Харьковский национальный университет им. В.Н. Каразина

Факультет компьютерных наук

# Лабораторная работа №12

По учебной дисциплине

**Математические методы и технологии тестирования и верификации программного обеспечения**

«Тестирование API»

|  |  |
| --- | --- |
|  | Выполнил:  Студент группы КС-22  Филатов Виталий Витальевич  Проверил:  Доц. Малахов Сергей Витальевич |

**Тема:** Тестирование API.

**Цель работы:** Изучить тестирование API, написать запросы к веб-ресурсу.

**ВЫПОЛНЕНИЕ ИНДИВИДУАЛЬНОГО ЗАДАНИЯ**

**public class** TestHttpConnection {  
  
 @Test  
 **public void** http\_status\_code\_is\_200() **throws** IOException {  
 HttpURLConnection connection;  
 String address = **"http://localhost"**;  
 connection = (HttpURLConnection) **new** URL(address).openConnection();  
 connection.setRequestMethod(**"GET"**);  
 connection.setUseCaches(**false**);  
 connection.setConnectTimeout(2500);  
 connection.setReadTimeout(2500);  
 connection.connect();  
  
 Assert.*assertEquals*(HttpURLConnection.***HTTP\_OK***, connection.getResponseCode());  
 }  
  
 @Test  
 **public void** http\_post\_request() **throws** IOException {  
 String parameters = **"x=3&y=4151&operation=%2B"**;  
 HttpURLConnection connection;  
 String address = **"http://localhost/calculate"**;  
 connection = (HttpURLConnection) **new** URL(address).openConnection();  
 connection.setRequestMethod(**"POST"**);  
 connection.setUseCaches(**false**);  
 connection.setConnectTimeout(2500);  
 connection.setReadTimeout(2500);  
  
 connection.setDoOutput(**true**);  
 OutputStream outputStream = connection.getOutputStream();  
 outputStream.write(parameters.getBytes());  
 outputStream.flush();  
 outputStream.close();  
  
 InputStream inputStream = connection.getInputStream();  
 BufferedReader reader = **new** BufferedReader(**new** InputStreamReader(inputStream));  
 String line;  
  
 **while** ((line = reader.readLine()) != **null**) {  
 System.***out***.println(line);  
 }  
  
 Assert.*assertEquals*(HttpURLConnection.***HTTP\_OK***, connection.getResponseCode());  
 }  
}

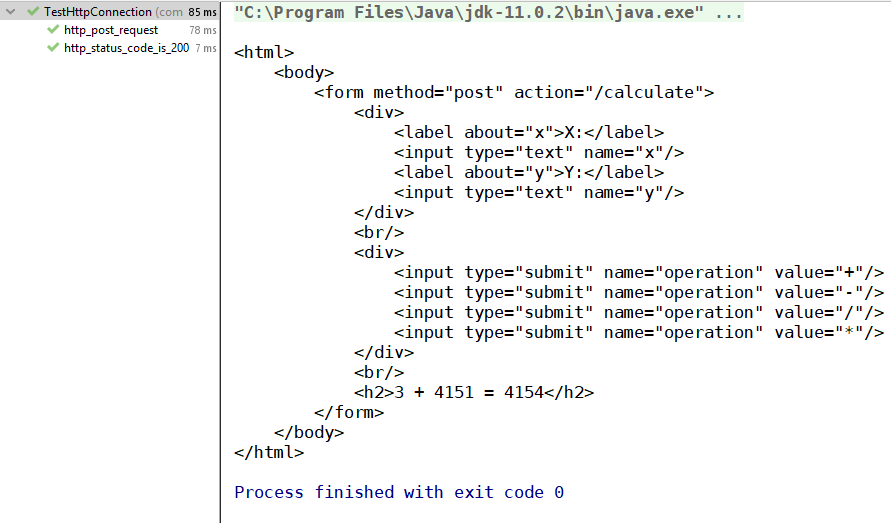


Рисунок 1 – успешное прохождение тестов

**ВЫВОДЫ**

Входе данной работы, я научился писать HTPP запросы (GET, POST). Были написаны два теста. Первый отправляет GET запрос на web приложение, разработанное для лабораторной работы №11 и проверяет что Status code равен 200. Второй тест отправляет POST запрос с параметрами «x=3», «y=4151», «operation=%2B». «%2B» соответствует символу «+». В результате отправки данного POST запроса нас перебрасывает на страницу, где виден результат сложения «x» и «y». Также проверяется чтобы Status code был 200.