

API. Домашнее задание для тестировщиков в ivi

Рекомендуемый стек для автотестов: Python+Pytest (но можно другое, главное - выполнить задачу).

Есть два способа выполнить это задание. В обоих случаях результат должен быть в виде репозитория на github или bitbucket.

Вариант 1

1. Реализовать наборы тестовых сценариев для команд; Результат: файл с тестовыми сценариями
2. В отчете описать запросы к API (в идеале с помощью утилиты curl, можно использовать curl | jq "); Результат: файл с тестовыми сценариями и запросами к API
3. Написать скрипт, автоматизирующий тестовые сценарии, обязательно с комментариями в коде (лучше: docstring); Результат: файл с тестовыми сценариями + файл .py с кодом

Вариант 2

1. Реализовать наборы тестовых сценариев для команд; Результат: файл с тестовыми сценариями
2. Написать фреймворк с автотестами;

Результат:

- приватный проект в github/bitbucket с тестовыми сценариями, readme.md, исходниками проекта
- доступ к проекту для пользователя qa-hub(для проекта на github) или qa_github@ivi.ru(для проекта на bitbucket)
- логина и пароля от тестового сервиса в репозитории быть не должно, их можно передавать через параметры запуска или переменные окружения

Документация

Аутентификация

Для работы с API необходима аутентификация. Для этого используется механизм HTTP Basic Authentication. Логин и пароль необходимо передавать в каждый запрос.

```
curl 'http://rest.test.ivi.ru/v2/characters' -u username:password
```

Команды API

GET /characters

```
curl 'http://rest.test.ivi.ru/v2/characters' -u username:password
```

```
{
  "result": [
    {
      "education": "Unrevealed",
      "height": 170,
      "identity": "Secret (known to the U.S. government)",
      "name": "Avalanche",
    }
  ]
}
```

```
    "other_aliases": "Jon Bloom",
    "universe": "Marvel Universe",
    "weight": 87.75
  },
  {
    "name": "Ben Parker",
    "other_aliases": "Ben, Uncle Ben, Unca' Ben"
  }
]
```

GET /character?name=Hero+Name

```
curl 'http://rest.test.ivu.ru/v2/character?name=Avalanche' -u username:password
```

Для указания имени, состоящего из нескольких слов, между словами нужно ставить знак +

```
{
  "result": {
    "education": "Unrevealed",
    "height": 170,
    "identity": "Secret (known to the U.S. government)",
    "name": "Avalanche",
    "other_aliases": "Jon Bloom",
    "universe": "Marvel Universe",
    "weight": 87.75
  }
}
```

POST /character

```
curl -X POST -H 'Content-type: application/json'
-u username:password
-d '{"name": "Hawkeye", "universe": "Marvel Universe",
  "education": "High school (unfinished)", "weight": 104,
  "height": 1.90, "identity": "Publicly known"}'
'http://rest.test.ivu.ru/v2/character'
```

```
{
  "result": {
    "education": "High school (unfinished)",
    "height": 1.9,
    "identity": "Publicly known",
    "name": "Hawkeye",
    "universe": "Marvel Universe",
    "weight": 104
  }
}
```

БД имеет ограничение на количество персонажей. Если база заполнена, то при запросе на добавление возникнет ошибка.

```
{
  "error": "Collection can't contain more than 500 items"
}
```

Если в теле запроса не указать необходимые поля, либо использовать неверный тип для поля, то возникнет ошибка.

```
{
  "error": "name: ['Missing data for required field.'], weight: ['Not a valid number.']"
}
```

PUT /character

```
curl -X PUT -H 'Content-type: application/json'
-u username:password
-d '{"name": "Hawkeye", "universe": "Marvel Universe",
  "education": "High School (unfinished)", "weight": 104,
  "height": 1.90, "identity": "Publicly known"}'
'http://rest.test.ivu.ru/v2/character'
```

```
{
  "result": [
    {
      "education": "High School (unfinished)",
      "height": 1.9,
      "identity": "Publicly known",
      "name": "Hawkeye",
      "universe": "Marvel Universe",
      "weight": 104
    }
  ]
}
```

DELETE /character?name=Hero+Name

```
curl -X DELETE 'http://rest.test.ivu.ru/v2/character?name=Abyss' -u username:password
```

Для указания имени, состоящего из нескольких слов, между словами нужно ставить знак +

```
{
  "result": [
    "Hero Abyss is deleted"
  ]
}
```

Если по паве невозможно найти персонажа, то возвращается ошибка.

```
curl -X DELETE 'http://rest.test.ivu.ru/v2/character?name=Vovan' -u username:password
```

```
{
  "error": "No such name"
}
```

Сброс коллекции в первоначальное состояние

```
curl -X POST 'http://rest.test.ivu.ru/v2/reset' -u username:password
```