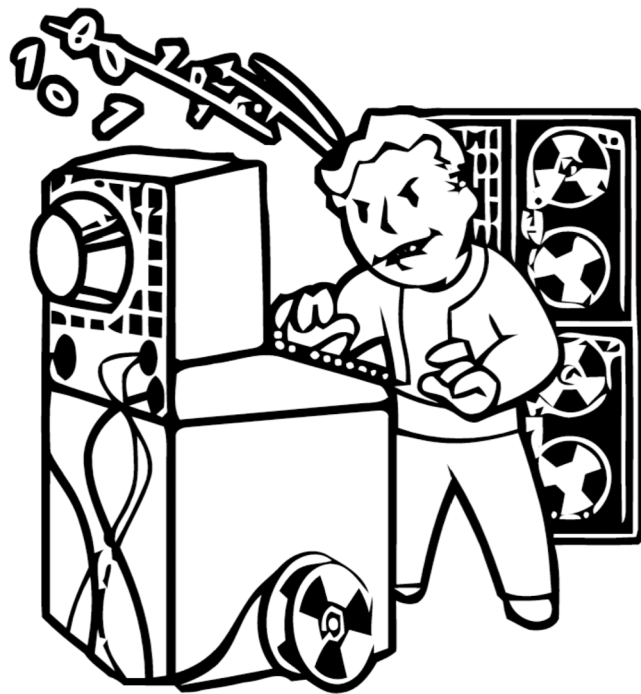
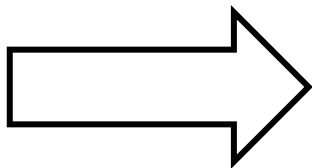
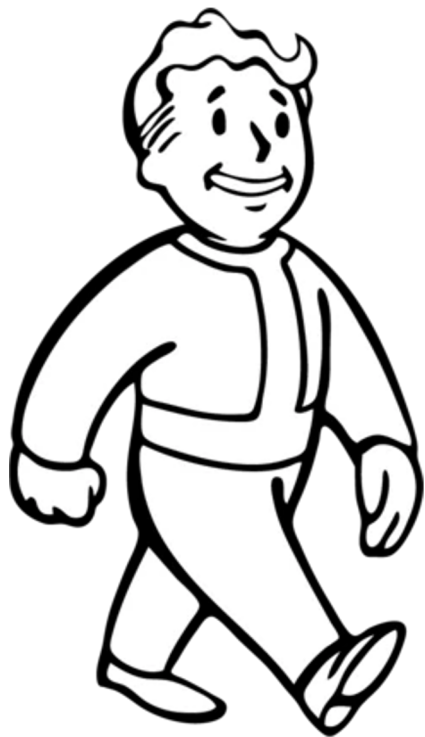


Трансформация из Manual QA в Automation

Heisenbug 2021, 6-9 апреля

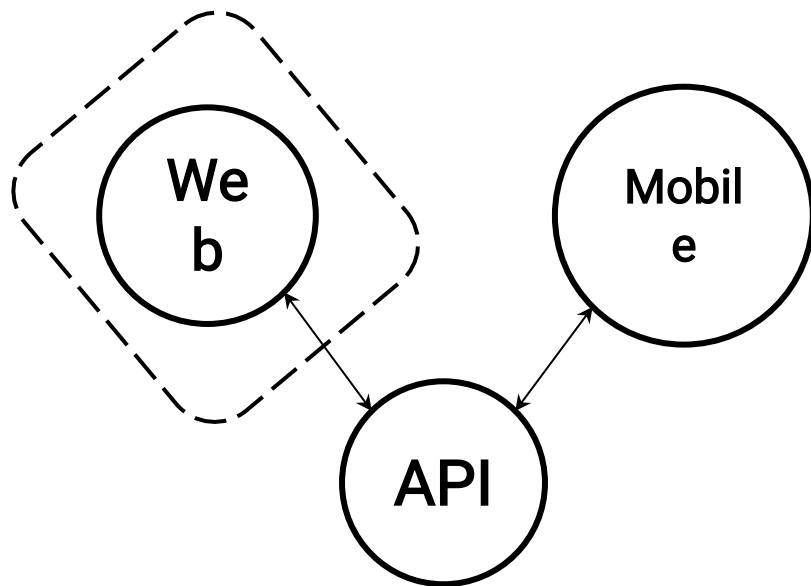


Об авторе

1. Фамилия, имя: **Бормотов Дмитрий**
2. Компания: **OnTarget Group, Inc.**
3. Позиция: **Senior QA Engineer**
4. Landing page: **www.autoqa.me**



План



План

1. Проблемы трансформации из Manual в Automation.
2. Минимум для входа в автоматизацию.
3. Автоматизация на ранних этапах.
4. Селекторы.
5. Выбор языка программирования.
6. Изучение автоматизации.
7. Дальнейшие шаги.



Проблемы трансформации из Manual в Automation

Проблемы трансформации из Manual в Automation

1. Первые шаги
 - а. Что изучать?
 - б. Выбор стека технологий
2. Откуда черпать информацию?
3. Я готов?
4. Что делать, когда я готов?



Code vs Record'n'Play

Первые шаги

Первые шаги / Code vs Record'n'Play

Record'n'Play

- + скорость
- + код не нужен
- гибкость настройки
- только простые сценарии

Code

- + гибкость настройки
- + сценарии любой сложности
- умение писать код
- интеграция с CI / запуск по кнопке



TestNG

JUnit

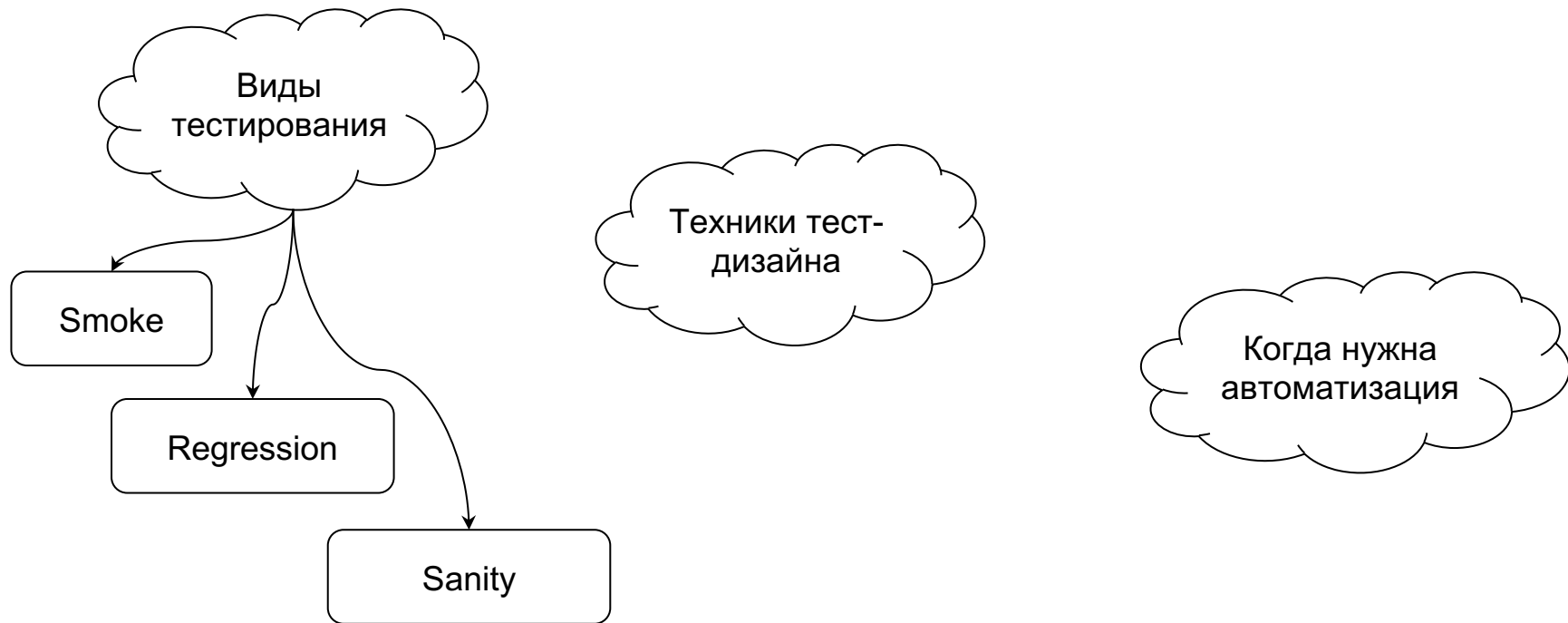
Первые шаги

Минимум для входа в автоматизацию

Минимум для входа в автоматизацию

Frontend + Backend





Минимум для входа в автоматизацию / Теория тестирования

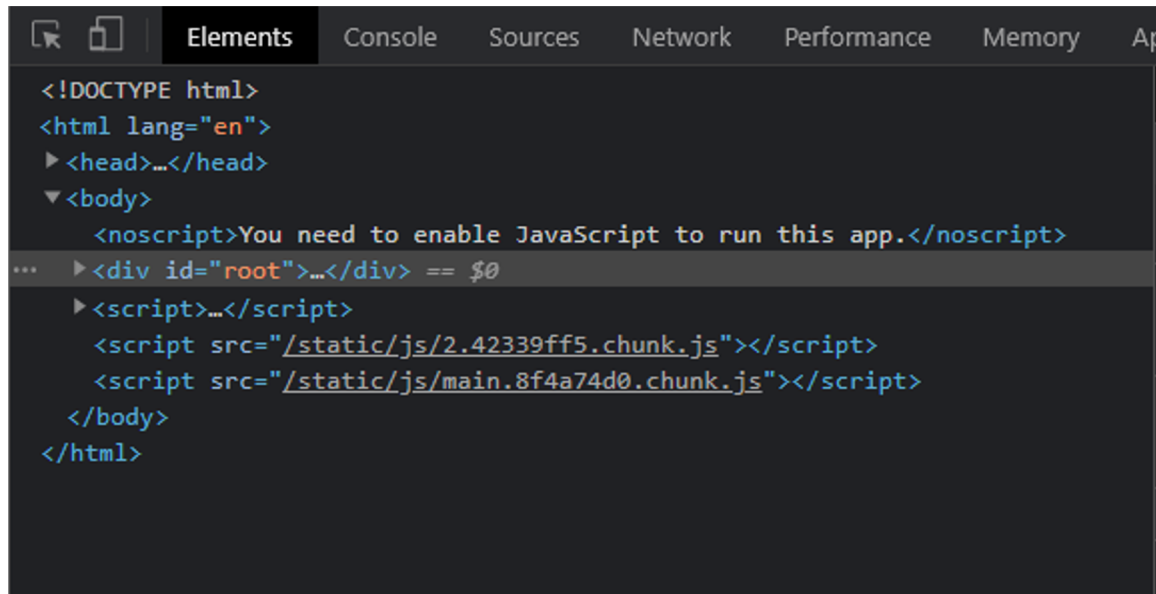
Минимум для входа / Chrome DevTools

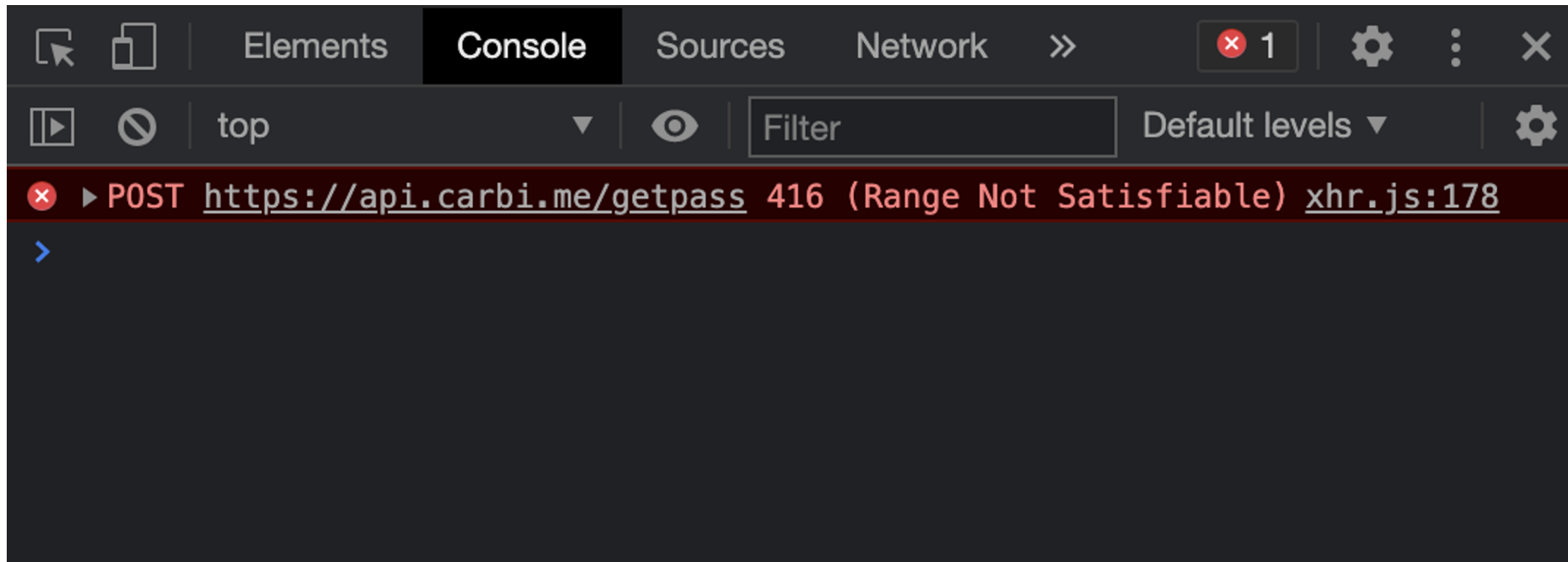
1. Console

- а. дебаг
- б. скрипты

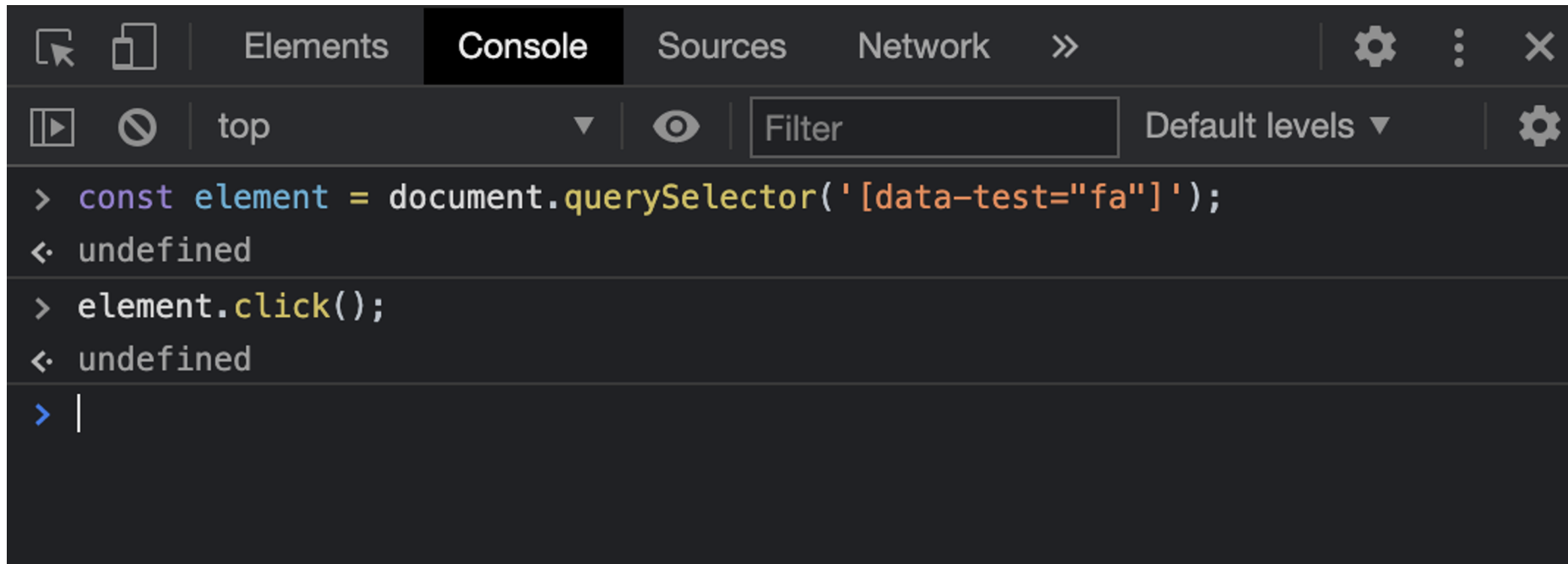
2. Network

3. Elements





Console / Debug



Console / Scripts

The screenshot shows the Chrome DevTools Network tab. The top bar includes tabs for Elements, Console, Sources, and Network. The Network tab is active, displaying a list of requests. The 'getpass' request is selected, and its details are shown in the right pane. The response is a JSON object indicating success.

Elements Console Sources **Network** >>

☐ Preserve log ☐ Disable cache No throttling ☐ Has blocked cookies

Filter ☐ Hide data URLs

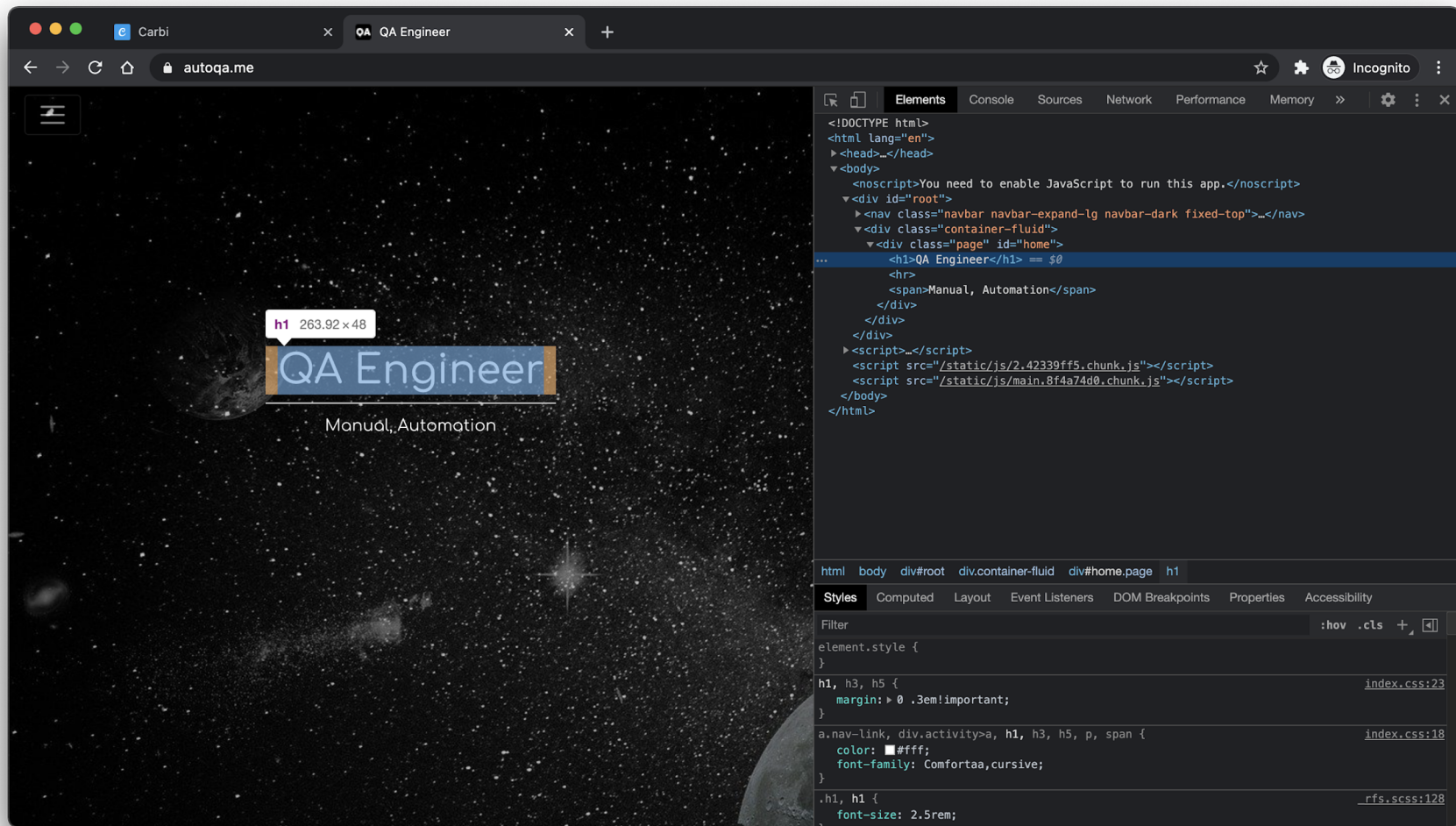
All **XHR** JS CSS Img Media Font Doc WS Manifest Other ☐ Blocked Requests

☐ Use large request rows ☐ Group by frame

☐ Show overview ☐ Capture screenshots

Name	× Headers Preview Response Initiator >>
getpass	<div>▼ {code: "Success", message: "Password sent"} code: "Success" message: "Password sent"</div>

Network



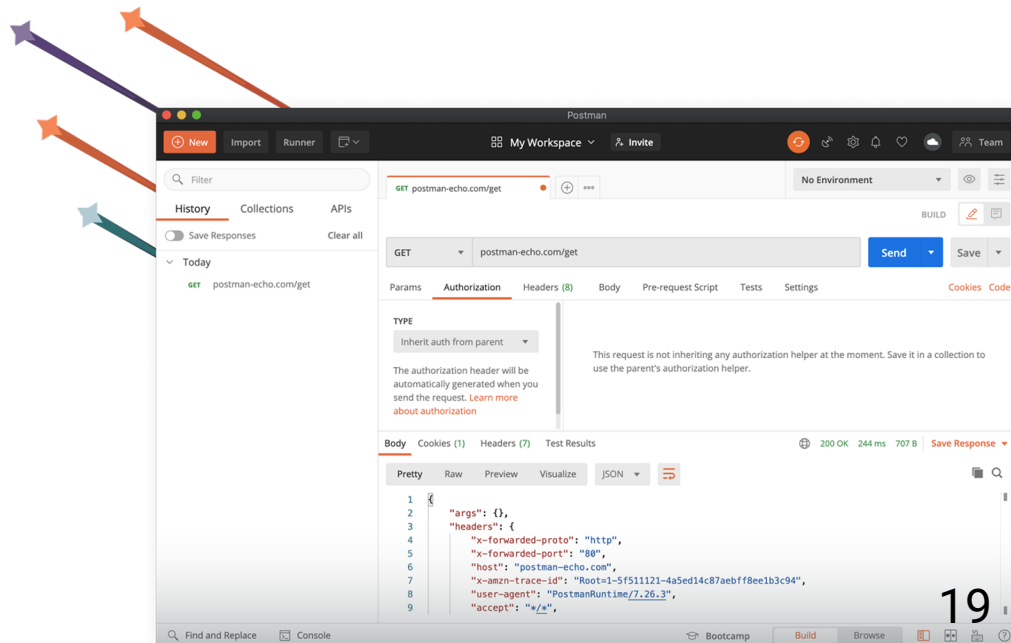
Минимум для входа в автоматизацию / Postman

1. CRUD (POST, GET, PUT/PATCH, DELETE)

2. Headers

а. Токен авторизации

3. Переменные окружения



Автоматизация на ранних этапах

Автоматизация на ранних этапах

1. Рутинные скрипты

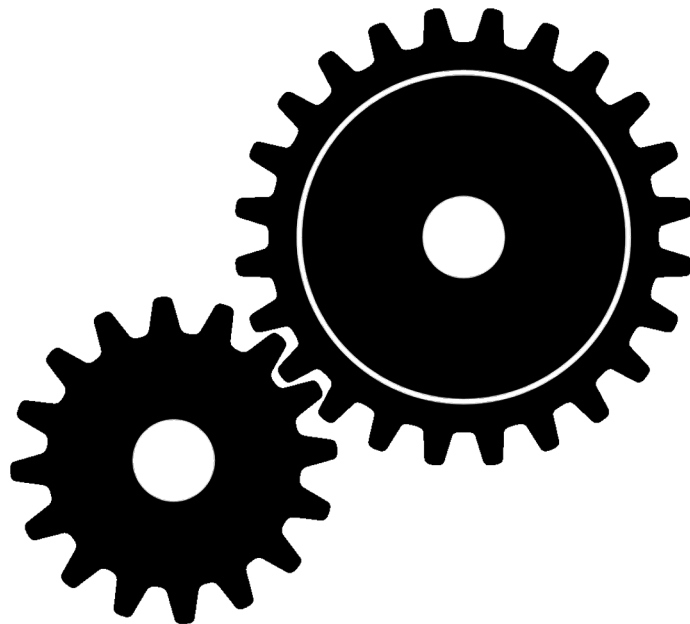
- a. Скрапперы (сборщики) данных
- b. Подготовка тестовых данных

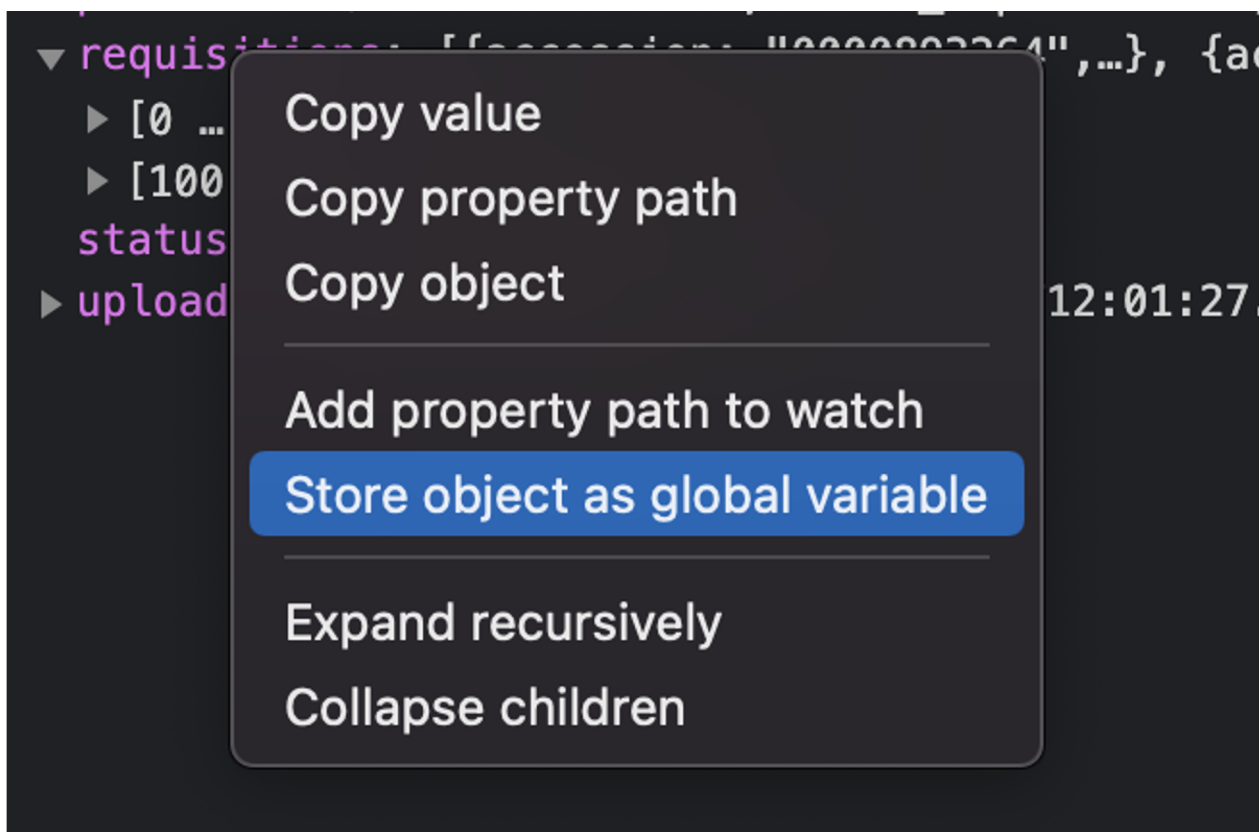
2. Простые UI тесты

- a. Selenium IDE
- b. Kalaton Studio

3. API тесты

- a. Postman
- b. SoapUI



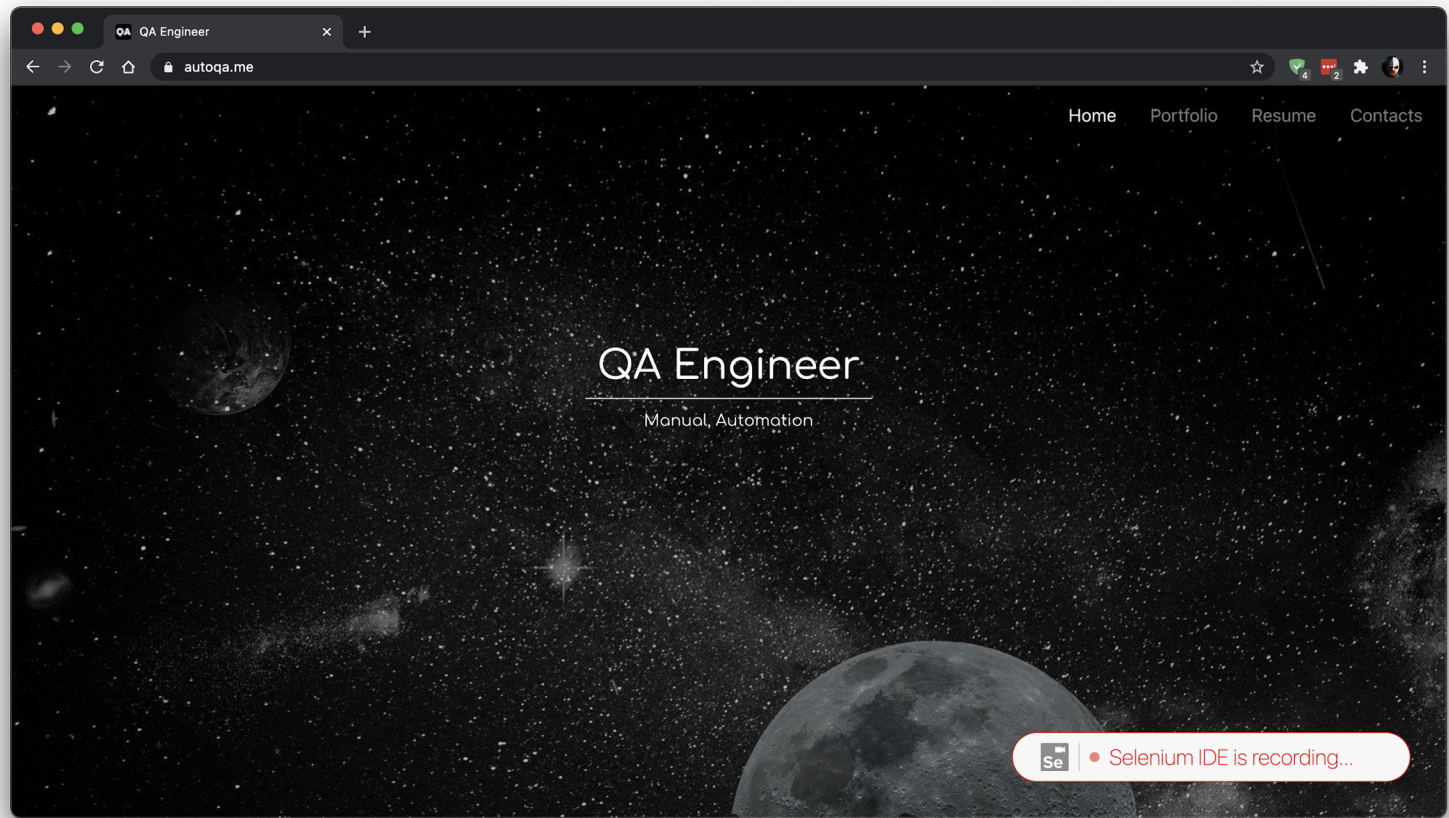


Рутинные скрипты

```
> temp1
```

```
< (200) [{...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...},  
      {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...},  
      {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...},  
      {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...},  
      {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...}, {...},  
      {...}, {...}, {...}, {...}, {...}, {...}, {...}, ...]
```

```
> copy(temp1.map(({ accession }) => accession).join(','));|
```



Selenium IDE



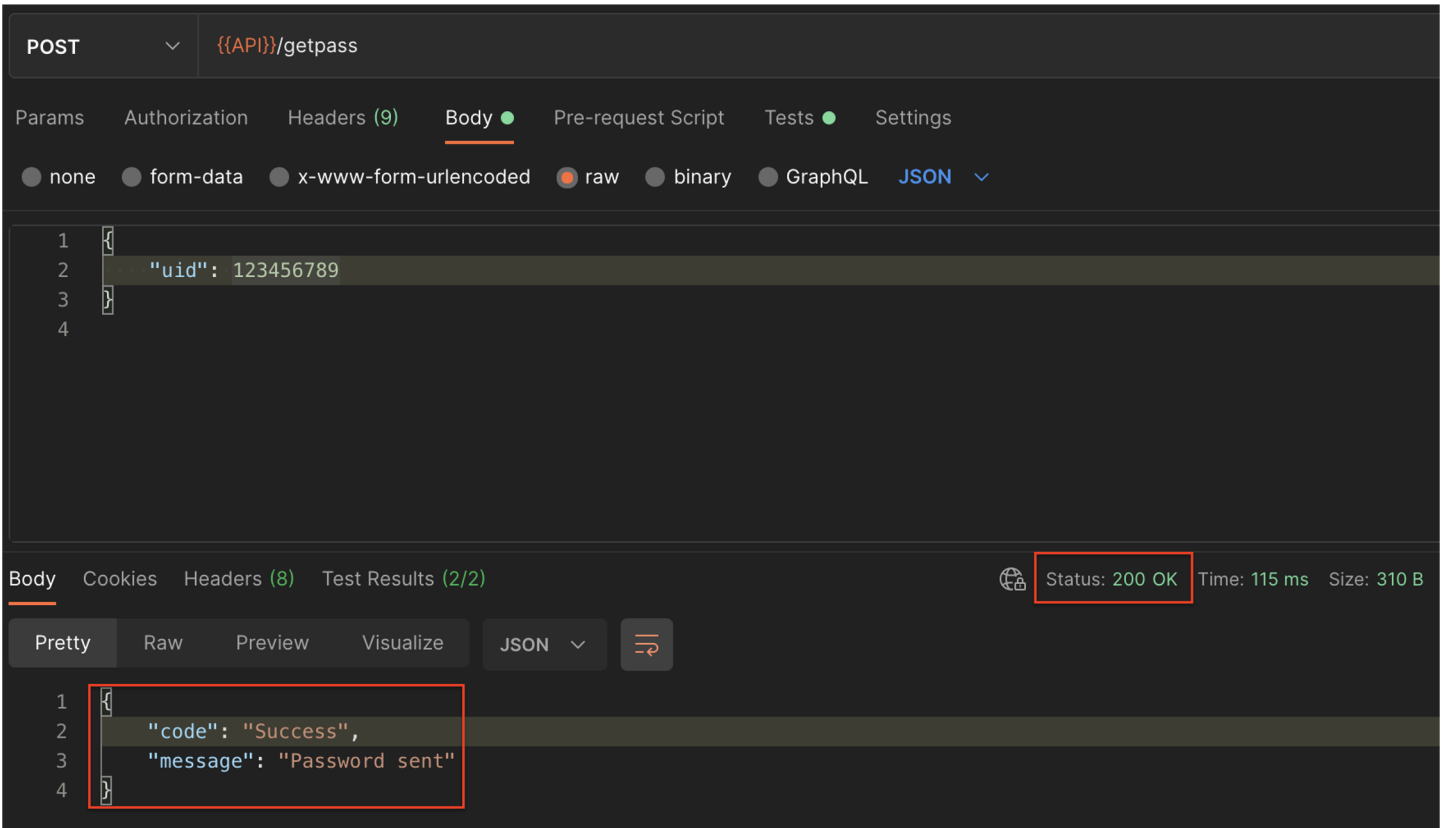
https://www.autoqa.me

Command

Target

1	<i>open</i>	https://www.autoqa.me
2	<i>set window size</i>	1440x813
3	<i>click</i>	linkText=Portfolio
4	<i>assert element present</i>	css=h1

Selenium IDE




Postman

POST {{API}}/getpass

Params Authorization Headers (9) Body ● Pre-request Script Tests ● Settings

```
1 pm.test('status code is 200', function() {
2   pm.response.to.have.status(200);
3   pm.response.to.be.ok;
4   pm.response.to.be.json;
5 });
6
7 pm.test('response body is correct', function() {
8   const responseBody = pm.response.json();
9   pm.expect(responseBody.code).to.eql('Success');
10  pm.expect(responseBody.message).to.eql('Password sent');
11 });
12
```

Body Cookies Headers (8) Test Results (2/2)  Status: 200 OK Time: 115 ms Size: 310 B

All Passed Skipped Failed

PASS status code is 200

PASS response body is correct

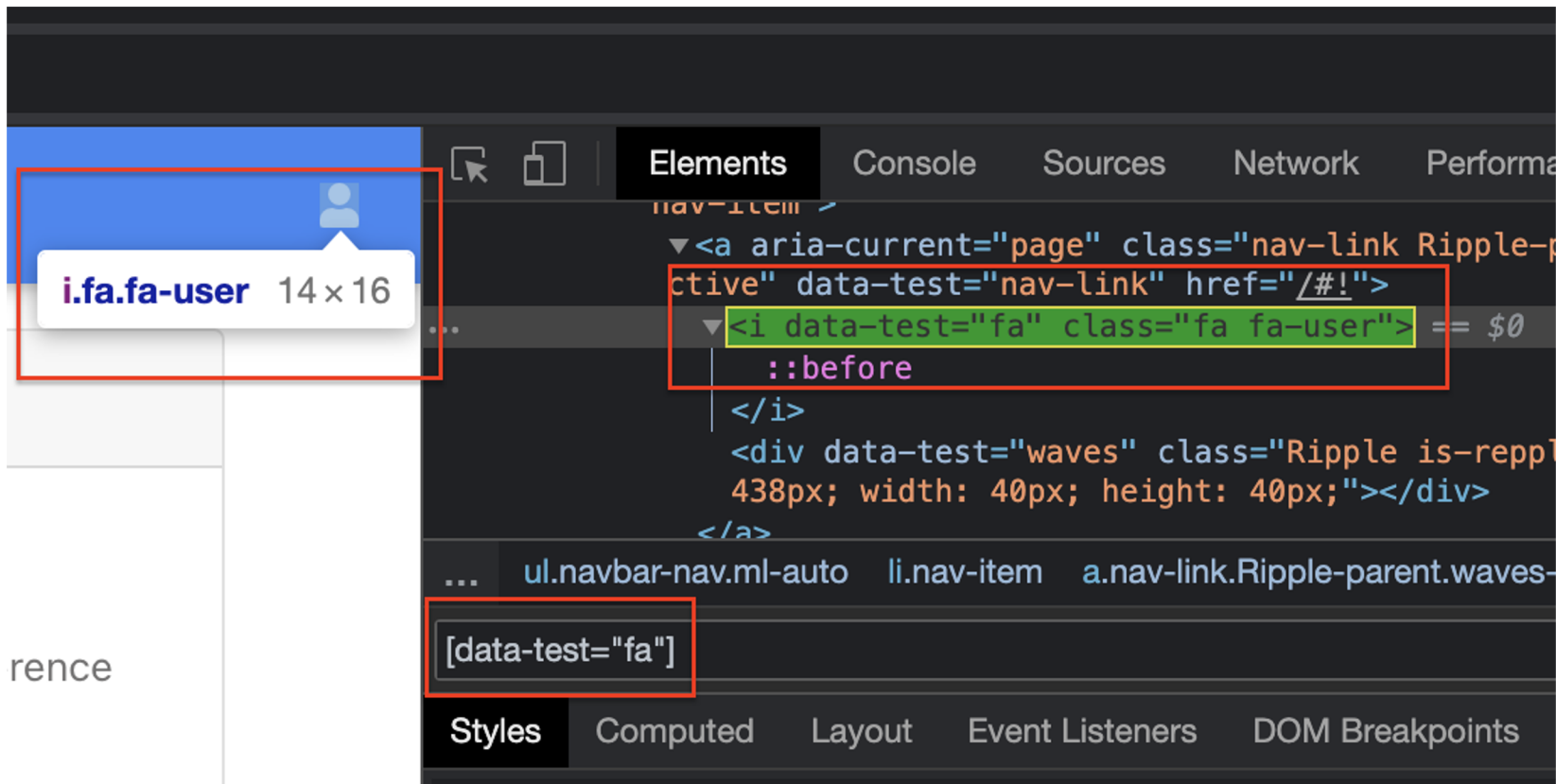
Postman

Селекторы

Селекторы

1. CSS
2. XPATH





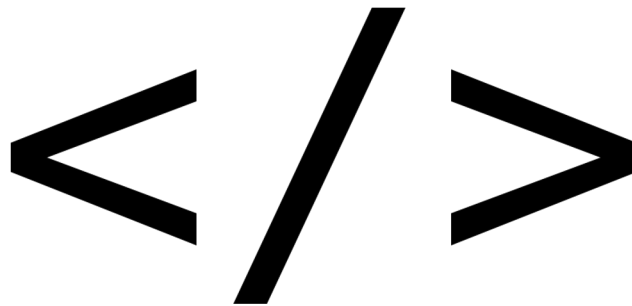
Селекторы

Выбор языка программирования

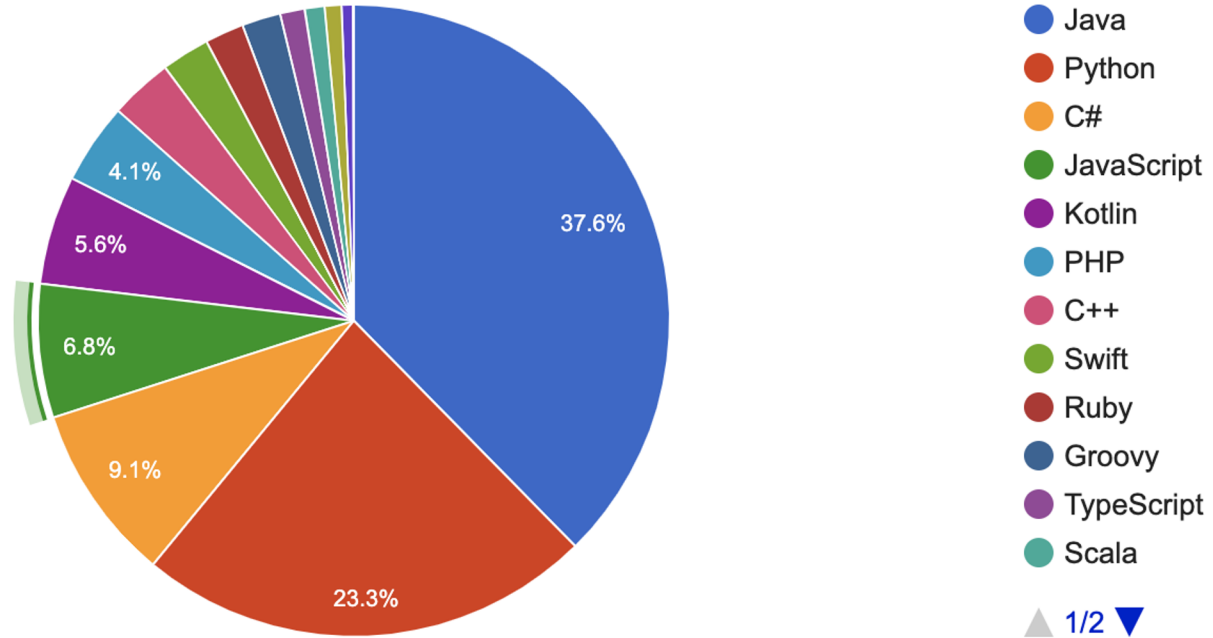
Выбор языка программирования

На выбор ЯП влияют:

1. Популярность на рынке автоматизации.
2. Тенденция дальнейшего развития.
3. Уровень сложности.

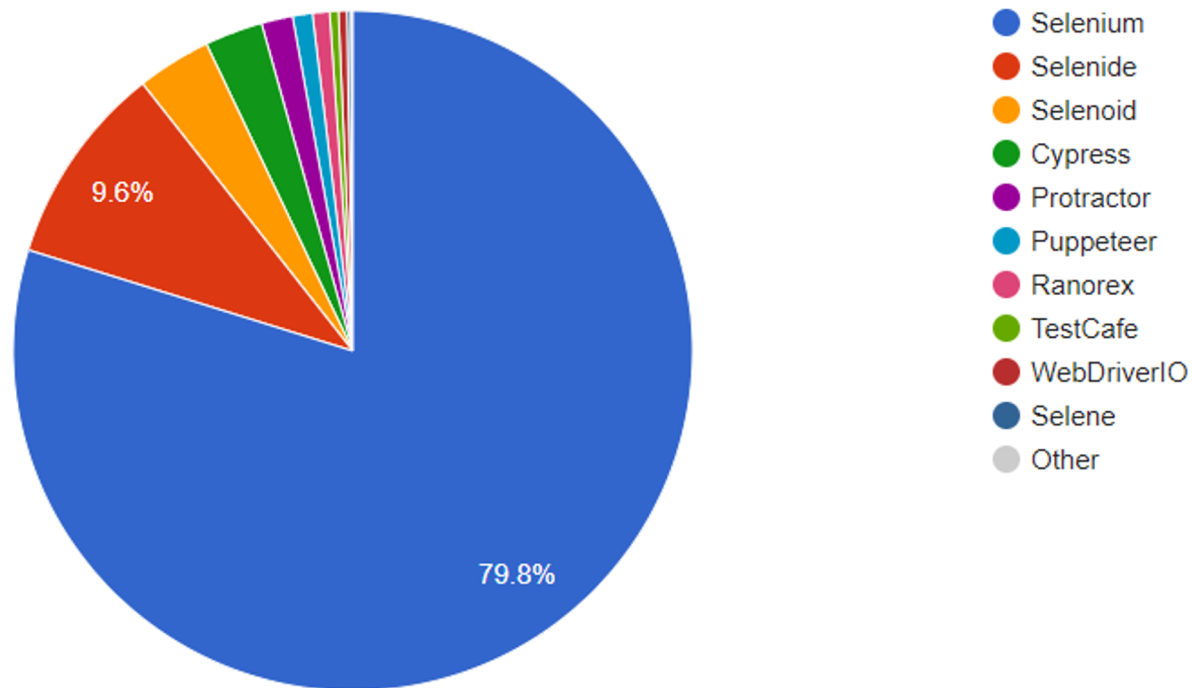


Количество упоминаний языков программирования в 2020 году.

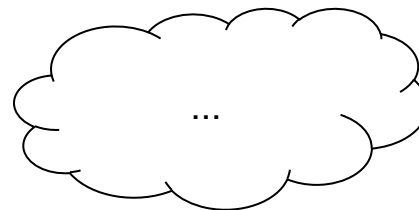
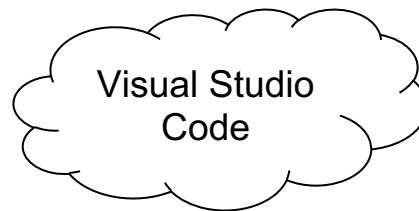
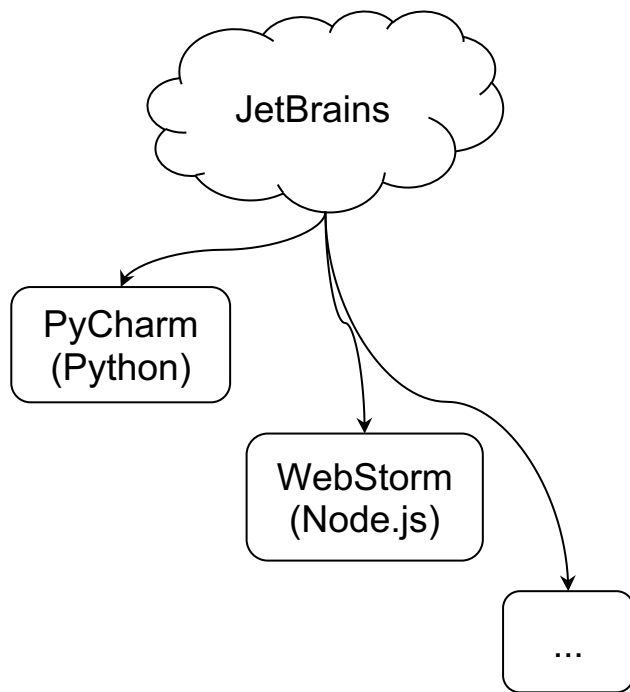


источник: <https://clinton.pythonanywhere.com/>

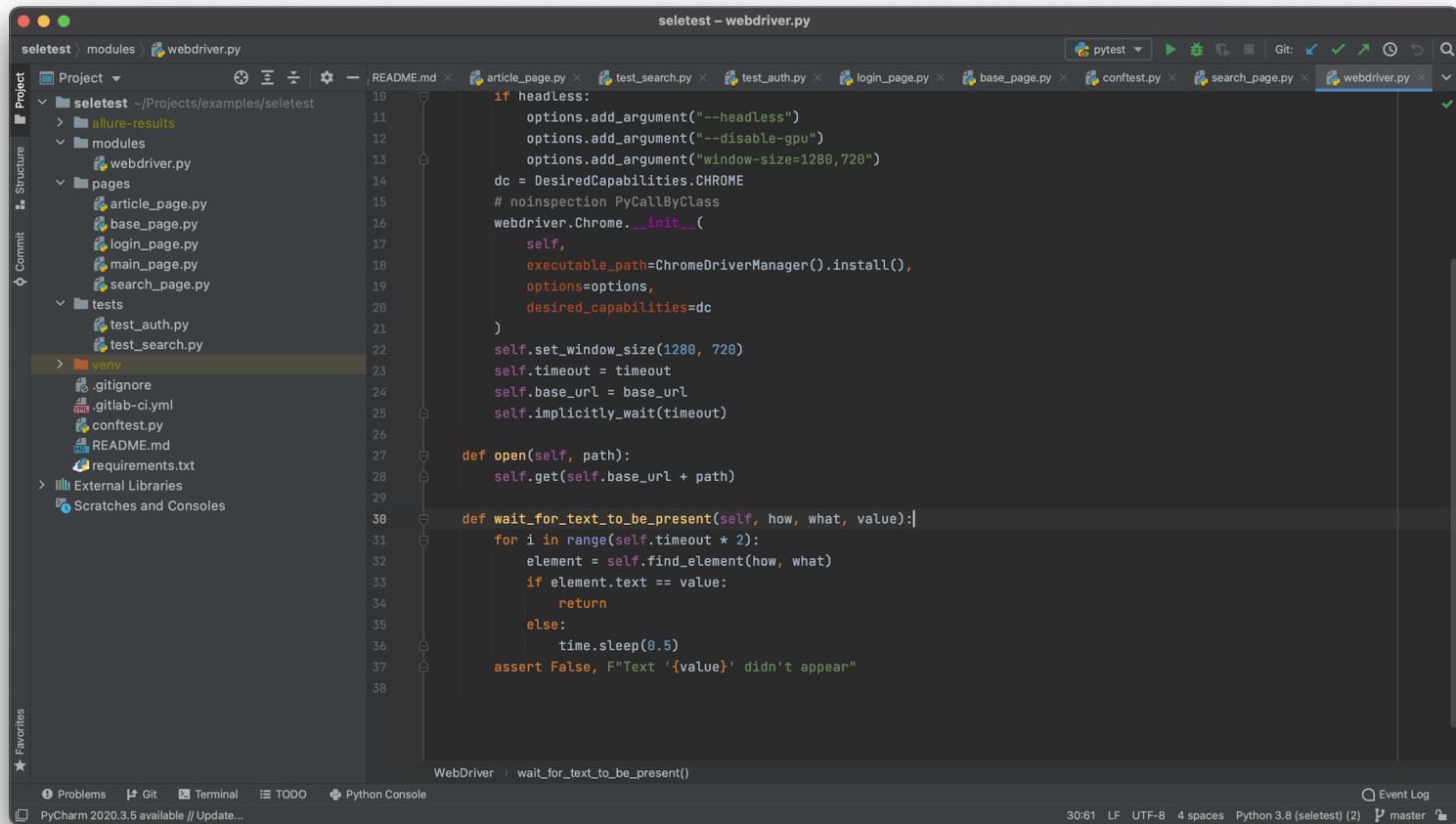
Количество упоминаний средств тестирования Web UI.

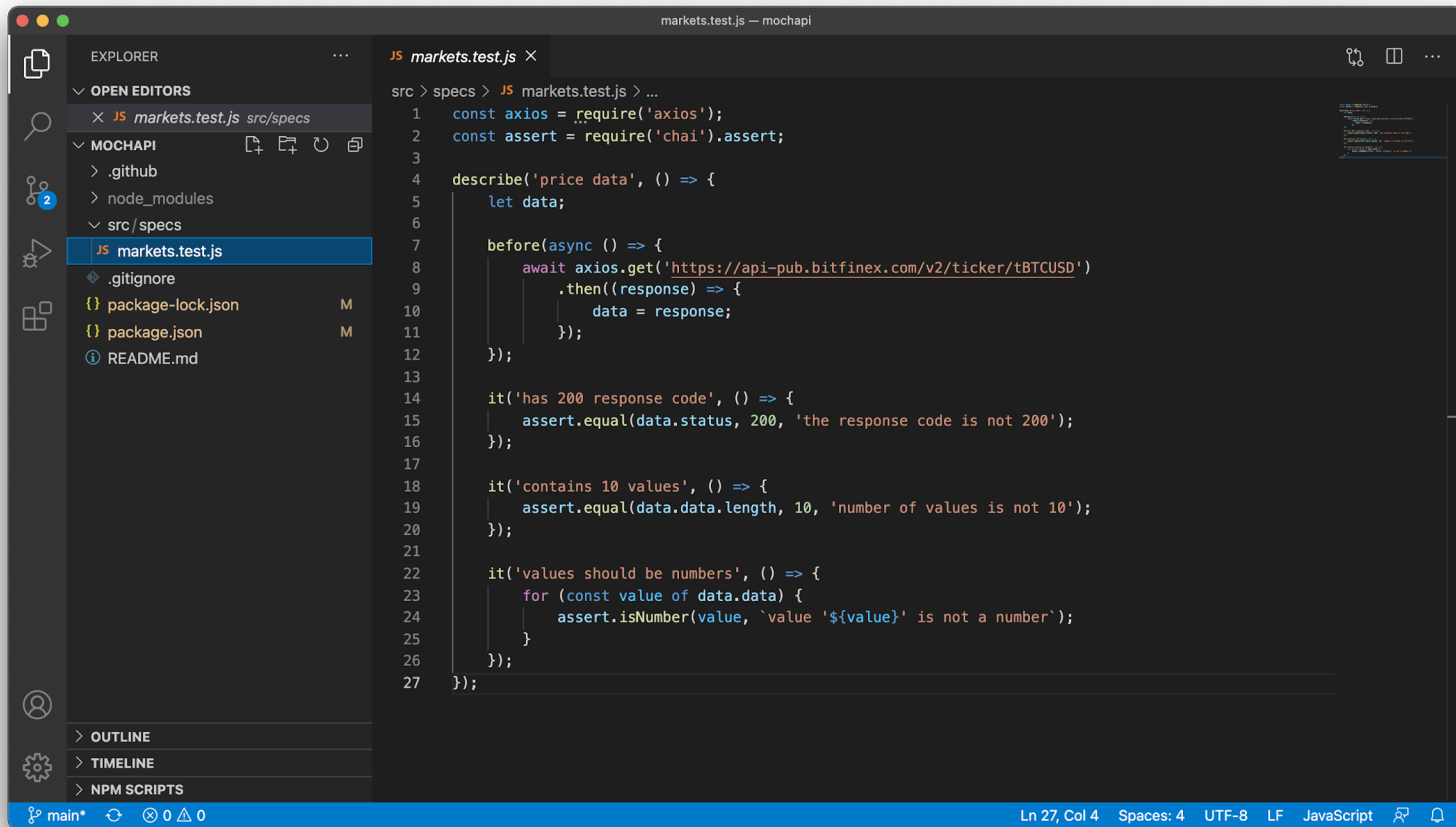


источник: <https://clingon.pythonanywhere.com/>



Выбор IDE





Visual Studio Code

Источники информации



🔍 Selenium NoSuchElementException python|



- 🔍 selenium nosuchelementexception python
- 🔍 selenium python nosuchelementexception **xpath**
- 🔍 selenium **catch** nosuchelementexception python
- 🔍 **handle** nosuchelementexception selenium python
- 🔍 python selenium nosuchelementexception **handling**
- 🔍 python selenium nosuchelementexception **example**

Поиск в Google

Мне повезёт!

Пожаловаться на неприемлемые подсказки

Источники информации

1. Google.

a. Quick start guide





🔍 webdriverio quick start ✕ 📄 🗣️

🔍 webdriverio quick start

🔍 **selenium webdriver** quick start **guide pdf**

🔍 **selenium webdriver** quick start **guide**

Поиск в Google Мне повезёт!

Пожаловаться на неприемлемые подсказки

Quick Start Guide

webdriverio quick start



Bce

Видео

Картинки

Покупки

Новости

Ещё

Настройки

Инструменты

Результатов: примерно 23 400 (0,49 сек.)

Результаты включают ссылки по запросу **webdriver quick start**

Искать только **webdriverio quick start**

webdriver.io › docs › gettingst... ▾ [Перевести эту страницу](#)

Getting Started | WebdriverIO

Run Test#. You can **start** your test suite by using the run command and pointing to the **WebdriverIO** config that you just created: `npx wdio run ./wdio.conf.js`. Copy.

Quick Start Guide

Installation

Use npm or Yarn to install the WebdriverIO test runner in your Node.js project. See [system requirements](#).

npm Yarn

```
npm install @wdio/cli
```

This single command downloads the WebdriverIO CLI tool that helps you set up WebdriverIO in your project.

Set Up

Once you've installed the CLI you can bootstrap a *Hello World* test suite into your project by running:

```
npx wdio config
```

Источники информации

1. Google.

- a. Quick start guide
- b. Документация



url

Protocol binding to load the URL of the browser. If a baseUrl is specified in the config, it will be prepended to the url parameter using node's url.resolve() method. Calling `browser.url('...')` with the same url as last time will trigger a page reload.

Usage

```
browser.url(url)
```

Parameters

Name	Type	Details
<i>url</i> optional	String	the URL to navigate to

Источники информации

1. Google.

- a. Quick start guide
- b. Документация
- c. StackOverflow



Error here:

```
NameError: name 'NoSuchElementException' is not defined
```



You need to import the exception first

154

```
from selenium.common.exceptions import NoSuchElementException
```



and then you can reference it



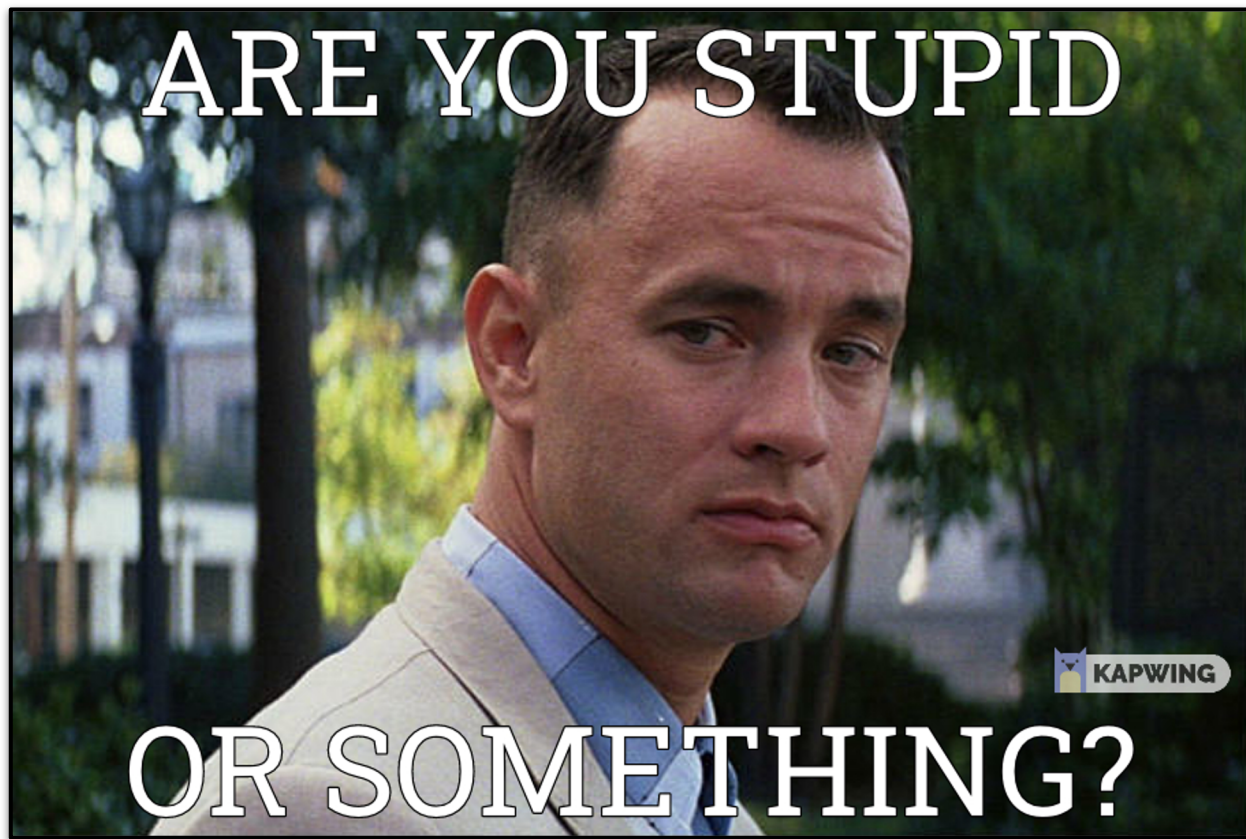
```
except NoSuchElementException:  
    # handle the element not existing
```

StackOverflow

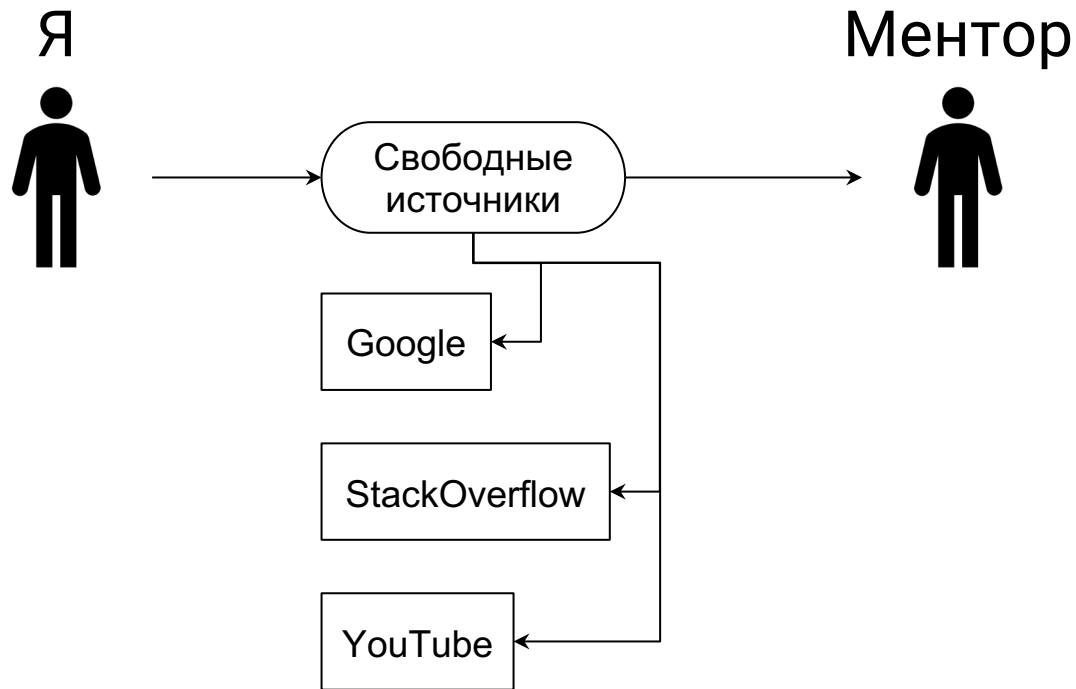
Источники информации

1. Google.
 - a. Quick start guide
 - b. Документация
 - c. StackOverflow
 - d. YouTube
2. Онлайн курсы.
 - a. Платные
 - b. Бесплатные
3. Наставник?
4. Разработчики?





Источники информации / Глупые вопросы?

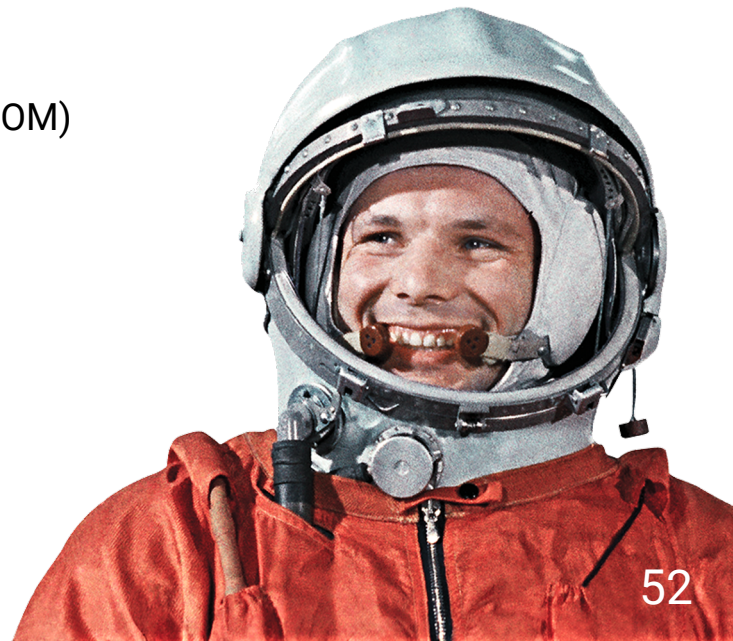


Источники информации / Взаимодействие с наставником

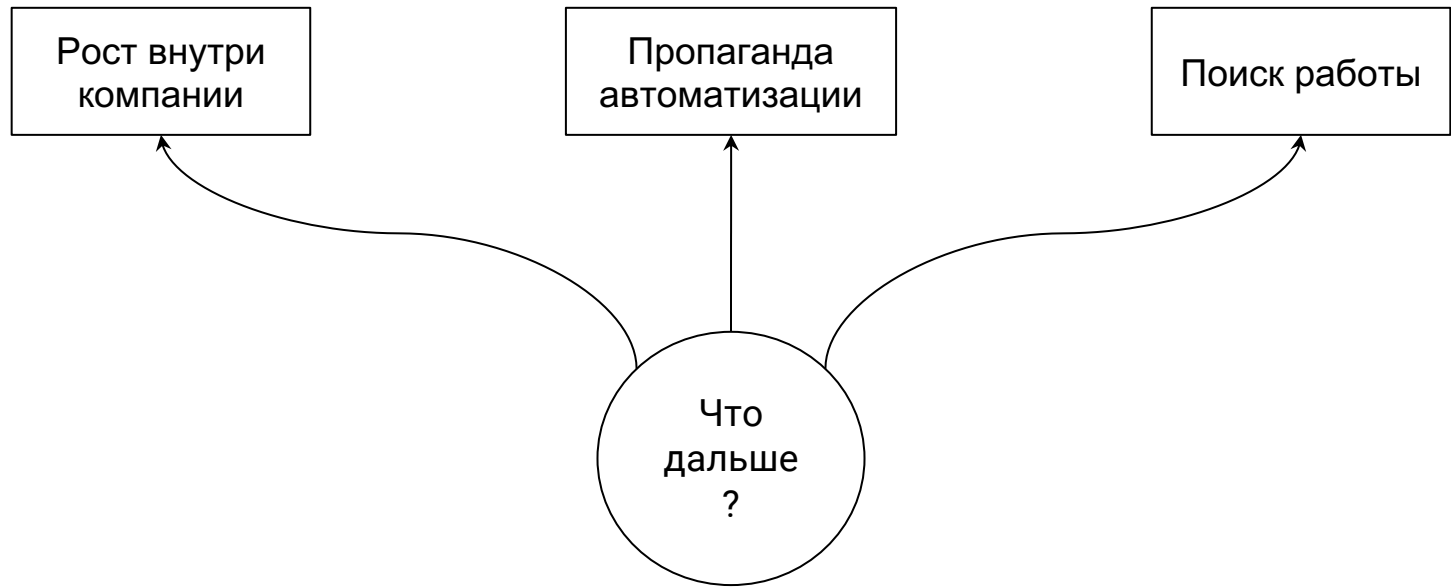
Готовность к автоматизации

Готовность к автоматизации

1. Теория тестирования.
2. Работа с кодом.
 - a. Основы программирования
 - i. Паттерн проектирования Page Object Model (POM)
 - b. Git
3. Результат.
 - a. Готовый проект с нуля.
 - b. Новые тесты на текущем проекте.



Дальнейшие шаги



Дальнейшие шаги

Дальнейшие шаги

Пропаганда автоматизации

- покрыть действующий функционал тестами
- сделать наглядную систему отчётов/алертов
- показать на цифрах, как автотесты помогли сэкономить время
 - время на прогон сценариев вручную
 - частота прогонов

Дальнейшие шаги

Поиск работы

- example project на GitHub
- позиции, совмещающие Manual и Automation роли
- outsource проекты/компании

Дальнейшие шаги

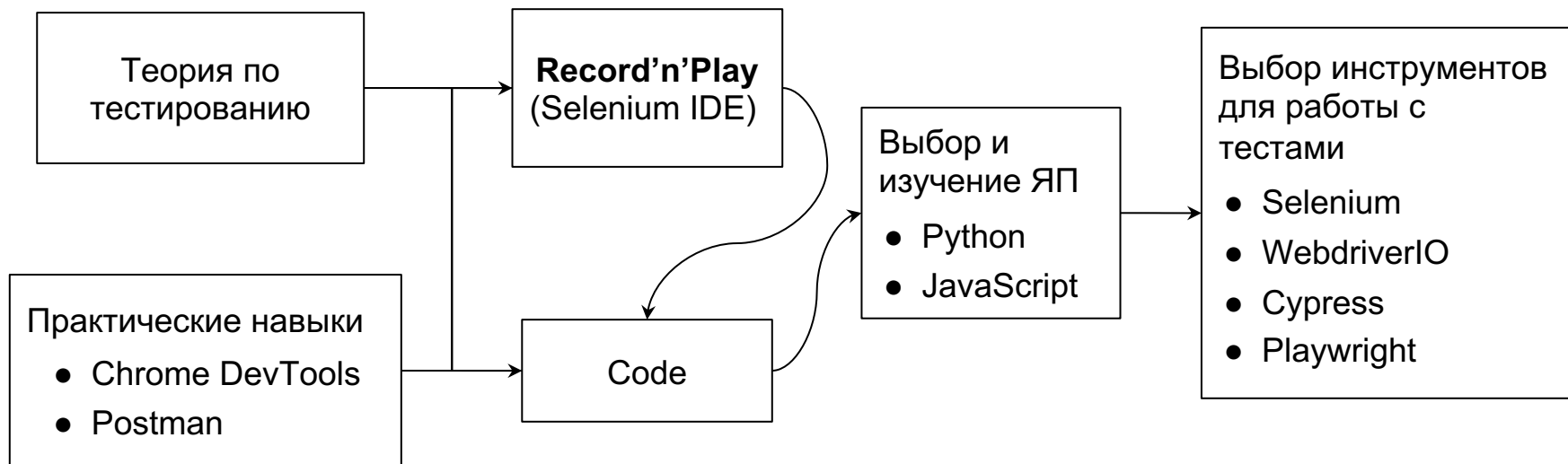
Развитие

Изучение:

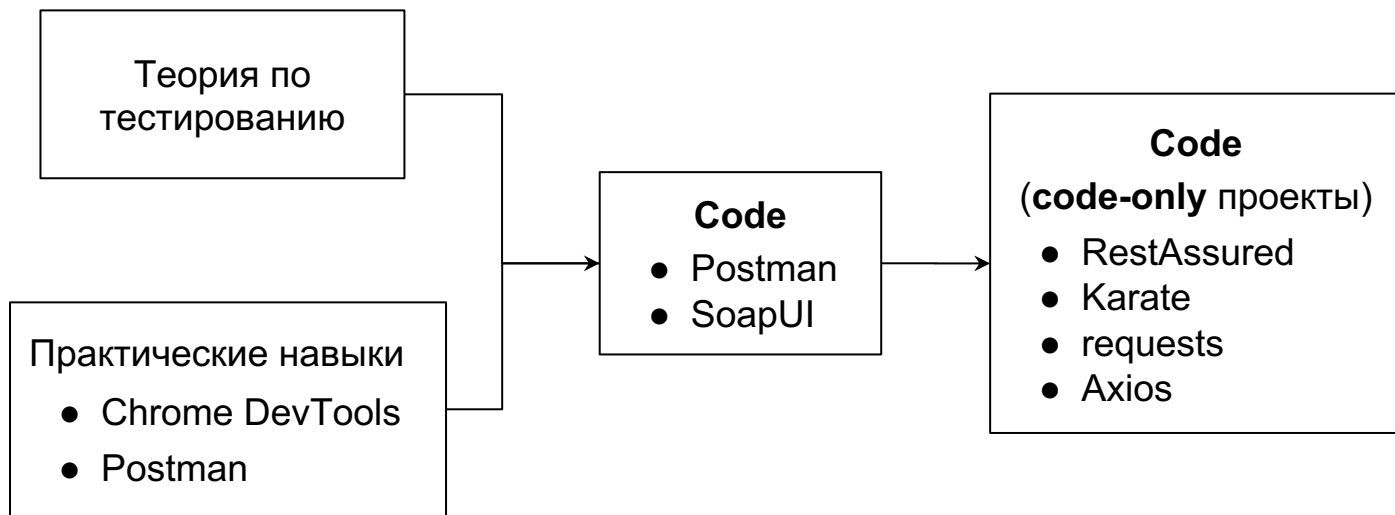
- язык программирования
- тест раннеры
- средства для манипуляций с тестируемым объектом
 - как инструмент работает “под капотом”
- инструменты для формирования отчётов
- CI/CD

Заключение

Заключение / Первые шаги (UI)



Заключение / Первые шаги (API)



Заключение / Откуда брать информацию?

Бесплатно, общедоступно, медленно

- Материалы в интернете
 - YouTube
 - StackOverflow
 - Тематические статьи
- Документация
- Бесплатные курсы

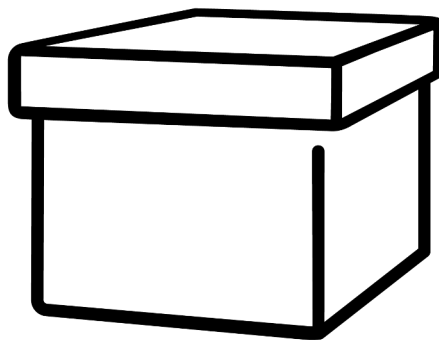
Платно, быстрее, эффективнее

- платные курсы
- менторство

Менторство на
месте работы

Заключение / Как узнать, что я готов?

1. Теория по тестированию.
2. Основы программирования.
3. Готовый проект.



Заключение / Что делать, когда готов?

Рост внутри
компании

Поиск работы



Пропаганда
автоматизации

Развитие

Q&A

- **Фамилия, имя: Бормотов Дмитрий**
- **Компания: OnTarget Group, Inc.**
- **Позиция: Senior QA Engineer**
- **Landing page: www.autoqa.me**



Спасибо за внимание!