding>
      <assertions>true</assertions>
      <subresults>true</subresults>
      <responseData>false</responseData>
      <samplerData>false</samplerData>
      <xml>false</xml>
      <fieldNames>true</fieldNames>
      <responseHeaders>false</responseHeaders>
      <requestHeaders>false</requestHeaders>
      <responseDataOnError>false</responseDataOnError>

```

```

    <saveAssertionResultsFailureMessage>true</saveAssertionResultsFailureMessage>
    <assertionsResultsToSave>0</assertionsResultsToSave>
    <bytes>true</bytes>
    <sentBytes>true</sentBytes>
    <url>true</url>
    <threadCounts>true</threadCounts>
    <idleTime>true</idleTime>
    <connectTime>true</connectTime>
  </value>
</objProp>
<stringProp name="filename"></stringProp>
</ResultCollector>
<hashTree/>
<ResultCollector guiclass="GraphVisualizer" testclass="ResultCollector" testname="Graph
Results" enabled="true">
  <boolProp name="ResultCollector.error_logging">false</boolProp>
  <objProp>
    <name>saveConfig</name>
    <value class="SampleSaveConfiguration">
      <time>true</time>
      <latency>true</latency>
      <timestamp>true</timestamp>
      <success>true</success>
      <label>true</label>
      <code>true</code>
      <message>true</message>
      <threadName>true</threadName>
      <dataType>true</dataType>
      <encoding>false</encoding>
      <assertions>true</assertions>
      <subresults>true</subresults>
      <responseData>false</responseData>
      <samplerData>false</samplerData>
      <xml>false</xml>
      <fieldNames>true</fieldNames>
      <responseHeaders>false</responseHeaders>
      <requestHeaders>false</requestHeaders>
      <responseDataOnError>false</responseDataOnError>
      <saveAssertionResultsFailureMessage>true</saveAssertionResultsFailureMessage>
      <assertionsResultsToSave>0</assertionsResultsToSave>
      <bytes>true</bytes>
      <sentBytes>true</sentBytes>
      <url>true</url>
      <threadCounts>true</threadCounts>

```

```

        <idleTime>true</idleTime>
        <connectTime>true</connectTime>
    </value>
</objProp>
<stringProp name="filename"></stringProp>
</ResultCollector>
<hashTree/>
<ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
    <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
    <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
        <boolProp name="LoopController.continue_forever">false</boolProp>
        <stringProp name="LoopController.loops">1</stringProp>
    </elementProp>
    <stringProp name="ThreadGroup.num_threads">50</stringProp>
    <stringProp name="ThreadGroup.ramp_time">35</stringProp>
    <boolProp name="ThreadGroup.scheduler">false</boolProp>
    <stringProp name="ThreadGroup.duration"></stringProp>
    <stringProp name="ThreadGroup.delay"></stringProp>
    <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
</ThreadGroup>
<hashTree>
    <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
        <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
            <collectionProp name="Arguments.arguments"/>
        </elementProp>
        <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
        <stringProp name="HTTPSampler.port"></stringProp>
        <stringProp name="HTTPSampler.protocol">http</stringProp>
        <stringProp name="HTTPSampler.contentEncoding"></stringProp>
        <stringProp name="HTTPSampler.path"></stringProp>
        <stringProp name="HTTPSampler.concurrentPool">6</stringProp>
        <stringProp name="HTTPSampler.connect_timeout"></stringProp>
        <stringProp name="HTTPSampler.response_timeout"></stringProp>
    </ConfigTestElement>
</hashTree/>
<HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="GET HTTP Request" enabled="true">

```

```

    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
    <collectionProp name="Arguments.arguments"/>
</elementProp>
<stringProp name="HTTPSampler.domain"></stringProp>
<stringProp name="HTTPSampler.port"></stringProp>
<stringProp name="HTTPSampler.protocol"></stringProp>
<stringProp name="HTTPSampler.contentEncoding"></stringProp>
<stringProp name="HTTPSampler.path">/posts</stringProp>
<stringProp name="HTTPSampler.method">GET</stringProp>
<boolProp name="HTTPSampler.follow_redirects">true</boolProp>
<boolProp name="HTTPSampler.auto_redirects">false</boolProp>
<boolProp name="HTTPSampler.use_keepalive">true</boolProp>
<boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
<stringProp name="HTTPSampler.embedded_url_re"></stringProp>
<stringProp name="HTTPSampler.connect_timeout"></stringProp>
<stringProp name="HTTPSampler.response_timeout"></stringProp>
</HTTPSamplerProxy>
<hashTree>
    <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="GET| Duration Assertion " enabled="true">
    <stringProp name="DurationAssertion.duration">250</stringProp>
</DurationAssertion>
<hashTree/>
    <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="GET| Response Assertion (Response Code)" enabled="true">
    <collectionProp name="Asserion.test_strings">
    <stringProp name="49586">200</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
</ResponseAssertion>
<hashTree/>
    <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="GET| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
    <stringProp name="110371416">title</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>

```

```

        <intProp name="Assertion.test_type">2</intProp>
    </ResponseAssertion>
    <hashTree/>
    <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="GET| Response Assertion (Text Response)" enabled="true">
        <collectionProp name="Asserion.test_strings">
            <stringProp name="3029410">body</stringProp>
        </collectionProp>
        <stringProp name="Assertion.custom_message"></stringProp>
        <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
        <boolProp name="Assertion.assume_success">false</boolProp>
        <intProp name="Assertion.test_type">2</intProp>
    </ResponseAssertion>
    <hashTree/>
</hashTree>
</hashTree>
    <ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
        <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
        <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
            <boolProp name="LoopController.continue_forever">false</boolProp>
            <stringProp name="LoopController.loops">1</stringProp>
        </elementProp>
        <stringProp name="ThreadGroup.num_threads">50</stringProp>
        <stringProp name="ThreadGroup.ramp_time">25</stringProp>
        <boolProp name="ThreadGroup.scheduler">false</boolProp>
        <stringProp name="ThreadGroup.duration"></stringProp>
        <stringProp name="ThreadGroup.delay"></stringProp>
        <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
    </ThreadGroup>
    <hashTree/>
    <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
        <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
            <collectionProp name="Arguments.arguments"/>
        </elementProp>
        <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
        <stringProp name="HTTPSampler.port"></stringProp>
        <stringProp name="HTTPSampler.protocol">http</stringProp>
        <stringProp name="HTTPSampler.contentEncoding"></stringProp>

```

```

<stringProp name="HTTPSampler.path"></stringProp>
<stringProp name="HTTPSampler.concurrentPool">6</stringProp>
<stringProp name="HTTPSampler.connect_timeout"></stringProp>
<stringProp name="HTTPSampler.response_timeout"></stringProp>
</ConfigTestElement>
<hashTree/>
<HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="POST HTTP Request" enabled="true">
  <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
    <collectionProp name="Arguments.arguments">
      <elementProp name="title" elementType="HTTPArgument">
        <boolProp name="HTTPArgument.always_encode">false</boolProp>
        <stringProp name="Argument.value">test1</stringProp>
        <stringProp name="Argument.metadata">=</stringProp>
        <boolProp name="HTTPArgument.use_equals">true</boolProp>
        <stringProp name="Argument.name">title</stringProp>
      </elementProp>
      <elementProp name="body" elementType="HTTPArgument">
        <boolProp name="HTTPArgument.always_encode">false</boolProp>
        <stringProp name="Argument.value">test body</stringProp>
        <stringProp name="Argument.metadata">=</stringProp>
        <boolProp name="HTTPArgument.use_equals">true</boolProp>
        <stringProp name="Argument.name">body</stringProp>
      </elementProp>
      <elementProp name="userId" elementType="HTTPArgument">
        <boolProp name="HTTPArgument.always_encode">false</boolProp>
        <stringProp name="Argument.value">1</stringProp>
        <stringProp name="Argument.metadata">=</stringProp>
        <boolProp name="HTTPArgument.use_equals">true</boolProp>
        <stringProp name="Argument.name">userId</stringProp>
      </elementProp>
    </collectionProp>
  </elementProp>
  <stringProp name="HTTPSampler.domain"></stringProp>
  <stringProp name="HTTPSampler.port"></stringProp>
  <stringProp name="HTTPSampler.protocol"></stringProp>
  <stringProp name="HTTPSampler.contentEncoding">UTF-8</stringProp>
  <stringProp name="HTTPSampler.path">/posts</stringProp>
  <stringProp name="HTTPSampler.method">POST</stringProp>
  <boolProp name="HTTPSampler.follow_redirects">true</boolProp>
  <boolProp name="HTTPSampler.auto_redirects">false</boolProp>
  <boolProp name="HTTPSampler.use_keepalive">true</boolProp>

```

```

    <boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
    <stringProp name="HTTPSampler.embedded_url_re"></stringProp>
    <stringProp name="HTTPSampler.connect_timeout"></stringProp>
    <stringProp name="HTTPSampler.response_timeout"></stringProp>
  </HTTPSamplerProxy>
  <hashTree>
    <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="POST| Duration Assertion " enabled="true">
      <stringProp name="DurationAssertion.duration">300</stringProp>
    </DurationAssertion>
  </hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="POST| Response Assertion (Response Code)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="49587">201</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
  <hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="POST| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="110371416">title</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
  <hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="POST| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="3029410">body</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
  <hashTree/>

```



```

    </hashTree>
  </hashTree>
  <ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
    <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
    <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
      <boolProp name="LoopController.continue_forever">false</boolProp>
      <stringProp name="LoopController.loops">1</stringProp>
    </elementProp>
    <stringProp name="ThreadGroup.num_threads">10</stringProp>
    <stringProp name="ThreadGroup.ramp_time">5</stringProp>
    <boolProp name="ThreadGroup.scheduler">false</boolProp>
    <stringProp name="ThreadGroup.duration"></stringProp>
    <stringProp name="ThreadGroup.delay"></stringProp>
    <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
  </ThreadGroup>
</hashTree>
  <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
      <collectionProp name="Arguments.arguments"/>
    </elementProp>
    <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
    <stringProp name="HTTPSampler.port"></stringProp>
    <stringProp name="HTTPSampler.protocol">http</stringProp>
    <stringProp name="HTTPSampler.contentEncoding"></stringProp>
    <stringProp name="HTTPSampler.path"></stringProp>
    <stringProp name="HTTPSampler.concurrentPool">6</stringProp>
    <stringProp name="HTTPSampler.connect_timeout"></stringProp>
    <stringProp name="HTTPSampler.response_timeout"></stringProp>
  </ConfigTestElement>
</hashTree/>
  <HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="PUT HTTP Request" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" enabled="true">
      <collectionProp name="Arguments.arguments">
        <elementProp name="title" elementType="HTTPArgument">
          <boolProp name="HTTPArgument.always_encode">false</boolProp>
          <stringProp name="Argument.value">test2</stringProp>
        </elementProp>
      </collectionProp>
    </elementProp>
  </HTTPSamplerProxy>
</hashTree>

```

```

    <stringProp name="Argument.metadata">=</stringProp>
    <boolProp name="HTTPArgument.use_equals">true</boolProp>
    <stringProp name="Argument.name">title</stringProp>
  </elementProp>
  <elementProp name="body" elementType="HTTPArgument">
    <boolProp name="HTTPArgument.always_encode">>false</boolProp>
    <stringProp name="Argument.value">HWPutbody</stringProp>
    <stringProp name="Argument.metadata">=</stringProp>
    <boolProp name="HTTPArgument.use_equals">true</boolProp>
    <stringProp name="Argument.name">body</stringProp>
  </elementProp>
  <elementProp name="userId" elementType="HTTPArgument">
    <boolProp name="HTTPArgument.always_encode">>false</boolProp>
    <stringProp name="Argument.value">1</stringProp>
    <stringProp name="Argument.metadata">=</stringProp>
    <boolProp name="HTTPArgument.use_equals">true</boolProp>
    <stringProp name="Argument.name">userId</stringProp>
  </elementProp>
</collectionProp>
</elementProp>
<stringProp name="HTTPSampler.domain"></stringProp>
<stringProp name="HTTPSampler.port"></stringProp>
<stringProp name="HTTPSampler.protocol"></stringProp>
<stringProp name="HTTPSampler.contentEncoding"></stringProp>
<stringProp name="HTTPSampler.path">/posts/1</stringProp>
<stringProp name="HTTPSampler.method">PUT</stringProp>
<boolProp name="HTTPSampler.follow_redirects">true</boolProp>
<boolProp name="HTTPSampler.auto_redirects">>false</boolProp>
<boolProp name="HTTPSampler.use_keepalive">true</boolProp>
<boolProp name="HTTPSampler.DO_MULTIPART_POST">>false</boolProp>
<stringProp name="HTTPSampler.embedded_url_re"></stringProp>
<stringProp name="HTTPSampler.connect_timeout"></stringProp>
<stringProp name="HTTPSampler.response_timeout"></stringProp>
<stringProp name="TestPlan.comments">
</stringProp>
</HTTPSamplerProxy>
<hashTree>
  <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="PUT| Duration Assertion" enabled="true">
    <stringProp name="DurationAssertion.duration">300</stringProp>
  </DurationAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PUT| Response Assertion (Response Code)" enabled="true">

```

```

    <collectionProp name="Asserion.test_strings">
      <stringProp name="49586">200</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PUT| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="110371416">title</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PUT| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="3029410">body</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
</hashTree>
</hashTree>
  <ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
    <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
    <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
      <boolProp name="LoopController.continue_forever">false</boolProp>
      <stringProp name="LoopController.loops">1</stringProp>
    </elementProp>
    <stringProp name="ThreadGroup.num_threads">10</stringProp>
    <stringProp name="ThreadGroup.ramp_time">5</stringProp>

```

```

    <boolProp name="ThreadGroup.scheduler">false</boolProp>
    <stringProp name="ThreadGroup.duration"></stringProp>
    <stringProp name="ThreadGroup.delay"></stringProp>
    <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
</ThreadGroup>
<hashTree>
  <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
        <collectionProp name="Arguments.arguments"/>
    </elementProp>
    <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
    <stringProp name="HTTPSampler.port"></stringProp>
    <stringProp name="HTTPSampler.protocol">http</stringProp>
    <stringProp name="HTTPSampler.contentEncoding"></stringProp>
    <stringProp name="HTTPSampler.path"></stringProp>
    <stringProp name="HTTPSampler.concurrentPool">6</stringProp>
    <stringProp name="HTTPSampler.connect_timeout"></stringProp>
    <stringProp name="HTTPSampler.response_timeout"></stringProp>
</ConfigTestElement>
</hashTree/>
  <HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="PATCH HTTP Request" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" enabled="true">
        <collectionProp name="Arguments.arguments">
            <elementProp name="body" elementType="HTTPArgument">
                <boolProp name="HTTPArgument.always_encode">false</boolProp>
                <stringProp name="Argument.value">HWPatchbody</stringProp>
                <stringProp name="Argument.metadata">=</stringProp>
                <boolProp name="HTTPArgument.use_equals">true</boolProp>
                <stringProp name="Argument.name">body</stringProp>
            </elementProp>
            <elementProp name="title" elementType="HTTPArgument">
                <boolProp name="HTTPArgument.always_encode">false</boolProp>
                <stringProp name="Argument.value">test3</stringProp>
                <stringProp name="Argument.metadata">=</stringProp>
                <boolProp name="HTTPArgument.use_equals">true</boolProp>
                <stringProp name="Argument.name">title</stringProp>
            </elementProp>
        </collectionProp>
    </elementProp>

```

```

<stringProp name="HTTPSampler.domain"></stringProp>
<stringProp name="HTTPSampler.port"></stringProp>
<stringProp name="HTTPSampler.protocol"></stringProp>
<stringProp name="HTTPSampler.contentEncoding"></stringProp>
<stringProp name="HTTPSampler.path">posts/1</stringProp>
<stringProp name="HTTPSampler.method">PATCH</stringProp>
<boolProp name="HTTPSampler.follow_redirects">true</boolProp>
<boolProp name="HTTPSampler.auto_redirects">false</boolProp>
<boolProp name="HTTPSampler.use_keepalive">true</boolProp>
<boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
<stringProp name="HTTPSampler.embedded_url_re"></stringProp>
<stringProp name="HTTPSampler.connect_timeout"></stringProp>
<stringProp name="HTTPSampler.response_timeout"></stringProp>
</HTTPSamplerProxy>
<hashTree>
  <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="PATCH| Duration Assertion" enabled="true">
    <stringProp name="DurationAssertion.duration">300</stringProp>
  </DurationAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PATCH| Response Assertion (Response Code)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="49586">200</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PATCH| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="110371416">title</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PATCH| Response Assertion (Text Response)" enabled="true">

```

```

    <collectionProp name="Asserion.test_strings">
      <stringProp name="3029410">body</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
</hashTree>
</hashTree>
<ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
  <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
  <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
    <boolProp name="LoopController.continue_forever">false</boolProp>
    <stringProp name="LoopController.loops">1</stringProp>
  </elementProp>
  <stringProp name="ThreadGroup.num_threads">10</stringProp>
  <stringProp name="ThreadGroup.ramp_time">5</stringProp>
  <boolProp name="ThreadGroup.scheduler">false</boolProp>
  <stringProp name="ThreadGroup.duration"></stringProp>
  <stringProp name="ThreadGroup.delay"></stringProp>
  <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
</ThreadGroup>
</hashTree>
<ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
  <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
    <collectionProp name="Arguments.arguments"/>
  </elementProp>
  <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
  <stringProp name="HTTPSampler.port"></stringProp>
  <stringProp name="HTTPSampler.protocol">http</stringProp>
  <stringProp name="HTTPSampler.contentEncoding"></stringProp>
  <stringProp name="HTTPSampler.path"></stringProp>
  <stringProp name="HTTPSampler.concurrentPool">6</stringProp>
  <stringProp name="HTTPSampler.connect_timeout"></stringProp>
  <stringProp name="HTTPSampler.response_timeout"></stringProp>
</ConfigTestElement>

```

```

<hashTree/>
  <HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="DELETE HTTP Request" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" enabled="true">
        <collectionProp name="Arguments.arguments"/>
    </elementProp>
    <stringProp name="HTTPSampler.domain"></stringProp>
    <stringProp name="HTTPSampler.port"></stringProp>
    <stringProp name="HTTPSampler.protocol"></stringProp>
    <stringProp name="HTTPSampler.contentEncoding"></stringProp>
    <stringProp name="HTTPSampler.path">posts/1</stringProp>
    <stringProp name="HTTPSampler.method">DELETE</stringProp>
    <boolProp name="HTTPSampler.follow_redirects">true</boolProp>
    <boolProp name="HTTPSampler.auto_redirects">false</boolProp>
    <boolProp name="HTTPSampler.use_keepalive">true</boolProp>
    <boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
    <stringProp name="HTTPSampler.embedded_url_re"></stringProp>
    <stringProp name="HTTPSampler.connect_timeout"></stringProp>
    <stringProp name="HTTPSampler.response_timeout"></stringProp>
  </HTTPSamplerProxy>
</hashTree>
  <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="DELETE| Duration Assertion" enabled="true">
    <stringProp name="DurationAssertion.duration">300</stringProp>
  </DurationAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="DELETE| Response Assertion (Response Code)" enabled="true">
    <collectionProp name="Asserion.test_strings">
        <stringProp name="49586">200</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="DELETE| Response Assertion (Response Message)" enabled="true">
    <collectionProp name="Asserion.test_strings">
        <stringProp name="2524">OK</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>

```

```

        <stringProp name="Assertion.test_field">Assertion.response_message</stringProp>
        <boolProp name="Assertion.assume_success">false</boolProp>
        <intProp name="Assertion.test_type">2</intProp>
    </ResponseAssertion>
</hashTree/>
    <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="DELETE| Response Assertion (Response Headers)" enabled="true">
        <collectionProp name="Assestion.test_strings">
            <stringProp name="2524">OK</stringProp>
        </collectionProp>
        <stringProp name="Assertion.custom_message"></stringProp>
        <stringProp name="Assertion.test_field">Assertion.response_headers</stringProp>
        <boolProp name="Assertion.assume_success">false</boolProp>
        <intProp name="Assertion.test_type">2</intProp>
    </ResponseAssertion>
</hashTree/>
</hashTree>
</hashTree>
</hashTree>
</hashTree>
</jmeterTestPlan>

```

38099@MILA MINGW64 ~/Desktop

\$ cat Stresstest.jmx

```

<?xml version="1.0" encoding="UTF-8"?>
<jmeterTestPlan version="1.2" properties="5.0" jmeter="5.5">
    <hashTree>
        <TestPlan guiclass="TestPlanGui" testclass="TestPlan" testname="Test Plan"
enabled="true">
            <stringProp name="TestPlan.comments"></stringProp>
            <boolProp name="TestPlan.functional_mode">false</boolProp>
            <boolProp name="TestPlan.tearDown_on_shutdown">true</boolProp>
            <boolProp name="TestPlan.serialize_threadgroups">false</boolProp>
            <elementProp name="TestPlan.user_defined_variables" elementType="Arguments"
guiclass="ArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
                <collectionProp name="Arguments.arguments"/>
            </elementProp>
            <stringProp name="TestPlan.user_define_classpath"></stringProp>
        </TestPlan>
        <hashTree>
            <ResultCollector guiclass="ViewResultsFullVisualizer" testclass="ResultCollector"
testname="View Results Tree" enabled="true">
                <boolProp name="ResultCollector.error_logging">false</boolProp>
            </ResultCollector>
        </hashTree>
    </hashTree>
</jmeterTestPlan>

```



```

<objProp>
  <name>saveConfig</name>
  <value class="SampleSaveConfiguration">
    <time>true</time>
    <latency>true</latency>
    <timestamp>true</timestamp>
    <success>true</success>
    <label>true</label>
    <code>true</code>
    <message>true</message>
    <threadName>true</threadName>
    <dataType>true</dataType>
    <encoding>false</encoding>
    <assertions>true</assertions>
    <subresults>true</subresults>
    <responseData>false</responseData>
    <samplerData>false</samplerData>
    <xml>false</xml>
    <fieldNames>true</fieldNames>
    <responseHeaders>false</responseHeaders>
    <requestHeaders>false</requestHeaders>
    <responseDataOnError>false</responseDataOnError>
    <saveAssertionResultsFailureMessage>true</saveAssertionResultsFailureMessage>
    <assertionsResultsToSave>0</assertionsResultsToSave>
    <bytes>true</bytes>
    <sentBytes>true</sentBytes>
    <url>true</url>
    <threadCounts>true</threadCounts>
    <idleTime>true</idleTime>
    <connectTime>true</connectTime>
  </value>
</objProp>
<stringProp name="filename"></stringProp>
</ResultCollector>
<hashTree/>
<ResultCollector guiclass="SummaryReport" testclass="ResultCollector"
testname="Summary Report" enabled="true">
  <boolProp name="ResultCollector.error_logging">false</boolProp>
  <objProp>
    <name>saveConfig</name>
    <value class="SampleSaveConfiguration">
      <time>true</time>
      <latency>true</latency>
      <timestamp>true</timestamp>

```

```

    <success>true</success>
    <label>true</label>
    <code>true</code>
    <message>true</message>
    <threadName>true</threadName>
    <dataType>true</dataType>
    <encoding>false</encoding>
    <assertions>true</assertions>
    <subresults>true</subresults>
    <responseData>false</responseData>
    <samplerData>false</samplerData>
    <xml>false</xml>
    <fieldNames>true</fieldNames>
    <responseHeaders>false</responseHeaders>
    <requestHeaders>false</requestHeaders>
    <responseDataOnError>false</responseDataOnError>
    <saveAssertionResultsFailureMessage>true</saveAssertionResultsFailureMessage>
    <assertionsResultsToSave>0</assertionsResultsToSave>
    <bytes>true</bytes>
    <sentBytes>true</sentBytes>
    <url>true</url>
    <threadCounts>true</threadCounts>
    <idleTime>true</idleTime>
    <connectTime>true</connectTime>
  </value>
</objProp>
<stringProp name="filename"></stringProp>
</ResultCollector>
<hashTree/>
<ResultCollector guiclass="GraphVisualizer" testclass="ResultCollector" testname="Graph
Results" enabled="true">
  <boolProp name="ResultCollector.error_logging">false</boolProp>
  <objProp>
    <name>saveConfig</name>
    <value class="SampleSaveConfiguration">
      <time>true</time>
      <latency>true</latency>
      <timestamp>true</timestamp>
      <success>true</success>
      <label>true</label>
      <code>true</code>
      <message>true</message>
      <threadName>true</threadName>
      <dataType>true</dataType>

```

```

    <encoding>false</encoding>
    <assertions>true</assertions>
    <subresults>true</subresults>
    <responseData>false</responseData>
    <samplerData>false</samplerData>
    <xml>false</xml>
    <fieldNames>true</fieldNames>
    <responseHeaders>false</responseHeaders>
    <requestHeaders>false</requestHeaders>
    <responseDataOnError>false</responseDataOnError>
    <saveAssertionResultsFailureMessage>true</saveAssertionResultsFailureMessage>
    <assertionsResultsToSave>0</assertionsResultsToSave>
    <bytes>true</bytes>
    <sentBytes>true</sentBytes>
    <url>true</url>
    <threadCounts>true</threadCounts>
    <idleTime>true</idleTime>
    <connectTime>true</connectTime>
  </value>
</objProp>
<stringProp name="filename"></stringProp>
</ResultCollector>
<hashTree/>
  <ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
    <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
    <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
      <boolProp name="LoopController.continue_forever">false</boolProp>
      <stringProp name="LoopController.loops">1</stringProp>
    </elementProp>
    <stringProp name="ThreadGroup.num_threads">50</stringProp>
    <stringProp name="ThreadGroup.ramp_time">35</stringProp>
    <boolProp name="ThreadGroup.scheduler">false</boolProp>
    <stringProp name="ThreadGroup.duration"></stringProp>
    <stringProp name="ThreadGroup.delay"></stringProp>
    <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
  </ThreadGroup>
<hashTree>
  <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">

```

```

    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
    <collectionProp name="Arguments.arguments"/>
</elementProp>
<stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
<stringProp name="HTTPSampler.port"></stringProp>
<stringProp name="HTTPSampler.protocol">http</stringProp>
<stringProp name="HTTPSampler.contentEncoding"></stringProp>
<stringProp name="HTTPSampler.path"></stringProp>
<stringProp name="HTTPSampler.concurrentPool">6</stringProp>
<stringProp name="HTTPSampler.connect_timeout"></stringProp>
<stringProp name="HTTPSampler.response_timeout"></stringProp>
</ConfigTestElement>
<hashTree/>
<HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="GET HTTP Request" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
    <collectionProp name="Arguments.arguments"/>
</elementProp>
<stringProp name="HTTPSampler.domain"></stringProp>
<stringProp name="HTTPSampler.port"></stringProp>
<stringProp name="HTTPSampler.protocol"></stringProp>
<stringProp name="HTTPSampler.contentEncoding"></stringProp>
<stringProp name="HTTPSampler.path">/posts</stringProp>
<stringProp name="HTTPSampler.method">GET</stringProp>
<boolProp name="HTTPSampler.follow_redirects">true</boolProp>
<boolProp name="HTTPSampler.auto_redirects">false</boolProp>
<boolProp name="HTTPSampler.use_keepalive">true</boolProp>
<boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
<stringProp name="HTTPSampler.embedded_url_re"></stringProp>
<stringProp name="HTTPSampler.connect_timeout"></stringProp>
<stringProp name="HTTPSampler.response_timeout"></stringProp>
</HTTPSamplerProxy>
<hashTree>
    <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="GET| Duration Assertion " enabled="true">
    <stringProp name="DurationAssertion.duration">250</stringProp>
</DurationAssertion>
<hashTree/>
    <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="GET| Response Assertion (Response Code)" enabled="true">

```

```

    <collectionProp name="Asserion.test_strings">
      <stringProp name="49586">200</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="GET| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="110371416">title</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="GET| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="3029410">body</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
</hashTree>
</hashTree>
  <ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
    <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
    <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
      <boolProp name="LoopController.continue_forever">false</boolProp>
      <stringProp name="LoopController.loops">1</stringProp>
    </elementProp>
    <stringProp name="ThreadGroup.num_threads">50</stringProp>
    <stringProp name="ThreadGroup.ramp_time">25</stringProp>

```

```

    <boolProp name="ThreadGroup.scheduler">false</boolProp>
    <stringProp name="ThreadGroup.duration"></stringProp>
    <stringProp name="ThreadGroup.delay"></stringProp>
    <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
</ThreadGroup>
<hashTree>
  <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
        <collectionProp name="Arguments.arguments"/>
    </elementProp>
    <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
    <stringProp name="HTTPSampler.port"></stringProp>
    <stringProp name="HTTPSampler.protocol">http</stringProp>
    <stringProp name="HTTPSampler.contentEncoding"></stringProp>
    <stringProp name="HTTPSampler.path"></stringProp>
    <stringProp name="HTTPSampler.concurrentPool">6</stringProp>
    <stringProp name="HTTPSampler.connect_timeout"></stringProp>
    <stringProp name="HTTPSampler.response_timeout"></stringProp>
</ConfigTestElement>
</hashTree>
  <HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="POST HTTP Request" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
        <collectionProp name="Arguments.arguments">
            <elementProp name="title" elementType="HTTPArgument">
                <boolProp name="HTTPArgument.always_encode">false</boolProp>
                <stringProp name="Argument.value">test1</stringProp>
                <stringProp name="Argument.metadata">=</stringProp>
                <boolProp name="HTTPArgument.use_equals">true</boolProp>
                <stringProp name="Argument.name">title</stringProp>
            </elementProp>
            <elementProp name="body" elementType="HTTPArgument">
                <boolProp name="HTTPArgument.always_encode">false</boolProp>
                <stringProp name="Argument.value">test body</stringProp>
                <stringProp name="Argument.metadata">=</stringProp>
                <boolProp name="HTTPArgument.use_equals">true</boolProp>
                <stringProp name="Argument.name">body</stringProp>
            </elementProp>
            <elementProp name="userId" elementType="HTTPArgument">

```

```

    <boolProp name="HTTPArgument.always_encode">false</boolProp>
    <stringProp name="Argument.value">1</stringProp>
    <stringProp name="Argument.metadata">=</stringProp>
    <boolProp name="HTTPArgument.use_equals">true</boolProp>
    <stringProp name="Argument.name">userId</stringProp>
  </elementProp>
</collectionProp>
</elementProp>
<stringProp name="HTTPSampler.domain"></stringProp>
<stringProp name="HTTPSampler.port"></stringProp>
<stringProp name="HTTPSampler.protocol"></stringProp>
<stringProp name="HTTPSampler.contentEncoding">UTF-8</stringProp>
<stringProp name="HTTPSampler.path">/posts</stringProp>
<stringProp name="HTTPSampler.method">POST</stringProp>
<boolProp name="HTTPSampler.follow_redirects">true</boolProp>
<boolProp name="HTTPSampler.auto_redirects">false</boolProp>
<boolProp name="HTTPSampler.use_keepalive">true</boolProp>
<boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
<stringProp name="HTTPSampler.embedded_url_re"></stringProp>
<stringProp name="HTTPSampler.connect_timeout"></stringProp>
<stringProp name="HTTPSampler.response_timeout"></stringProp>
</HTTPSamplerProxy>
<hashTree>
  <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="POST| Duration Assertion " enabled="true">
    <stringProp name="DurationAssertion.duration">300</stringProp>
  </DurationAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="POST| Response Assertion (Response Code)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="49587">201</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="POST| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="110371416">title</stringProp>
    </collectionProp>

```

```

    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="POST| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Assertion.test_strings">
      <stringProp name="3029410">body</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
</hashTree>
</hashTree>
  <ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
    <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
    <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
      <boolProp name="LoopController.continue_forever">false</boolProp>
      <stringProp name="LoopController.loops">1</stringProp>
    </elementProp>
    <stringProp name="ThreadGroup.num_threads">10</stringProp>
    <stringProp name="ThreadGroup.ramp_time">5</stringProp>
    <boolProp name="ThreadGroup.scheduler">false</boolProp>
    <stringProp name="ThreadGroup.duration"></stringProp>
    <stringProp name="ThreadGroup.delay"></stringProp>
    <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
  </ThreadGroup>
</hashTree>
  <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
      <collectionProp name="Arguments.arguments"/>
    </elementProp>
    <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>

```



```

<stringProp name="HTTPSampler.port"></stringProp>
<stringProp name="HTTPSampler.protocol">http</stringProp>
<stringProp name="HTTPSampler.contentEncoding"></stringProp>
<stringProp name="HTTPSampler.path"></stringProp>
<stringProp name="HTTPSampler.concurrentPool">6</stringProp>
<stringProp name="HTTPSampler.connect_timeout"></stringProp>
<stringProp name="HTTPSampler.response_timeout"></stringProp>
</ConfigTestElement>
<hashTree/>
<HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="PUT HTTP Request" enabled="true">
  <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" enabled="true">
    <collectionProp name="Arguments.arguments">
      <elementProp name="title" elementType="HTTPArgument">
        <boolProp name="HTTPArgument.always_encode">false</boolProp>
        <stringProp name="Argument.value">test2</stringProp>
        <stringProp name="Argument.metadata">=</stringProp>
        <boolProp name="HTTPArgument.use_equals">true</boolProp>
        <stringProp name="Argument.name">title</stringProp>
      </elementProp>
      <elementProp name="body" elementType="HTTPArgument">
        <boolProp name="HTTPArgument.always_encode">false</boolProp>
        <stringProp name="Argument.value">HWPutbody</stringProp>
        <stringProp name="Argument.metadata">=</stringProp>
        <boolProp name="HTTPArgument.use_equals">true</boolProp>
        <stringProp name="Argument.name">body</stringProp>
      </elementProp>
      <elementProp name="userId" elementType="HTTPArgument">
        <boolProp name="HTTPArgument.always_encode">false</boolProp>
        <stringProp name="Argument.value">1</stringProp>
        <stringProp name="Argument.metadata">=</stringProp>
        <boolProp name="HTTPArgument.use_equals">true</boolProp>
        <stringProp name="Argument.name">userId</stringProp>
      </elementProp>
    </collectionProp>
  </elementProp>
  <stringProp name="HTTPSampler.domain"></stringProp>
  <stringProp name="HTTPSampler.port"></stringProp>
  <stringProp name="HTTPSampler.protocol"></stringProp>
  <stringProp name="HTTPSampler.contentEncoding"></stringProp>
  <stringProp name="HTTPSampler.path">/posts/1</stringProp>
  <stringProp name="HTTPSampler.method">PUT</stringProp>
  <boolProp name="HTTPSampler.follow_redirects">true</boolProp>

```

```

    <boolProp name="HTTPSampler.auto_redirects">false</boolProp>
    <boolProp name="HTTPSampler.use_keepalive">true</boolProp>
    <boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
    <stringProp name="HTTPSampler.embedded_url_re"></stringProp>
    <stringProp name="HTTPSampler.connect_timeout"></stringProp>
    <stringProp name="HTTPSampler.response_timeout"></stringProp>
    <stringProp name="TestPlan.comments">
</stringProp>
  </HTTPSamplerProxy>
  <hashTree>
    <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="PUT| Duration Assertion" enabled="true">
      <stringProp name="DurationAssertion.duration">300</stringProp>
    </DurationAssertion>
  </hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PUT| Response Assertion (Response Code)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="49586">200</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
  <hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PUT| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="110371416">title</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
  <hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PUT| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="3029410">body</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>

```

```

        <boolProp name="Assertion.assume_success">false</boolProp>
        <intProp name="Assertion.test_type">2</intProp>
    </ResponseAssertion>
    <hashTree/>
</hashTree>
</hashTree>
<ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
    <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
    <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
        <boolProp name="LoopController.continue_forever">false</boolProp>
        <stringProp name="LoopController.loops">1</stringProp>
    </elementProp>
    <stringProp name="ThreadGroup.num_threads">10</stringProp>
    <stringProp name="ThreadGroup.ramp_time">5</stringProp>
    <boolProp name="ThreadGroup.scheduler">false</boolProp>
    <stringProp name="ThreadGroup.duration"></stringProp>
    <stringProp name="ThreadGroup.delay"></stringProp>
    <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
</ThreadGroup>
<hashTree>
    <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
        <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
            <collectionProp name="Arguments.arguments"/>
        </elementProp>
        <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
        <stringProp name="HTTPSampler.port"></stringProp>
        <stringProp name="HTTPSampler.protocol">http</stringProp>
        <stringProp name="HTTPSampler.contentEncoding"></stringProp>
        <stringProp name="HTTPSampler.path"></stringProp>
        <stringProp name="HTTPSampler.concurrentPool">6</stringProp>
        <stringProp name="HTTPSampler.connect_timeout"></stringProp>
        <stringProp name="HTTPSampler.response_timeout"></stringProp>
    </ConfigTestElement>
</hashTree>
<HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="PATCH HTTP Request" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" enabled="true">

```

```

    <collectionProp name="Arguments.arguments">
      <elementProp name="body" elementType="HTTPArgument">
        <boolProp name="HTTPArgument.always_encode">false</boolProp>
        <stringProp name="Argument.value">HWPatchbody</stringProp>
        <stringProp name="Argument.metadata">=</stringProp>
        <boolProp name="HTTPArgument.use_equals">true</boolProp>
        <stringProp name="Argument.name">body</stringProp>
      </elementProp>
      <elementProp name="title" elementType="HTTPArgument">
        <boolProp name="HTTPArgument.always_encode">false</boolProp>
        <stringProp name="Argument.value">test3</stringProp>
        <stringProp name="Argument.metadata">=</stringProp>
        <boolProp name="HTTPArgument.use_equals">true</boolProp>
        <stringProp name="Argument.name">title</stringProp>
      </elementProp>
    </collectionProp>
  </elementProp>
  <stringProp name="HTTPSampler.domain"></stringProp>
  <stringProp name="HTTPSampler.port"></stringProp>
  <stringProp name="HTTPSampler.protocol"></stringProp>
  <stringProp name="HTTPSampler.contentEncoding"></stringProp>
  <stringProp name="HTTPSampler.path">posts/1</stringProp>
  <stringProp name="HTTPSampler.method">PATCH</stringProp>
  <boolProp name="HTTPSampler.follow_redirects">true</boolProp>
  <boolProp name="HTTPSampler.auto_redirects">false</boolProp>
  <boolProp name="HTTPSampler.use_keepalive">true</boolProp>
  <boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
  <stringProp name="HTTPSampler.embedded_url_re"></stringProp>
  <stringProp name="HTTPSampler.connect_timeout"></stringProp>
  <stringProp name="HTTPSampler.response_timeout"></stringProp>
</HTTPSamplerProxy>
<hashTree>
  <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="PATCH| Duration Assertion" enabled="true">
    <stringProp name="DurationAssertion.duration">300</stringProp>
  </DurationAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PATCH| Response Assertion (Response Code)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="49586">200</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>

```

```

    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PATCH| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="110371416">title</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
  <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="PATCH| Response Assertion (Text Response)" enabled="true">
    <collectionProp name="Asserion.test_strings">
      <stringProp name="3029410">body</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_data</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
  </ResponseAssertion>
</hashTree/>
</hashTree>
</hashTree>
  <ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup" testname="Thread
Group" enabled="true">
    <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
    <elementProp name="ThreadGroup.main_controller" elementType="LoopController"
guiclass="LoopControlPanel" testclass="LoopController" testname="Loop Controller"
enabled="true">
      <boolProp name="LoopController.continue_forever">false</boolProp>
      <stringProp name="LoopController.loops">1</stringProp>
    </elementProp>
    <stringProp name="ThreadGroup.num_threads">10</stringProp>
    <stringProp name="ThreadGroup.ramp_time">5</stringProp>
    <boolProp name="ThreadGroup.scheduler">false</boolProp>
    <stringProp name="ThreadGroup.duration"></stringProp>
    <stringProp name="ThreadGroup.delay"></stringProp>
    <boolProp name="ThreadGroup.same_user_on_next_iteration">true</boolProp>
  </ThreadGroup>

```

```

<hashTree>
  <ConfigTestElement guiclass="HttpDefaultsGui" testclass="ConfigTestElement"
testname="HTTP Request Defaults" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" testname="User Defined Variables"
enabled="true">
        <collectionProp name="Arguments.arguments"/>
    </elementProp>
    <stringProp name="HTTPSampler.domain">jsonplaceholder.typicode.com</stringProp>
    <stringProp name="HTTPSampler.port"></stringProp>
    <stringProp name="HTTPSampler.protocol">http</stringProp>
    <stringProp name="HTTPSampler.contentEncoding"></stringProp>
    <stringProp name="HTTPSampler.path"></stringProp>
    <stringProp name="HTTPSampler.concurrentPool">6</stringProp>
    <stringProp name="HTTPSampler.connect_timeout"></stringProp>
    <stringProp name="HTTPSampler.response_timeout"></stringProp>
  </ConfigTestElement>
</hashTree>
  <HTTPSamplerProxy guiclass="HttpTestSampleGui" testclass="HTTPSamplerProxy"
testname="DELETE HTTP Request" enabled="true">
    <elementProp name="HTTPSampler.Arguments" elementType="Arguments"
guiclass="HTTPArgumentsPanel" testclass="Arguments" enabled="true">
        <collectionProp name="Arguments.arguments"/>
    </elementProp>
    <stringProp name="HTTPSampler.domain"></stringProp>
    <stringProp name="HTTPSampler.port"></stringProp>
    <stringProp name="HTTPSampler.protocol"></stringProp>
    <stringProp name="HTTPSampler.contentEncoding"></stringProp>
    <stringProp name="HTTPSampler.path">posts/1</stringProp>
    <stringProp name="HTTPSampler.method">DELETE</stringProp>
    <boolProp name="HTTPSampler.follow_redirects">true</boolProp>
    <boolProp name="HTTPSampler.auto_redirects">false</boolProp>
    <boolProp name="HTTPSampler.use_keepalive">true</boolProp>
    <boolProp name="HTTPSampler.DO_MULTIPART_POST">false</boolProp>
    <stringProp name="HTTPSampler.embedded_url_re"></stringProp>
    <stringProp name="HTTPSampler.connect_timeout"></stringProp>
    <stringProp name="HTTPSampler.response_timeout"></stringProp>
  </HTTPSamplerProxy>
</hashTree>
  <DurationAssertion guiclass="DurationAssertionGui" testclass="DurationAssertion"
testname="DELETE| Duration Assertion" enabled="true">
    <stringProp name="DurationAssertion.duration">300</stringProp>
  </DurationAssertion>
</hashTree>

```

```

    <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="DELETE| Response Assertion (Response Code)" enabled="true">
    <collectionProp name="Assertion.test_strings">
        <stringProp name="49586">200</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_code</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
</ResponseAssertion>
<hashTree/>
    <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="DELETE| Response Assertion (Response Message)" enabled="true">
    <collectionProp name="Assertion.test_strings">
        <stringProp name="2524">OK</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_message</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
</ResponseAssertion>
<hashTree/>
    <ResponseAssertion guiclass="AssertionGui" testclass="ResponseAssertion"
testname="DELETE| Response Assertion (Response Headers)" enabled="true">
    <collectionProp name="Assertion.test_strings">
        <stringProp name="2524">OK</stringProp>
    </collectionProp>
    <stringProp name="Assertion.custom_message"></stringProp>
    <stringProp name="Assertion.test_field">Assertion.response_headers</stringProp>
    <boolProp name="Assertion.assume_success">false</boolProp>
    <intProp name="Assertion.test_type">2</intProp>
</ResponseAssertion>
<hashTree/>
</hashTree>
</hashTree>
</hashTree>
</jmeterTestPlan>

```