

Практическое занятие. Тестирование API

Интеграционное тестирование в Postman

Цель: получить навыки интеграционного тестирования в **Postman**

Задача: изучить теорию, **выполнить задания** и **ответить на вопросы**.

Postman — инструмент, который помогает формировать запросы и анализировать ответы со стороны сервера. Он позволяет сформировать и отправить HTTP-запросы, чтобы протестировать API.

Тестовые данные:

Сайт — <http://users.bugred.ru/>

Дополнительной авторизации нет.

ТЗ — техническое задание на Users

<http://okiseleva.blogspot.com/2017/04/users-soap-rest.html>

Полная документация (описание методов, форматов данных и т.д)

<https://testbase.atlassian.net/wiki/spaces/USERS/overview>

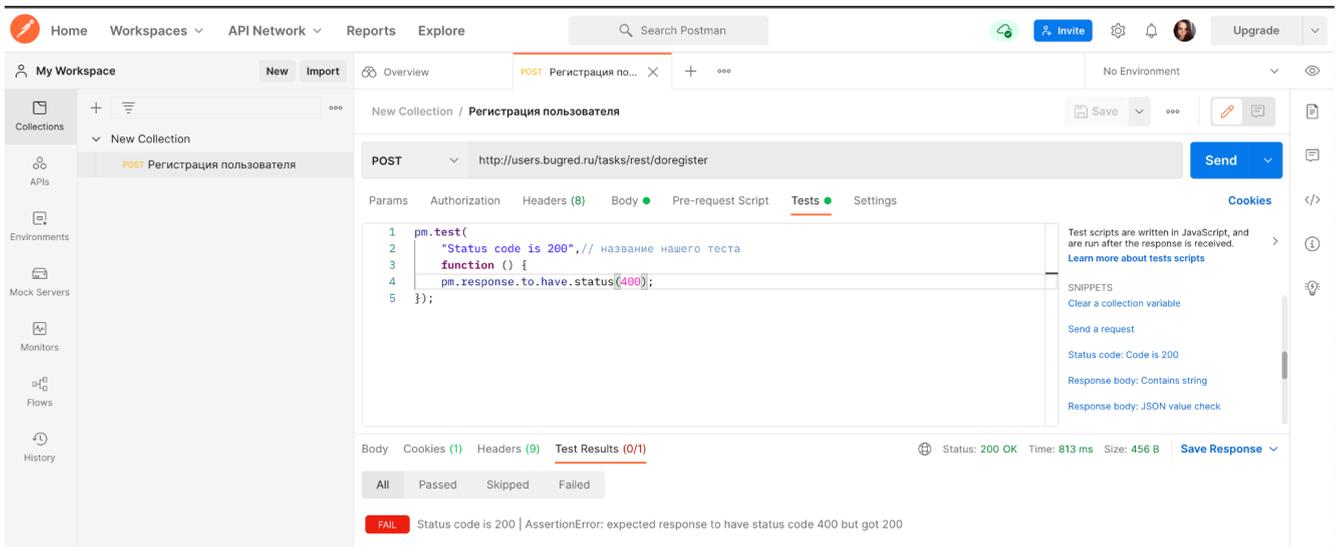
1. Ознакомиться с системой Users — бесплатный тестовый проект с багами и методами SOAP / REST.
2. Создать нового пользователя по методу `doRegister` для этого нужно:
 - Создать новый запрос и указать метод POST и URL запроса (`doRegister`), который находится в документации.
 - В разделе Body выберите raw, формат JSON, и введите данные для создания пользователя (формат для ввода данных находится в документации)
 - Отправьте запрос и убедитесь, что пользователь успешно создан (статус 200)
3. Написать автотест на проверку статус кода 200, используя Snippets

The screenshot shows the Postman interface for a REST client request. The request is a POST to `http://users.bugred.ru/tasks/rest/doregister`. The test script is as follows:

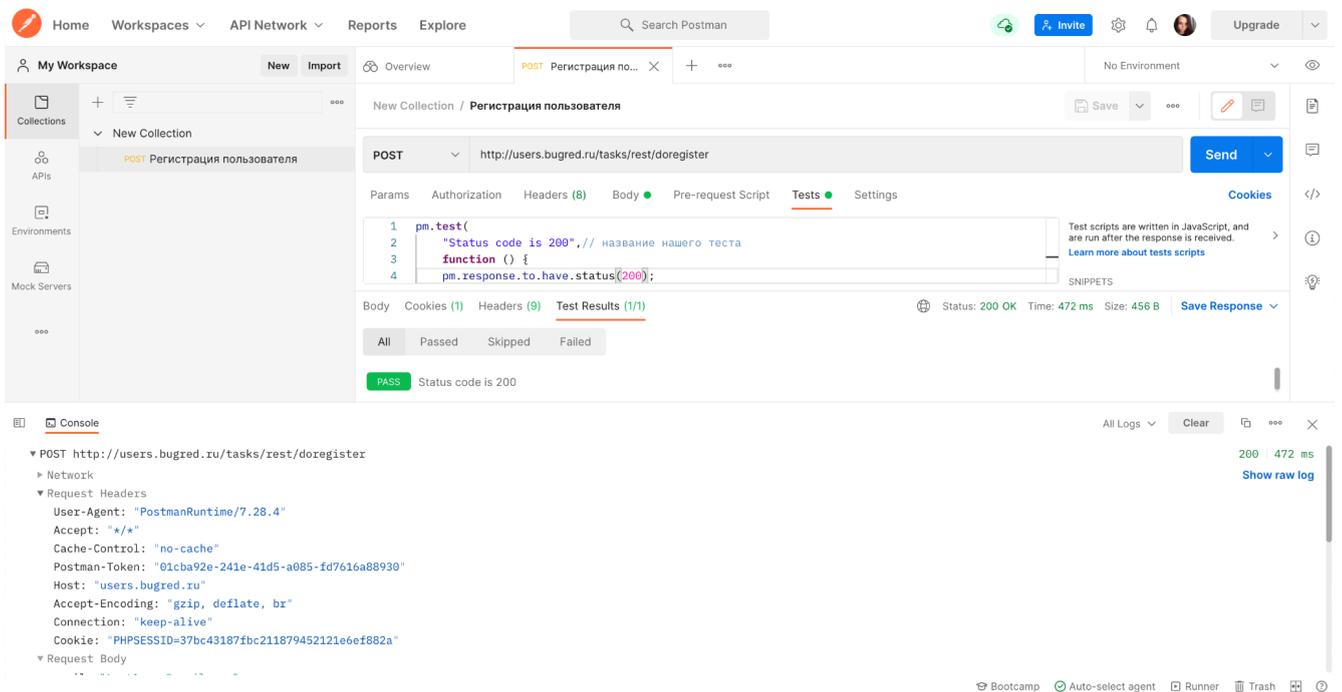
```
1 pm.test(
2   "Status code is 200", // название нашего теста
3   function () {
4     pm.response.to.have.status(200);
5   });
```

The test results show a status of 200 OK, with a time of 557 ms and a size of 628 B. The test passed, with the message "PASS: Status code is 200".

4. Проверить свой автотест, сломав его, таким образом убеждаемся, что автотест работает. Замените статус код 200 на 400.



5. Откройте консоль в Postman (Ctrl+Alt+C). Когда мы пишем автотесты консоль помогает нам в качестве отладчика.



6. Для того, чтобы что-то вывести на консоль необходимо набрать команду `console.log`

7. Выведем на консоль подробную информацию о `pm.response`.

8. При запуске теста мы получаем дополнительную информацию об объекте `pm.response`, у которого есть какие-то свойства. Например статус код.

9. Попробуем вывести его (статус код 200), используя точечную запись (строка 7). И в консоли мы получили дополнительную запись 200.

10. Прочсть подробнее об объектах в Postman можно в [документации](#)

11. Самостоятельно из документации изучите объект `pm.request`, и добавьте в ваш автотест. **Прокомментируйте полученную информацию в консоли.**

Самостоятельная работа:

MagicSearch: проверяем статус коды

Открываем документацию по Users, находим метод MagicSearch.

Ваша задача — написать автотесты на блок «Статусы ответов» для метода MagicSearch. Статусы 200 и 400 тестировать не надо, они не должны встречаться.

Для каждого статуса (230, 231, 232, 233, 234, 235, 455, 456, 457, 458, 459) создайте два теста:

Обычный тест: проверяет, что сервер возвращает ожидаемый статус

Сломанный тест: для проверки того, что тест действительно что-то делает

В отчете должны присутствовать скриншот на каждый статус — на нем должны быть видны оба теста (pass + fail) и результат их прогона.

Помните, что это уже реальные автотесты — так что давайте им осмысленные имена, а не просто 11 тестов с копиями из сниппета "проверка кода 200".

- 230 — никого не нашли по запросу, пустой ответ
- 231 — найден 1 юзер
- 232 — найдено больше 1 юзера (но без компаний)
- 233 — найдена 1 компания
- 234 — найдено больше 1 компании (но без юзеров)
- 235 — найдены как юзеры, так и компании

Ошибки на клиенте:

- 455 — не указан параметр query в запросе. Body сообщения — “Не найден обязательный параметр query”
- 456 — длина запроса выше 1000 символов. Текст ошибки — “Длина запроса не должна превышать 1000 символов”
- 457 — неправильно задан partyType (значение не из доступного списка выбора). Текст ошибки — “Параметр partyType может принимать только значения: ALL, USER, COMPANY”
- 458 — неправильно задан параметр taskStatus (не из списка выбора значение). Текст ошибки — “Параметр taskStatus может принимать только значения: ALL, ACTUAL, COMPLETE, FAIL”

- 459 — неправильно задан параметр include (не из списка выбора значение). Текст ошибки — “Параметр include может принимать только значения: ALL, USER, COMPANY, TASK, WHY”

Критерии оценивания:

- **Оценка "5" (отлично):**

- Все HTTP-запросы сформированы и отправлены корректно.
- Все ответы сервера проанализированы точно и в полном объеме.
- Все автоматизированные тесты (обычные и "сломанные") работоспособны, корректно проверяют статус-коды и осмысленно названы.
- Достигнуто полное тестовое покрытие согласно заданию (все статусы проверены).
- Отчет оформлен в соответствии с требованиями, содержит все необходимые скриншоты с видимыми результатами прогона тестов (pass + fail) и четкие пояснения.
- Продемонстрировано глубокое понимание принципов интеграционного тестирования и уверенное владение Postman.

- **Оценка "4" (хорошо):**

- Большинство HTTP-запросов сформированы и отправлены корректно, возможно, с незначительными ошибками, не влияющими на общий результат.
- Анализ ответов сервера в основном точен, с небольшими неточностями, которые не привели к существенным ошибкам в тестах.
- Большинство автоматизированных тестов работоспособны и корректны, возможно, с небольшими недочетами в наименовании или структуре.
- Тестовое покрытие почти полное, возможно, пропущен 1-2 статуса или не все аспекты задания учтены, но основные требования выполнены.
- Отчет оформлен достаточно хорошо, содержит большинство необходимых скриншотов и пояснений, но может иметь незначительные недостатки в оформлении.
- Продемонстрировано хорошее понимание интеграционного тестирования и умение работать с Postman.

- **Оценка "3" (удовлетворительно):**

- Часть HTTP-запросов содержит ошибки, которые требуют корректировки.
- Анализ ответов сервера содержит заметные неточности или неполноту, что привело к ошибкам в тестах.
- Часть автоматизированных тестов неработоспособна или некорректна, или отсутствуют "сломанные" тесты для части статусов.
- Тестовое покрытие неполное, пропущены несколько статусов или значительные части задания.
- Отчет оформлен небрежно, отсутствуют некоторые скриншоты или пояснения, затрудняющие проверку.
- Продемонстрировано базовое понимание интеграционного тестирования и Postman, но с существенными пробелами в практических навыках.