# Java中HTTP请求的两种方式

package cn.gyyx.manage.client;

import java.net.HttpURLConnection;

import java.net.URL;

import org.apache.http.HttpResponse;

import org.apache.http.client.HttpClient;

import org.apache.http.client.methods.HttpGet;

import org.apache.http.impl.client.DefaultHttpClient;

import org.apache.log4j.Logger;

import org.apache.log4j.PropertyConfigurator;

public class Test {

private static Logger logger=Logger.getLogger(Test.class);

private static String urlPath = "http://192.168.6.107:8080/test/tologin";

// private static String urlPath = "http://localhost:8080/ui/getAll.do";

private static int count=2000;

public static long url() {

long startTime = System.currentTimeMillis();

for (int i = 0; i < count; i++) {

new Thread() {

@Override

public void run() {

try {

URL url = new URL(urlPath);

HttpURLConnection httpURLConnection = (HttpURLConnection) url

.openConnection();

int statusCode=httpURLConnection.getResponseCode();

logger.debug(statusCode);

// System.out.println(httpURLConnection.getResponseCode());

} catch (Exception e) {

e.printStackTrace();

}

};

}.start();

}

long endTime = System.currentTimeMillis();

return endTime - startTime;

}

public static long httpClient(){

long startTime = System.currentTimeMillis();

for (int i = 0; i < count; i++) {

new Thread() {

@Override

public void run() {

try {

HttpClient httpClient=new DefaultHttpClient();

HttpGet httpGet=new HttpGet(urlPath);

HttpResponse httpResponse=httpClient.execute(httpGet);

String status=httpResponse.getStatusLine().toString();

logger.debug(status);

// System.out.println(httpResponse.getStatusLine());

} catch (Exception e) {

e.printStackTrace();

}

};

}.start();

}

long endTime = System.currentTimeMillis();

return endTime - startTime;

}

public static void main(String[] args) {

// PropertyConfigurator.configure("log4j-config.xml");

System.out.println("URL请求"+count+"次所花时间为："+url());//超过一定的线程数程序报错，速度快，客户占用内存小

System.out.println("HttpClient请求"+count+"次所花时间为："+httpClient());//占客户端内存极大，线程安全，服务端内存占用不大

}

}