Welcome Instructions

- WIFI Details:
  - Network: **Sauce Labs Guest**
  - Password: **secretsauce**
- Login to GitHub (or create an account)
- Go to the repo below and follow the **Gitpod** setup instructions:

https://github.com/saucelabs-training/crash-course-in-automated-selenium-testing

Crash Course in Manual to Automated Testing

By James Tacker

January 20, 2020

SAUCELABS

AGENDA

1. Software Testing 101
2. First Steps to Automated Testing
3. Writing your First Test
4. Level Up your Testing Skills

Scratching the Surface on Software Testing Basics

Speaker Bio

James Tacker

Technical Content Producer at Sauce Labs

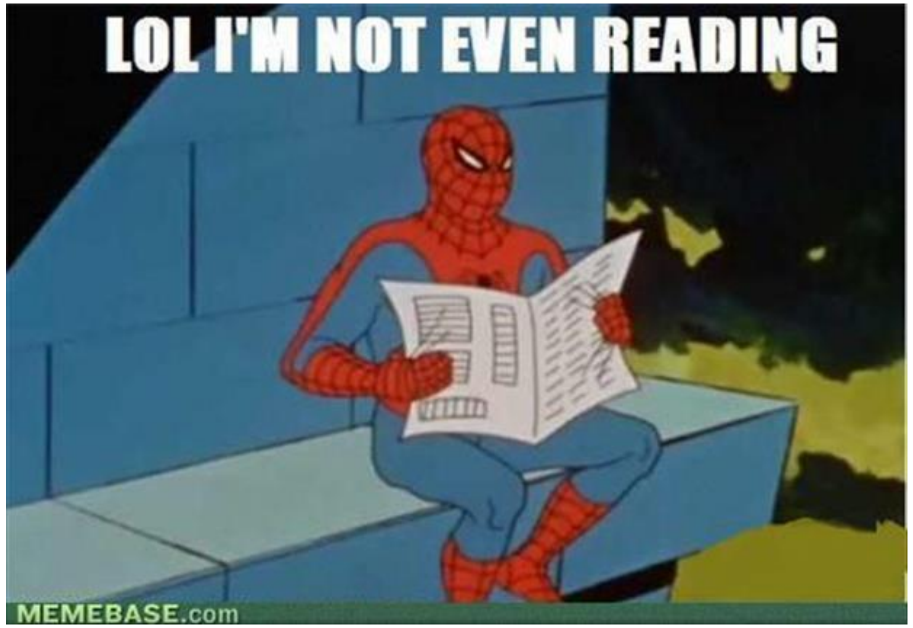
@spider-sauce

@SauceSpider

/in/jamestacker

Speaker Journey

- James Tacker
  - Technical Content Developer at Sauce Labs
  - Started out in Liberal/Fine Arts
  - Self taught Selenium enthusiast
  - Proof that this stuff can be learned!
  - Loves Spider Man



Don't worry, this is only 1 of 2 spiderman memes

Software Testing Type Examples

Manual Testing

- Alpha Testing
- Acceptance Testing
- System Testing
- Black Box / White Box Testing (Behavioral Testing)

Automated Testing

- Unit Testing
- Component Testing
- Integration Testing
- Regression Testing

Why Manual Testing Isn’t Going Away...yet

Scenarios:

- Troubleshoot failing release/build Automation
- Discovery test when receiving customer bugs from automated components
- Verify bug fixes and enhancements before patch/release
- Troubleshoot failing automation scripts
- Train automation and dev people in functional/behavioral testing so they write better code

First Steps to Becoming an Automation Engineer

Automation Engineer Prerequisites

Beginner knowledge in:

- Development Languages
  - Python
  - or Ruby
  - or JavaScript

- Operation Tools
  - Cloud Platforms (AWS/GCP/Azure)
  - Containers
  - Command Line/Prompt

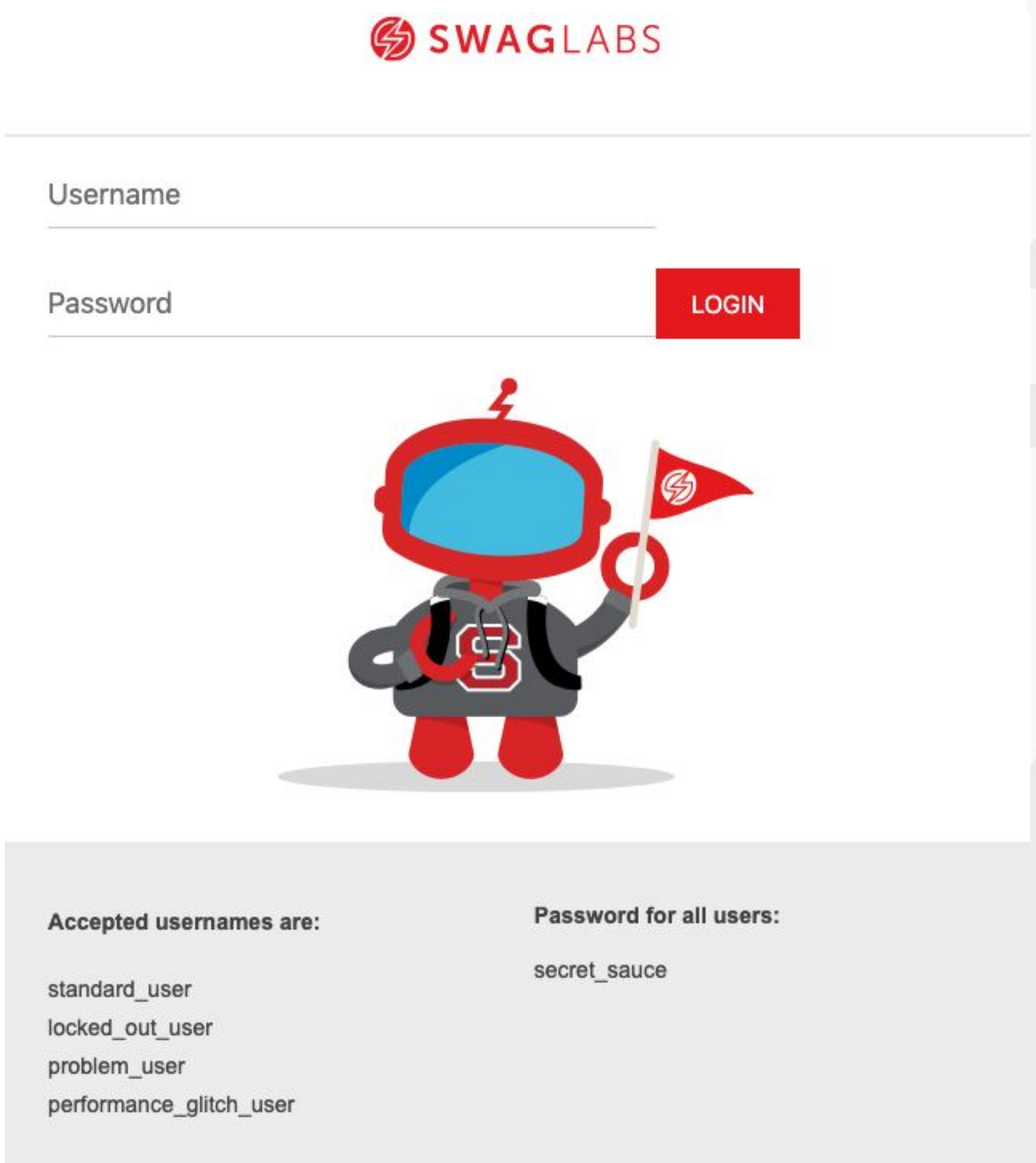
Preparing to Write Your First Script

Gather / Create a Requirements Sheet

- What does this **application** do?
- What does this **page** of this **application** do?
- What does this **function**, on this **page**, of this **application** do?
- What are the required **elements** for each **feature/function**?
- How do we locate these **elements** on the **page**?
- What are the **results** of **correct user** input?
- What are the **results** of **incorrect user** input?

Sample Application: Swag Labs

URL: https://www.saucedemo.com



Accepted usernames are:

standard_user
locked_out_user
problem_user
performance_glitch_user

Password for all users:

secret_sauce

Extract Human Interaction and Create Actions

“The first thing about cooking rabbit stew is catching the rabbit” - Isaac Asimov (I, Robot)

Identify human actions to inform your script **functions**:

- Enter correct credentials for a successful login
- Enter incorrect, or empty credentials, for an unsuccessful login

Those functions, and their expected outcomes, inform your **tests**:

- `validLoginTest()`
- `invalidLoginTest()`

Selenium in Five Minutes

What is it and what do I need to know?

- It’s an Automated Testing Tool
- Uses the W3C WebDriver Protocol (as of 2019)
- Download and install the relevant Selenium language bindings
- Install the WebDriver for a specific browser (chrome, firefox etc.)

W3C WebDriver Protocol

- Documentation
- WebDriver is an Interface
- W3C is a web protocol
- The first thing your script should do is create a WebDriver session that communicates with your local browser.

Demo: Use a Selenium WebDriver to open a browser.

GitHubLink: https://github.com/saucelabs-training/crash-course-in-automated-selenium-testing

Identifying Elements on a Page

- HTML DOM
- Element locator strategies
  - IDs
  - Class Names
  - CSS
  - XPath
- Use browser's developer tools
- Test your locators in the **Console** before you begin scripting!
  - CSS: `$$("#user-name")`
  - XPath: `$x("//*[@id=\"user-name\"]")`

Demo: Locate Elements using Chrome Dev Tools

Locating Elements with Selenium

Example in Java:

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebElement;
WebElement usernameElement = driver.findElement(By.id("username"));
WebElement passwordElement = driver.findElement(By.id("password"));
WebElement submitElement =
driver.findElement(By.className(“btn_action”);
```

Demo: Locate Elements in an Application

Perform Actions on Elements with Selenium

Example in Java:

```
usernameElement.sendKeys(“peter_parker”);
passwordElement.sendKeys(“spider_man”);
submitElement.click();
```

Demo: Perform Actions on Located Elements

Assert Expected Results of Test with Selenium

Example in Java:

```
String actualResult =
driver.getCurrentUrl().equals("https://www.saucedemo.com/inventory.ht
ml") ? "passed" : "failed";

System.out.println("The Test " + actualResult);
```

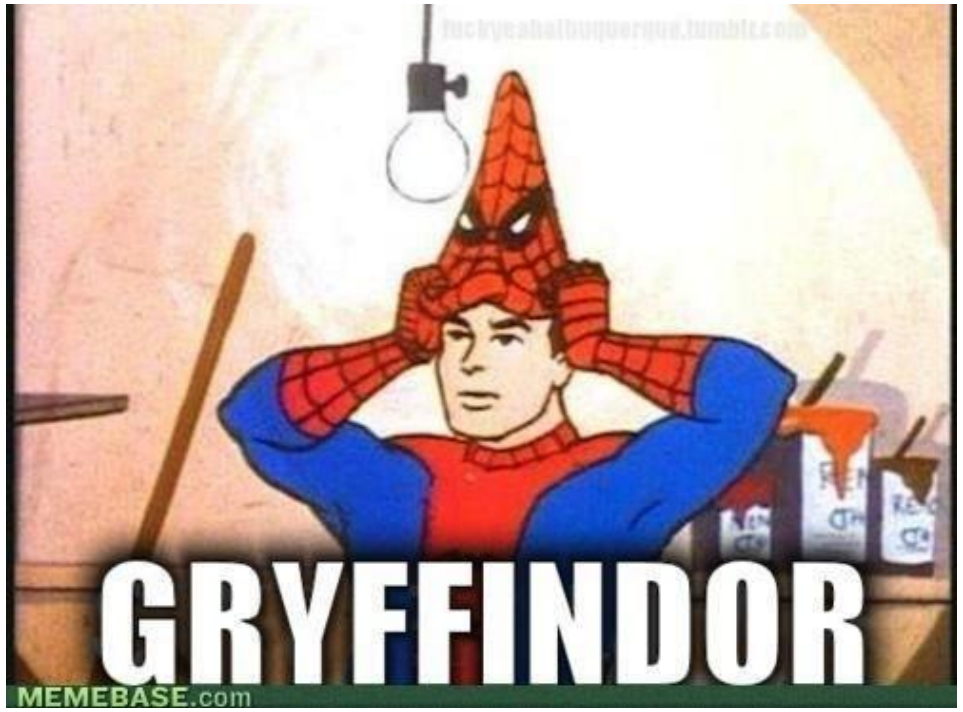
Demo: Write Test Assertion for Test Actions

What Next?

SAUCELABS

Put on your Automation Hat and...

- Practice, Practice, Practice!
- Try Appium (for mobile application testing)
- Implement the Page Object Model for improved test performance
- Configure Parallel Testing
- Cross Browser/Platform Testing (try testing on Sauce Labs! :))



Selenium Resources

Developer Tutorials

Selenium HQ

Elemental Selenium

Sauce Labs Cookbook

Thank You!

SAUCELABS