




Get Some UI In Your CI Trivia Edition

Jeff McKenzie • jeff.mckenzie@insight.com • [@jeffreymckenzie](https://twitter.com/jeffreymckenzie)

Good morning/afternoon everyone –
Thanks for being here...


Today – UI testing, specifically within
A continuous Integration environment.
This is the special Trivia Edition,
Which we will get into a little later.

My name is Jeff McKenzie,
And I am a Practice Manager for App Dev and Infrastructure
At Insight Digital Innovation in Columbus Ohio



What Insight do you need today?

Invest smarter.




Supply Chain Optimization

By optimizing your IT supply chain, we help you simplify processes, maximize resources and invest in the future.

[Learn more](#)

Work smarter.




Connected Workforce

Seamlessly connect employees and organizations with the tools and support workers need to be productive, engaged and inspired.

[Learn more](#)

Run workloads smarter.




Cloud + Data Center Transformation

We'll help you achieve greater agility, efficiency and security by transforming your cloud and data center platforms.

[Learn more](#)

Innovate smarter.



Digital Innovation

From ideation to execution, we'll help you drive meaningful outcomes with the Internet of Things, artificial intelligence and other emerging tech.

[Learn more](#)

Insight is a global, fortune 500 company

-- does a lot of things in tech

But, acquired Cardinal to help expand their Digital Innovation division

Digital innovation solves business problems using:

- Custom development
- With established as well as emergent technologies

Pillars of Digital Innovation

Digital Experience provides winning solutions that engage customers and employees in new, meaningful ways.

Disruptive Technology Innovations, both proven and emerging, captivate customers, empower employees and transform business operations.

Continuous Services sustains innovation with strategy, support, delivery, and data-driven insights.



DI



- Do a lot of cloud work, app modernization
- Big data, predictive analytics
- Devops on both the Microsoft and open source side
- As well as a fair amount of IoT solutions

Awards



Our combined IT industry knowledge and technology expertise have earned us numerous Microsoft honors through the years.

2018

- Worldwide Artificial Intelligence Partner of the Year
- Worldwide Modern Workplace Partner of the Year
- U.S. Partner Award for Apps and Infrastructure – DevOps
- U.S. Partner Award for Data & AI – Internet of Things
- U.S. Partner Award for Apps & Infrastructure – Open Source Apps and Infrastructure on Azure

2017

- Worldwide Mobile Application Development Partner of the Year
- Worldwide Open Source on Azure Partner of the Year
- U.S. NSP Partner of the Year (and runner up)
- U.S. FED Partner of the Year
- U.S. EDU Partner of the Year

2016

- Worldwide Internet of Things Partner of the Year
- U.S. FED Partner of the Year
- OEM Device Partner of the Year

As a worldwide Microsoft partner, Insight has received a good share of awards,

Including: IoT Partner of the year in 2016

Mobile App Dev and Open Source Azure in 2017

AI and Modern Workplace awards in 2018

@jeffreymckenzie

Ci(ui)

Insight.

Ok, so UI in your CI...
So we are talking about UI Tests,
Specifically how they fit in your continuous integration process.

We are going to get into the details of how to set up
A UI project in ci pipelines
For both Java and dotnet core
In Jenkins and Azure devops,

But first we have some trivia...



When we are doing trivia, the way this works is -----
I ask you questions, and you tell me the answers,
Guessing is allowed
Or at least give me your best guess.

And no cheating... no phones..

[https://commons.wikimedia.org/wiki/File:Transit_Trivia_Night_\(12240727563\).jpg](https://commons.wikimedia.org/wiki/File:Transit_Trivia_Night_(12240727563).jpg)
https://upload.wikimedia.org/wikipedia/commons/7/72/Transit_Trivia_Night_%2812240727563%29.jpg
Metropolitan Transportation Authority of the State of New York [CC BY 2.0
(<https://creativecommons.org/licenses/by/2.0>)]

Q1: What is Selenium?

! Insight.

Going to start you off easy..

What is Selenium?

And here I'm going to give you a hint...



Not talking about this kind, but about this kind.....

www.amazon.com/Dr-Seths-Selenomethionine-Hypoallergic-Antioxidant/dp/B07FYW36T9

https://commons.wikimedia.org/wiki/File:Selenium_black.jpg

https://upload.wikimedia.org/wikipedia/commons/9/98/Selenium_black.jpg

W. Oelen [CC BY-SA 3.0 (<https://creativecommons.org/licenses/by-sa/3.0/>)]

[https://commons.wikimedia.org/wiki/File:34_selenium_\(Se\)_enhanced_Bohr_model.png](https://commons.wikimedia.org/wiki/File:34_selenium_(Se)_enhanced_Bohr_model.png)

https://upload.wikimedia.org/wikipedia/commons/8/8e/34_selenium_%28Se%29_enhanced_Bohr_model.png

Ahazard.sciencewriter [CC BY-SA 4.0 (<https://creativecommons.org/licenses/by-sa/4.0/>)]



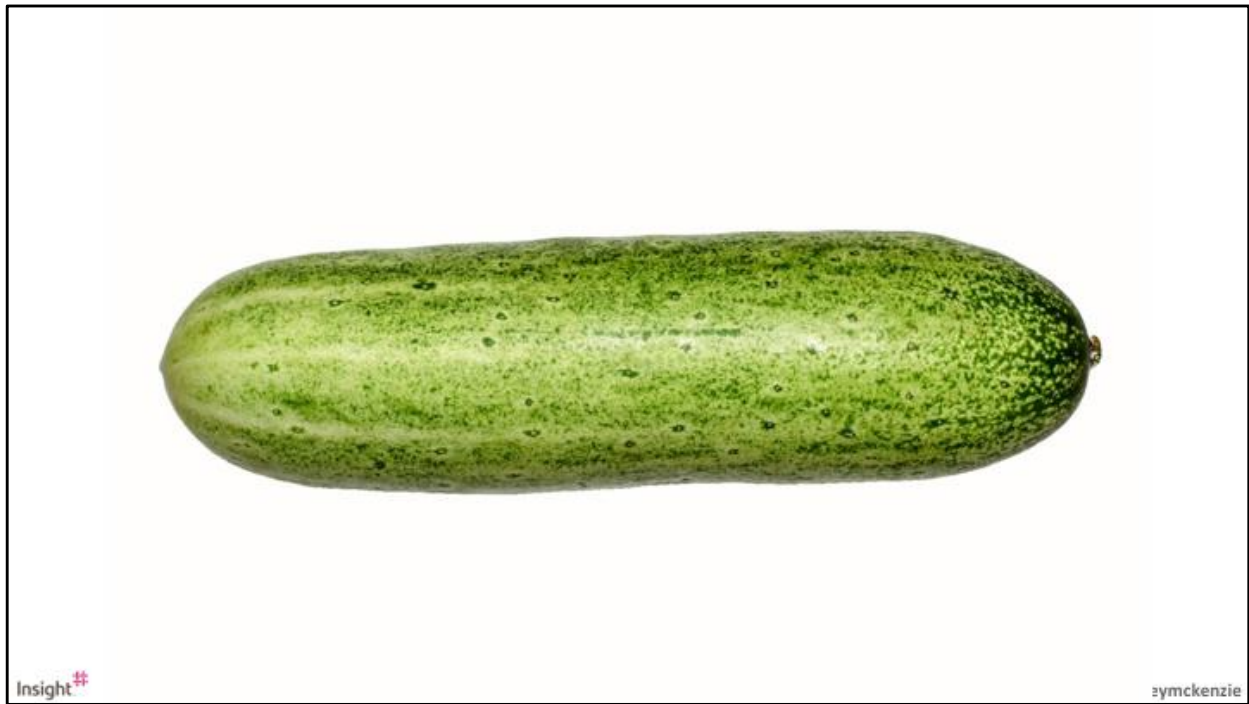
Yeah so it's a framework that automates browsers,
Which makes it perfect for web user interface testing
[not a selenium talk]--

Q2: What is Cucumber?

🔗 Insight.

What is Cucumber?

And here I'm going to give you a hint...



Not talking about this kind....

https://commons.wikimedia.org/wiki/File:Cucumber_BNC.jpg

https://upload.wikimedia.org/wikipedia/commons/a/a3/Cucumber_BNC.jpg

I, Prathyush Thomas [GFDL 1.2 (<http://www.gnu.org/licenses/old-licenses/fdl-1.2.html>) or FAL]



There's also the South African Spiny Cucumber

[https://commons.wikimedia.org/wiki/File:South_African_Spiny_Cucumber_\(Cucumis_zeyherii_\(14014788071\).jpg](https://commons.wikimedia.org/wiki/File:South_African_Spiny_Cucumber_(Cucumis_zeyherii_(14014788071).jpg)

https://upload.wikimedia.org/wikipedia/commons/8/8a/South_African_Spiny_Cucumber_%28Cucumis_zeyherii%29_%2814014788071%29.jpg

Bernard DUPONT from FRANCE [CC BY-SA 2.0

(<https://creativecommons.org/licenses/by-sa/2.0>)]

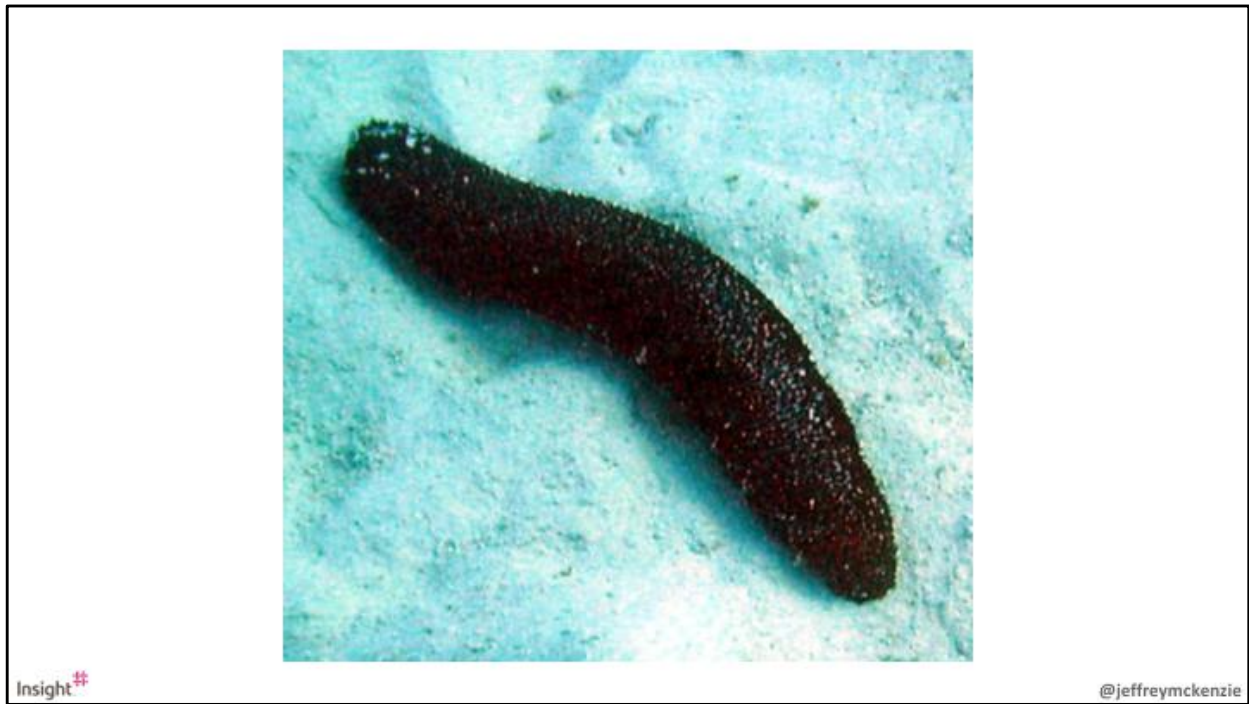


The horn cucumber

<https://commons.wikimedia.org/wiki/File:Horn-cucumber-74267.jpg>

<https://upload.wikimedia.org/wikipedia/commons/6/6d/Horn-cucumber-74267.jpg>

<https://pixabay.com/en/users/Lebensmittelfotos-13/> [CC0]



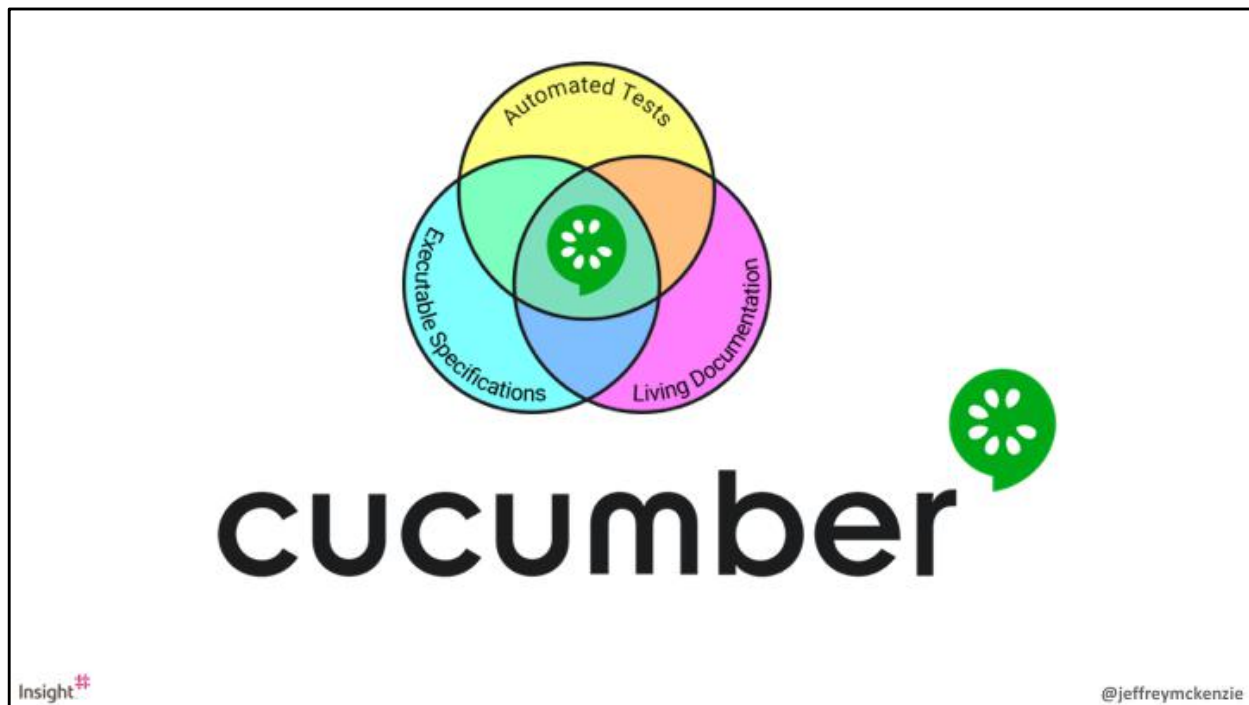
...and the sea cucumber

https://commons.wikimedia.org/wiki/File:Sea_cucumber.jpg

https://upload.wikimedia.org/wikipedia/commons/d/df/Sea_cucumber.jpg

Jacinta Richardson and Paul Fenwick [CC BY-SA 3.0

(<http://creativecommons.org/licenses/by-sa/3.0/>)]

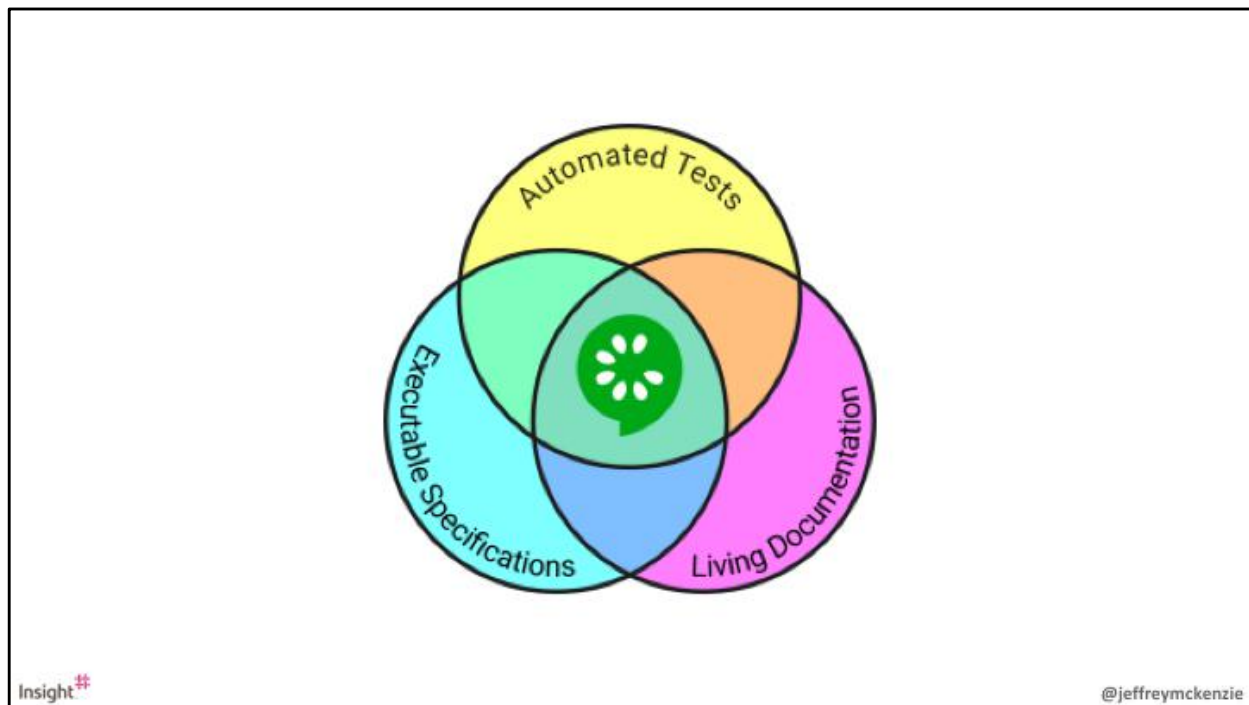


Cucumber the BDD framework

(BTW, what's this called in .NET?) [Specflow]

I like this graphic at the top –

Cucumber.io



Cucumber is not really a testing framework (part of graphic)
But it enables “executable specifications”
[not a cucumber or specflow talk]

Cucumber.io

UI + BDD?



Selenium and Cucumber together gives us
UI testing with BDD --
Why would we combine
behavior driven development with
UI Testing?
[combination of business and tech,
Where it's natural for end-users
And developers to work together]

Q3: Name That Language

🔍 Insight.

Speaking of cucumber

What kind of language is this? [click to reveal]

Feature: Bank Account

Scenario: Add Funds

Given a new savings account

when I withdraw 1000000000 dollars

Then I receive an error message

Gherkin

What do we do with it?

How can we actually get it to execute?

[map it to executable code]

Use plain English to automate behavior

```
import org.openqa.selenium.*;
import cucumber.api.*;

public class withdrawFundsSteps {

    WebDriver driver = new ChromeDriver();

    . . .

}
```

Insight

@jeffreymckenzie

So here is an example test, in java

As can see at top, references to
both selenium and cucumber here

This is how the gherkin is mapped
To executable code.

Cucumber the framework takes care of the wiring...

Given a new savings account

```
@Given("^a new savings account$")
public void iHaveANewSavingsAccount() {
    driver.get("https://www.abank.com/savings");
}
```

Create an instance of chrome, and when we say
"Given a savings account", go to the bank web site...

when I withdraw 1000000000 dollars

```
@when("^I withdraw {int} dollars$")
public void iwithdrawDollars(int dollar) {
    driver.findElement(By.id("amt"))
        .sendKeys(dollar);
}
```

Then we find the text box on the screen
Where we enter the billion dollars

Then I receive an error message

```
@Then("^I receive an error message$")
public void iReceiveAnErrorMessage(){
    assert(driver.findElement(By.id("error")));
}
}
```

And then we verify that we get an error message
Of insufficient funds.

All good?

Q4: What's Wrong With This Test?

 Insight.

Question 4

So what is wrong with this test?

Let's look at the whole thing together...

```

import org.openqa.selenium.*;
import cucumber.api.*;

public class withdrawFundsSteps {
    WebDriver driver = new ChromeDriver();

    @Given("^I have a new savings account$")
    public void iHaveANewSavingsAccount() {
        driver.get("http://www.mybank.com/my_savings");
    }

    @When("^I withdraw {int} dollars$")
    * * *

    @Then("^I receive an error message$")
    public void iReceiveAnErrorMessage(){
        assert(driver.findElement(By.id("error")));
    }
}

```

Insight

@jeffreymckenzie

Aside from being a fairly silly test, what is the problem with it

From a design or architecture perspective?

Essentially pulled from web...

[performance, new instance]

[tight coupling:

browser version

business logic to technical implementation

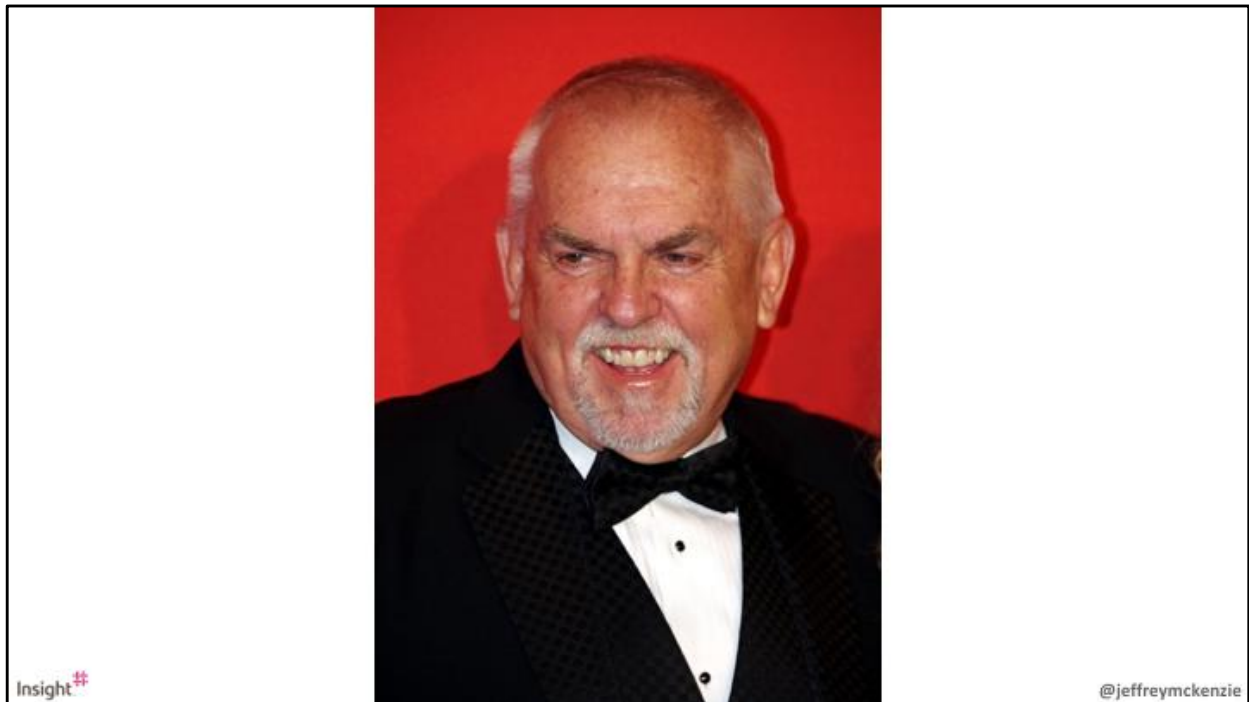
could put this in your CI, but...

you don't really want to.

Q5: Who Is This Guy?

 Insight.

Question 5 –
Who is this guy?



John Ratzenberger

The actor...

What's he been in?

https://commons.wikimedia.org/wiki/File:John_Ratzenberger_2011_Shankbone.JPG

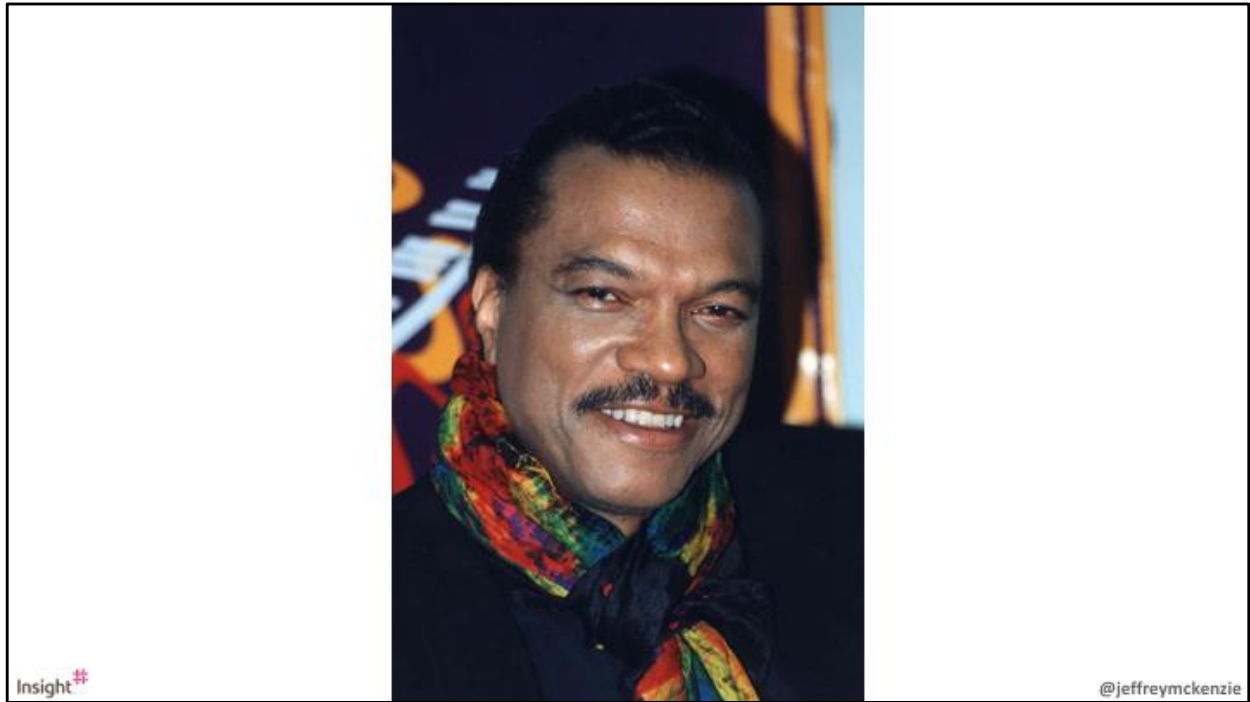
https://upload.wikimedia.org/wikipedia/commons/5/58/John_Ratzenberger_2011_Shankbone.JPG

David Shankbone [CC BY 3.0 (<https://creativecommons.org/licenses/by/3.0/>)]

Q6: Who Is This Other Guy?

 Insight.

Ok good, now question 6, who is this other guy?



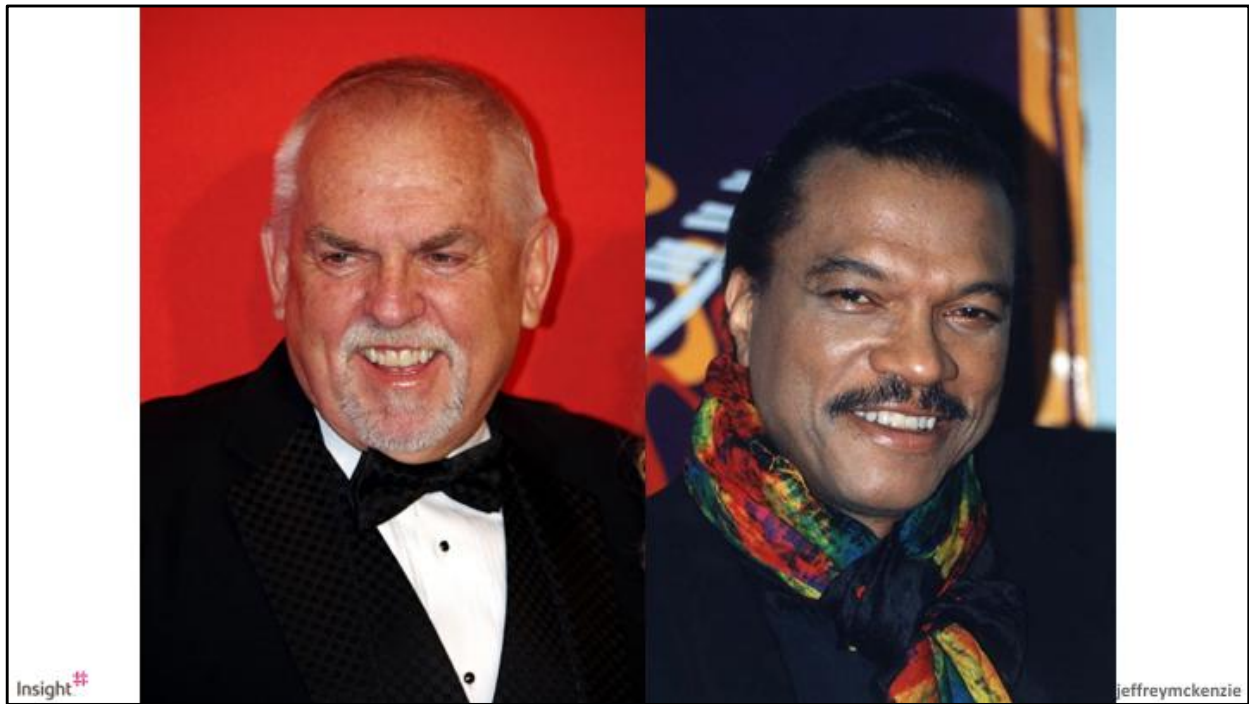
Billy Dee Williams, the actor.
What's he been in?

https://commons.wikimedia.org/wiki/File:Billy_De_Williams_in_Washington_D.C.jpg

https://upload.wikimedia.org/wikipedia/commons/d/da/Billy_De_Williams_in_Washington_D.C.jpg

Kingkongphoto & www.celebrity-photos.com from Laurel Maryland, USA [CC BY-SA 2.0 (<https://creativecommons.org/licenses/by-sa/2.0>)]

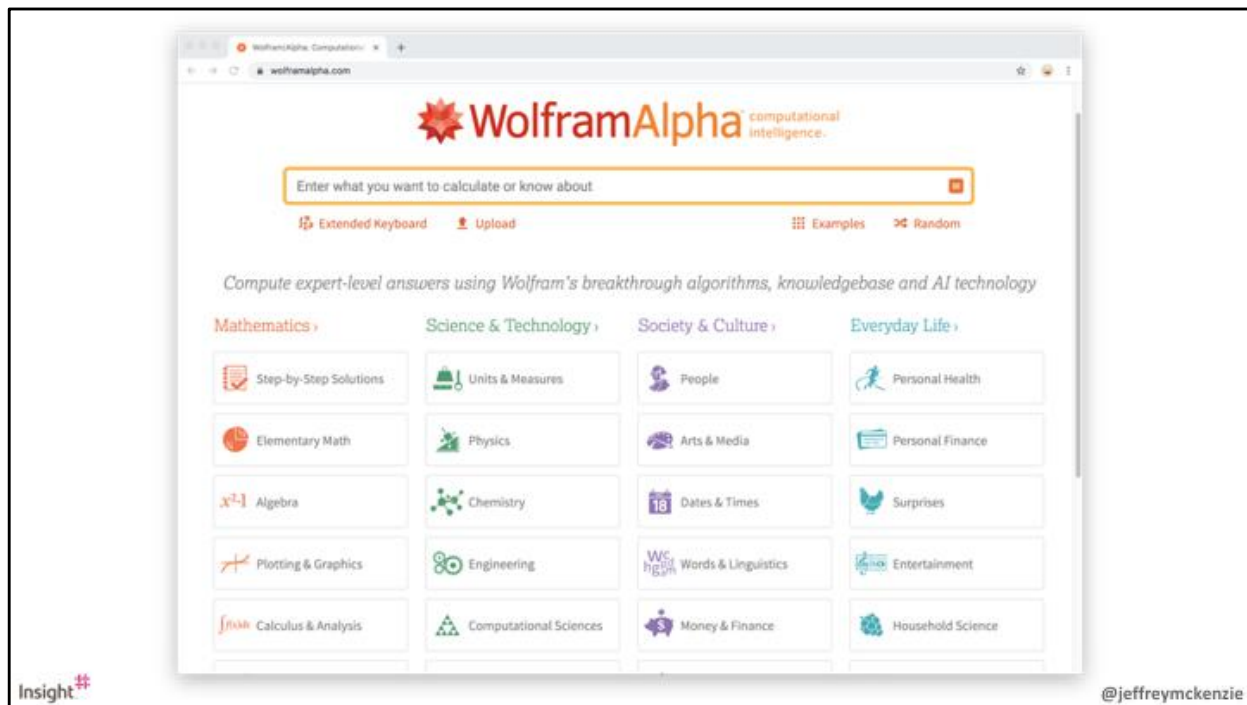
Q7: What Movies Star Both Of These Guys?



Any takers?

OK, hold on to that....

See previous for photo credits



OK well we are going to use UI testing
To help us get the answer,
Specifically with a site called Wolfram Alpha?
Anyone familiar with it?
Kind of like a QA search engine...

Feature: wolfram Trivia

Scenario: wolfram Knows Movies

Given I navigate to wolfram Alpha

When I ask "what movies star both

Billy Dee Williams and John Ratzenberger"

Then wolfram Alpha answers "?????????"

So here is our gherkin...

Let's take a look at our testing solution

@jeffreymckenzie



Insight.

Here's what the test solution looks like –

We have four modules or projects.

At the bottom, we have the foundational web project
That contains selenium...

https://www.flaticon.com/free-icon/cucumber_201710

<https://icons-for-free.com/file+multiple+page+icon+icon-1320085907501139898/>

<http://freeiconshop.com/icon/globe-icon-outline-filled/>

<http://www.iconhot.com/icon/project/todo.html>

<https://www.pngfans.com/middle-ff0ec362c9ffb9ae-gherkin-icon.html>

@jeffreymckenzie



Insight.

Then we have the pages project (or the page objects)
Which has references to the web project,
But none to selenium itself.

https://www.flaticon.com/free-icon/cucumber_201710
<https://icons-for-free.com/file+multiple+page+icon+icon-1320085907501139898/>
<http://freeiconshop.com/icon/globe-icon-outline-filled/>
<http://www.iconhot.com/icon/project/todo.html>
<https://www.pngfans.com/middle-ff0ec362c9ffb9ae-gherkin-icon.html>

@jeffreymckenzie



🔗 Insight.

Then Our steps project, which maps to our gherkin,
Have references to pages and cucumber.

https://www.flaticon.com/free-icon/cucumber_201710

<https://icons-for-free.com/file+multiple+page+icon+icon-1320085907501139898/>

<http://freeiconshop.com/icon/globe-icon-outline-filled/>

<http://www.iconhot.com/icon/project/todo.html>

<https://www.pngfans.com/middle-ff0ec362c9ffb9ae-gherkin-icon.html>

 Features
 Steps(, )
 Pages()
 Web()

 Insight.

And then features with references to the steps.

https://www.flaticon.com/free-icon/cucumber_201710

<https://icons-for-free.com/file+multiple+page+icon+icon-1320085907501139898/>

<http://freeiconshop.com/icon/globe-icon-outline-filled/>

<http://www.iconhot.com/icon/project/todo.html>

<https://www.pngfans.com/middle-ff0ec362c9ffb9ae-gherkin-icon.html>



Insight.

What this buys you is loose coupling
And promotes reuse,
Which eases maintainability.
It also helps you scale
Because you can add to features
And steps, without having
To touch pages or web.

https://www.flaticon.com/free-icon/cucumber_201710

<https://icons-for-free.com/file+multiple+page+icon+icon-1320085907501139898/>

<http://freeiconshop.com/icon/globe-icon-outline-filled/>

<http://www.iconhot.com/icon/project/todo.html>

<https://www.pngfans.com/middle-ff0ec362c9ffb9ae-gherkin-icon.html>

@jeffreymckenzie

Design

Insight.

So when we talk about UI Testing
In a continuous integration context
It is partially about design (or architecture)

@jeffreymckenzie

Mechanics

 Insight.

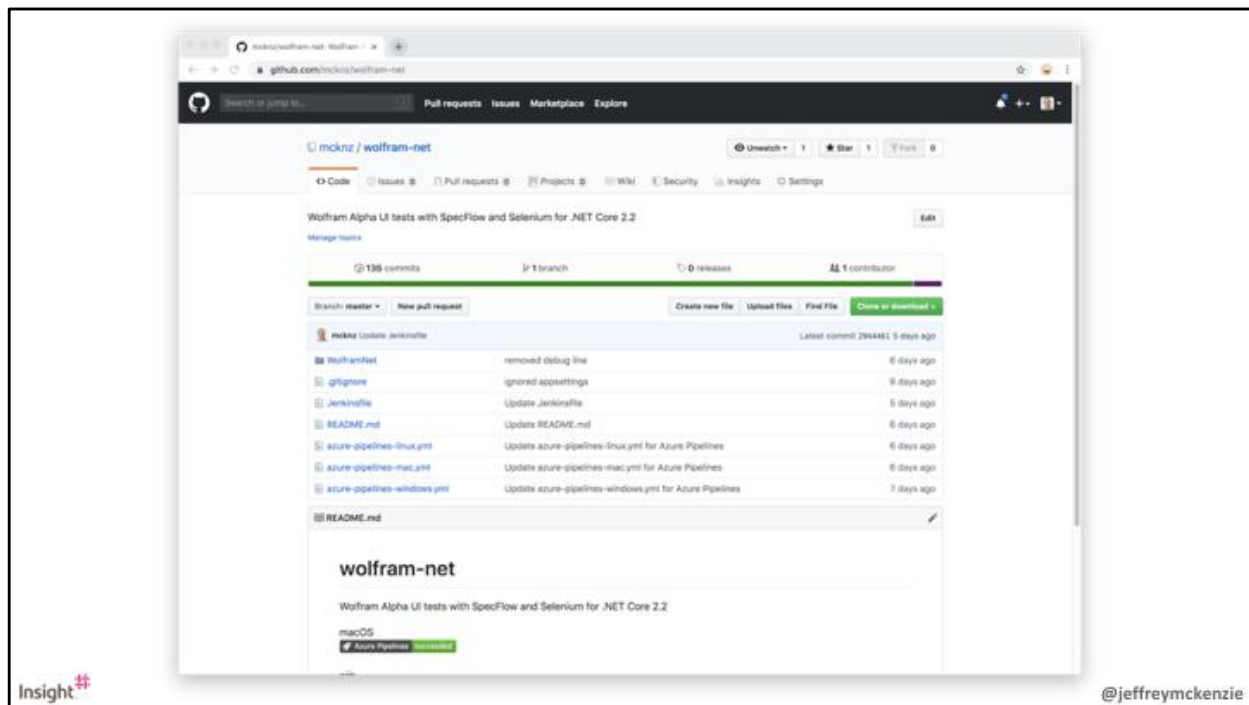
And partially about mechanics,
Or implementation.

Design & Mechanics

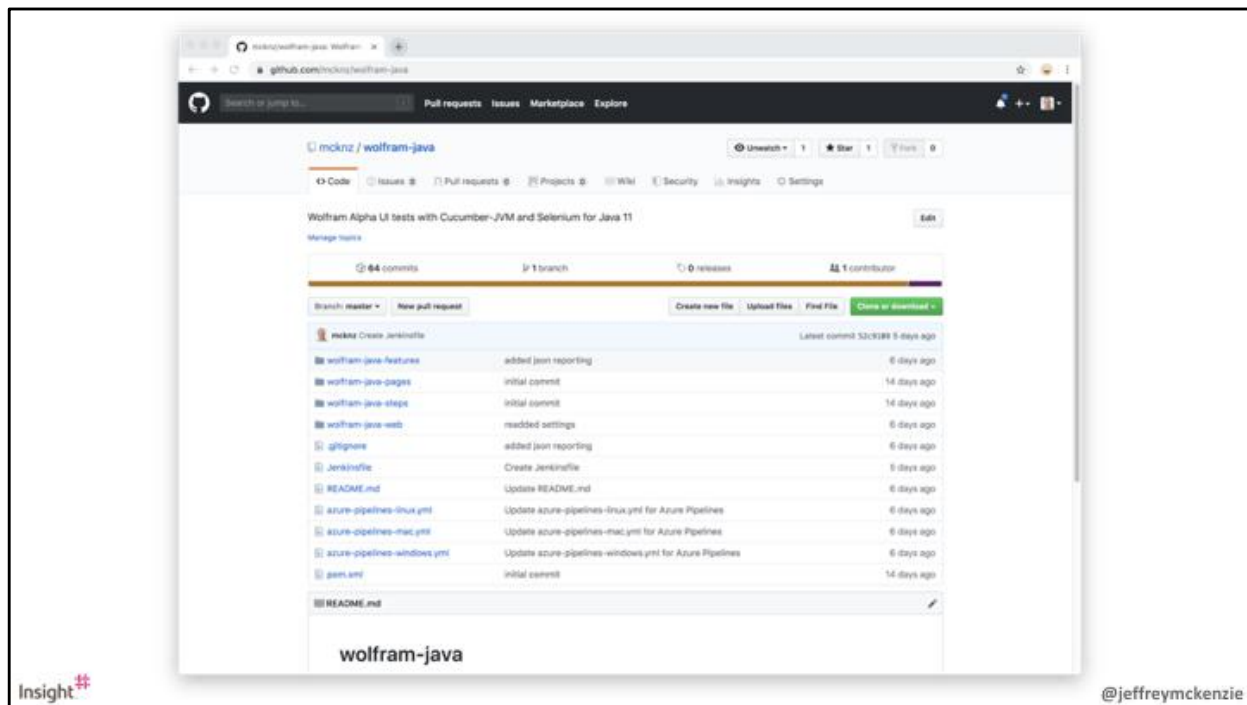
⚙️ Insight.

We need to look at both.

All of the code is available on my github, but....



Here is the .NET project



And the java project
(I will have links to these at the end)

@jeffreymckenzie

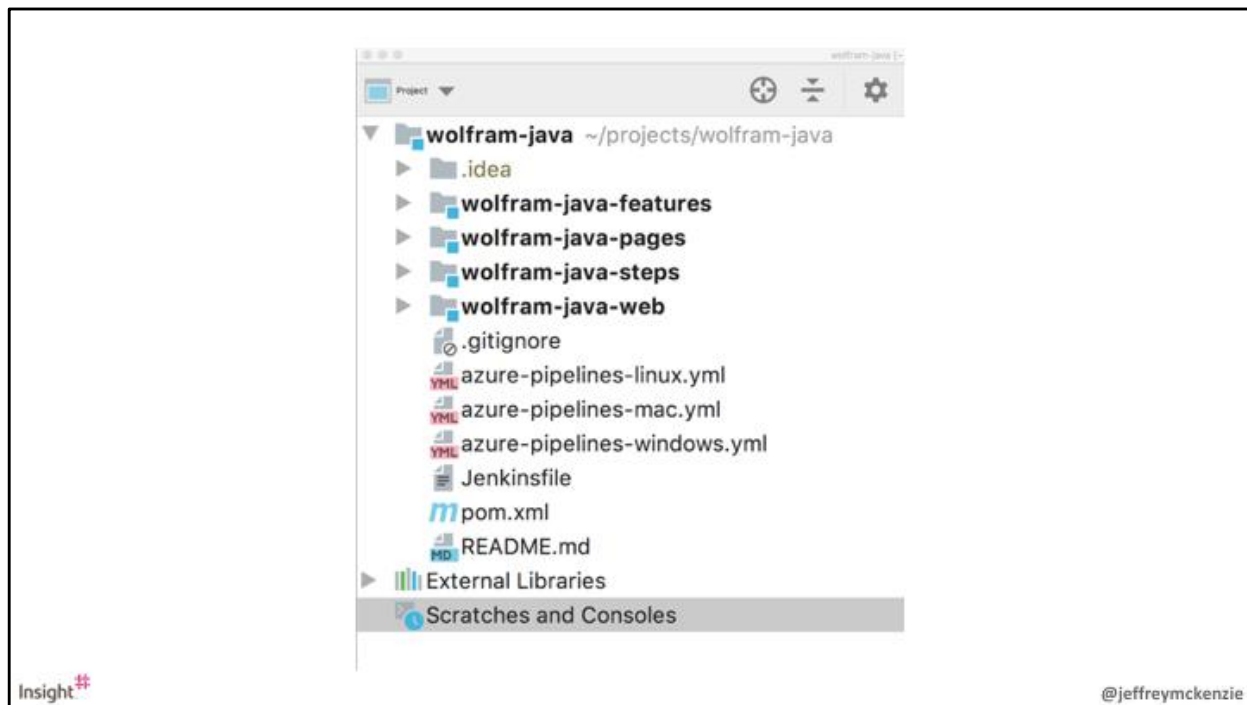
YMMMV

Insight.

With this particular design....
I just want to say your mileage may vary.

I want you to look at the code for ideas,
And to understand the concepts.

Not all of it may work with your existing test projects,
But hopefully there should be something for you
To take away.
[invariably, there will be or you will have something better...]
The good news is.-- no tests, this can be your baseline



So this is our java project,


as it appears in IntelliJ...

our four modules..

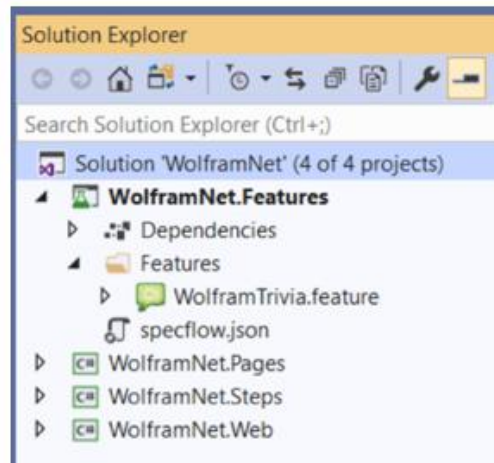
features, steps, pages, and web



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0"
3         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/
5         <modelVersion>4.0.0</modelVersion>
6
7         <groupId>com.mcknz.wolfram</groupId>
8         <artifactId>wolfram-java</artifactId>
9         <version>1.0-SNAPSHOT</version>
10        <packaging>pom</packaging>
11
12        <modules>
13            <module>wolfram-java-features</module>
14            <module>wolfram-java-steps</module>
15            <module>wolfram-java-web</module>
16            <module>wolfram-java-pages</module>
17        </modules>
18
19        <properties>
20            <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
21            <java.version>11</java.version>
22            <maven.compiler.source>${java.version}</maven.compiler.source>
23            <maven.compiler.target>${java.version}</maven.compiler.target>
24            <maven.compiler.version>3.8.1</maven.compiler.version>
25        </properties>
26    </project>
```

Insight  @jeffreymckenzie

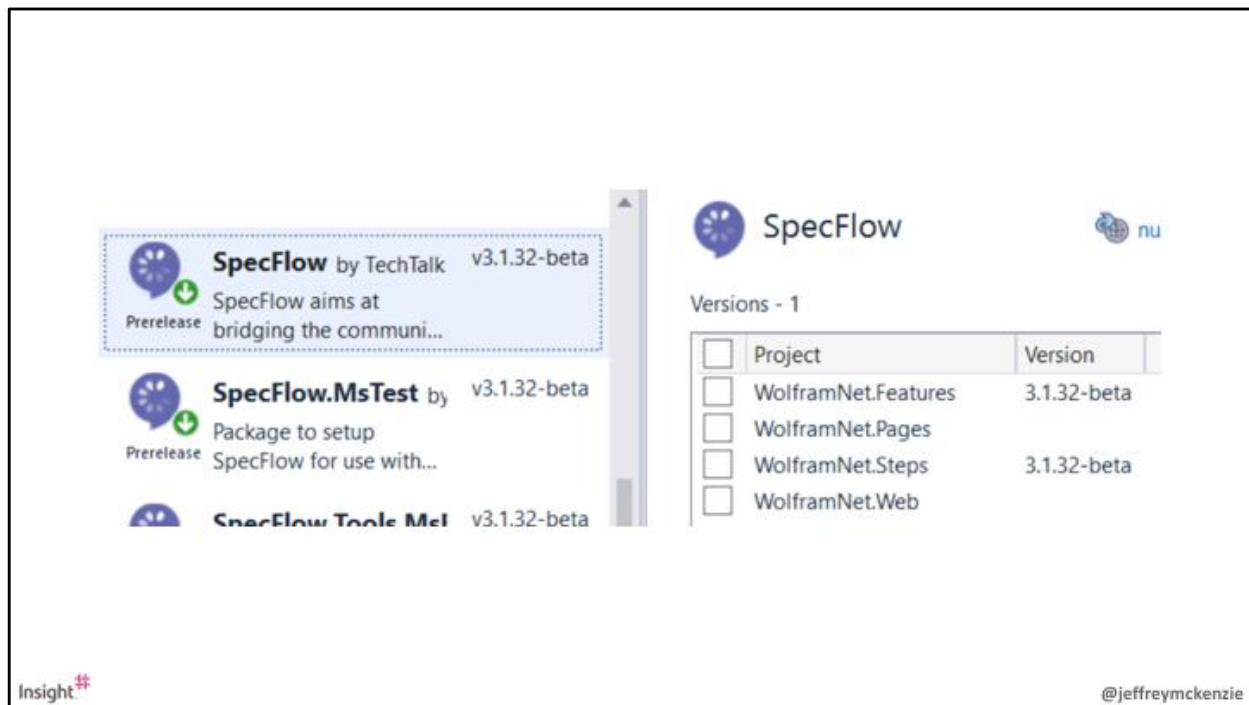
Here is the POM file,
It is a multi-module maven project,
I know not everyone is a maven fan
But I find it a convenient way to
Integrate all this into CI,
Which is the whole point.



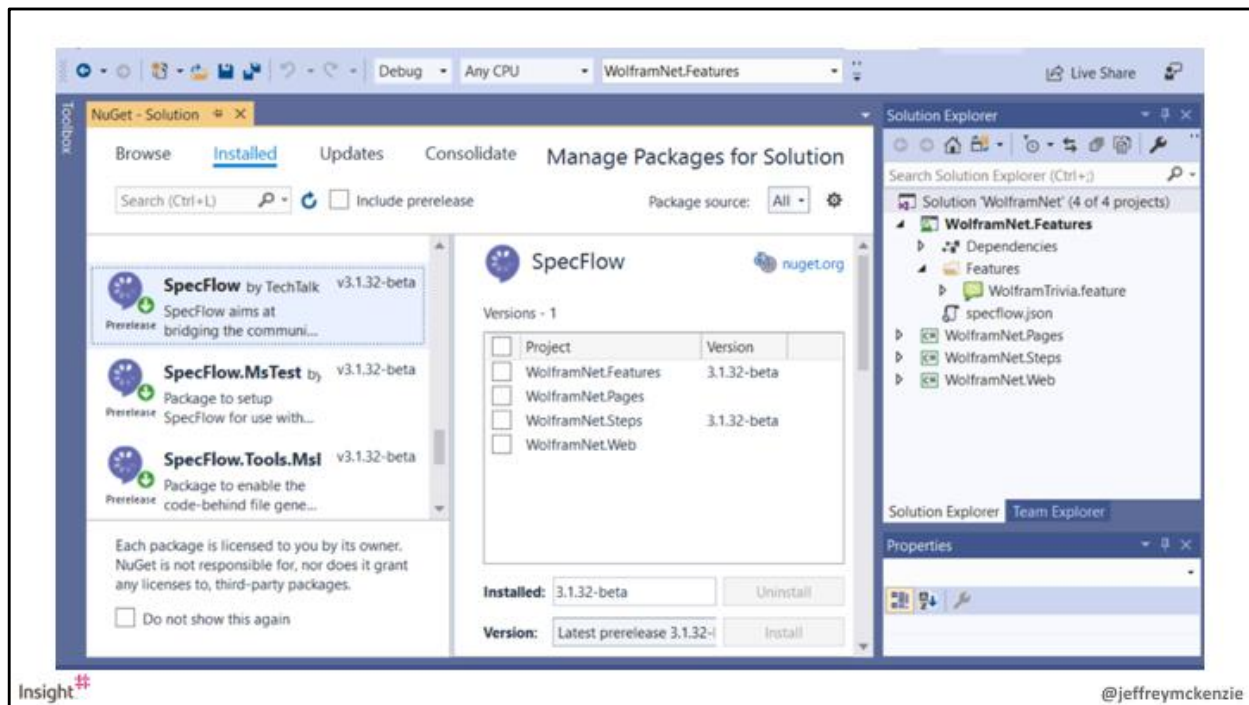
In the dotnet core solution,
We have 4 projects –
The first, WolframNet.Features,
Has really just two files,
The feature file,
And the specflow.json file.



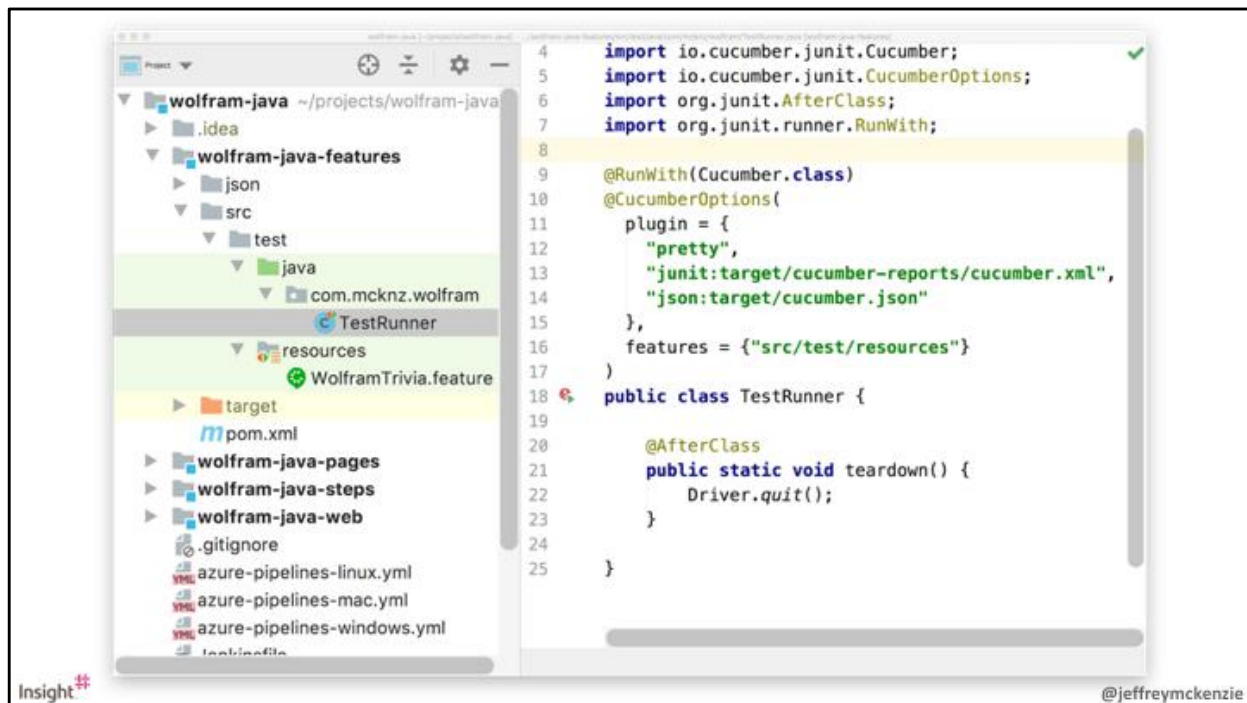
The specflow json file is new to SpecFlow version 3
Which supports dotnet core
Main thing here = if we want to separate
The features from the steps,
We indicate here where the features should look
For the steps files – in this case, the steps project.



Specflow and cucumber don't include test runners –
In dotnet I'm using the mstest provider which is available in Nuget,
And you can see it's referenced in the features and steps projects – using the beta
here but the stable will work fine



So that's what it all looks like in Visual Studio with Nuget on the left
And the solution explorer on the right



In java, we are going to use Junit as our test runner,

so we need to create a test runner class for that

...References to cucumber and junit

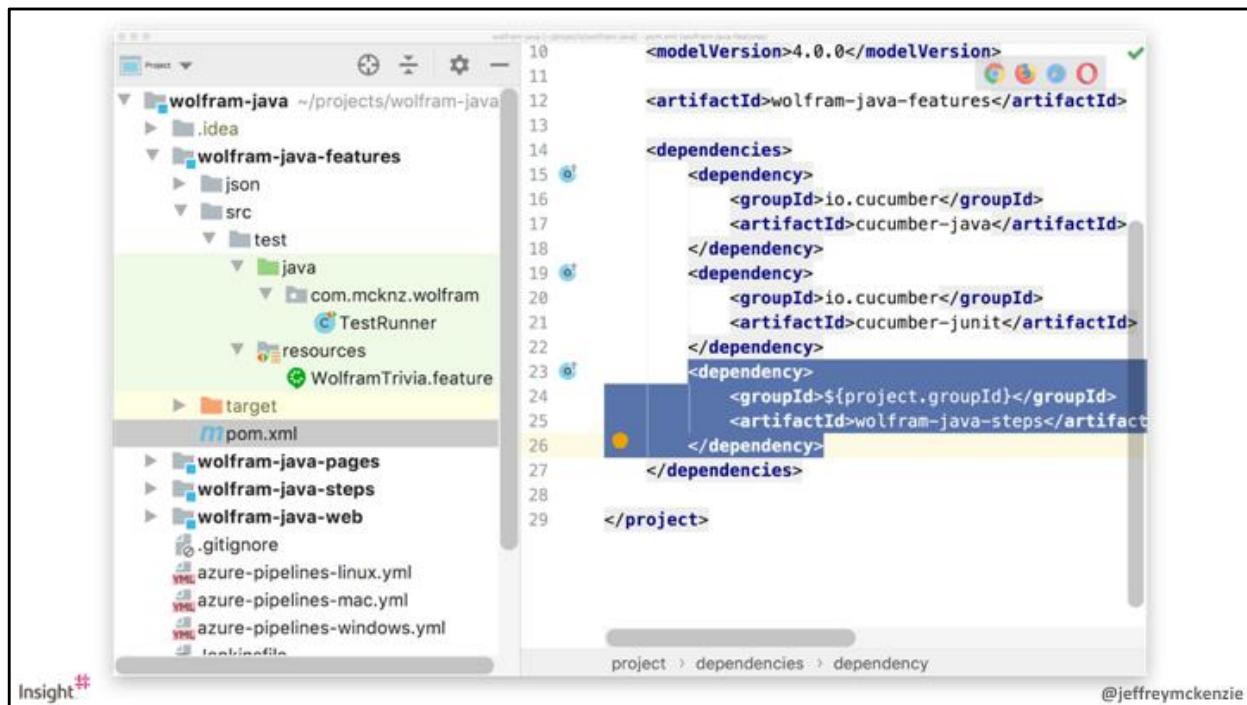
...attributes

PLUGIN – these are important because we will see these in our CI reports

-- pretty, enables formatted console output

-- cucumber xml, Junit reports

-- json, will use in Jenkins to generate nice HTML reports



In order for cucumber to know where the step files are,
We need to include the steps module
As a dependency in our features pom file

Feature: wolfram Trivia

Scenario: wolfram Knows Movies

Given I navigate to wolfram Alpha

When I ask "what movies star both

Billy Dee Williams and John Ratzenberger"

Then wolfram Alpha answers "?????????"

So let's revisit our gherkin,
And see how the solutions proceed from there

```

using Microsoft.VisualStudio.TestTools.UnitTesting;
using TechTalk.SpecFlow;
using WolframNet.Pages;

namespace WolframNet.Steps
{
    [Binding]
    0 references | Jeff McKenzie, 5 days ago | 1 author, 6 changes
    public class WolframTriviaSteps
    {
        readonly WolframPage page = new WolframPage();

        [Given]
        0 references | Jeff McKenzie, 5 days ago | 1 author, 4 changes
        public void Given_I_navigate_to_Wolfram_Alpha() {
            page.Go();
        }

        [When]
        0 references | Jeff McKenzie, 8 days ago | 1 author, 2 changes
        public void When_I_ask_QUESTION(string question) {
            page.Ask(question);
        }

        [Then]
        0 references | Jeff McKenzie, 6 days ago | 1 author, 4 changes
        public void Then_Wolfram_Alpha_answers_ANSWER(string answer) {
            Assert.IsTrue(page.GetAnswer().Contains(answer));
        }
    }
}

```

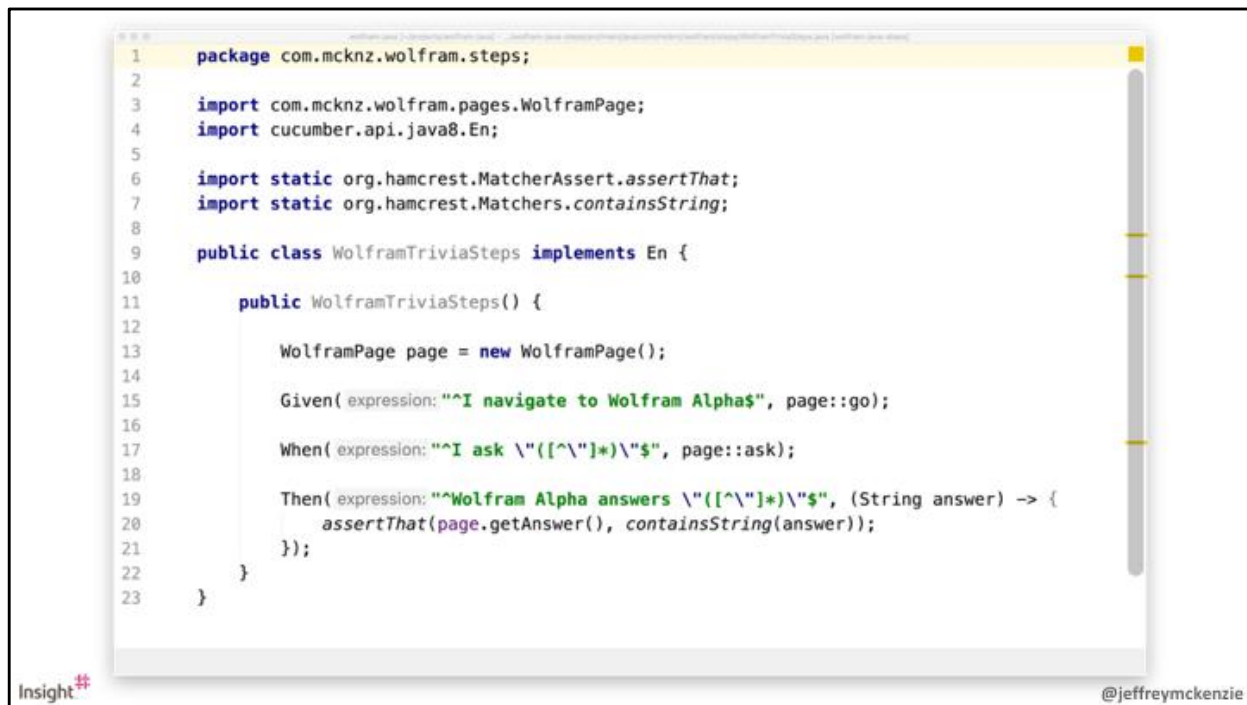
Insight 

@jeffreymckenzie


Here's what the dotnet steps class looks like –
 What I like about specflow is that you can map steps
 To gherkin using the underscore notation.
 Normally you use Regular Expressions
 (which are always fun)
 But here you can use an allcaps parameter
 To pass the value into the method

First thing, instance of the page...

Then, very simple interactions
 (steps don't know anything about page details)

A screenshot of a code editor window showing a Java class named WolframTriviaSteps. The code is written in a light-themed editor with line numbers on the left. The package is com.mcknz.wolfram.steps. It imports WolframPage, cucumber.api.java8.En, and static methods from hamcrest.Matchers. The class implements the En interface and contains a constructor that initializes a WolframPage object and defines three steps: 'I navigate to Wolfram Alpha', 'I ask', and 'Wolfram Alpha answers'.

```
1 package com.mcknz.wolfram.steps;  
2  
3 import com.mcknz.wolfram.pages.WolframPage;  
4 import cucumber.api.java8.En;  
5  
6 import static org.hamcrest.MatcherAssert.assertThat;  
7 import static org.hamcrest.Matchers.containsString;  
8  
9 public class WolframTriviaSteps implements En {  
10  
11     public WolframTriviaSteps() {  
12  
13         WolframPage page = new WolframPage();  
14  
15         Given( expression: "I navigate to Wolfram Alpha$", page::go);  
16  
17         When( expression: "I ask \"([^\"]*)\"$", page::ask);  
18  
19         Then( expression: "Wolfram Alpha answers \"([^\"]*)\"$", (String answer) -> {  
20             assertThat(page.getAnswer(), containsString(answer));  
21         });  
22     }  
23 }
```

Insight  @jeffreymckenzie

Here is the steps java class that corresponds.
We have our imports, and then simply our constructor
What I like about cucumber-jvm is the lambda constructions –

Instead of creating separate methods,
you have these very concise statements

Again, just calling methods in that page object

So, let's see what that page object does...

```

using WolframNet.Web;

namespace WolframNet.Pages
{
    3 references | Jeff McKenzie, 6 days ago | 1 author, 5 changes
    public class WolframPage : WebPage
    {
        1 reference | Jeff McKenzie, 13 days ago | 1 author, 1 change
        public WolframPage() : base("https://www.wolframalpha.com/")
        {
        }

        1 reference | Jeff McKenzie, 13 days ago | 1 author, 1 change
        public void Go()
        {
            NavigateToStartingUrl();
        }
    }
}

```

Insight 

@jeffreymckenzie

Again, this is very simple...

Most of the work is done in the base class web page.

The url for the page here, because it is specific to the page,
and is passed up to the base page constructor...


```
package com.mcknz.wolfram.pages;

import com.mcknz.wolfram.web.WebPage;

public class WolframPage extends WebPage {

    public WolframPage() {
        super( url: "https://www.wolframalpha.com/");
    }

    public void go() {
        navigateToStartingUrl();
    }
}
```

Insight

@jeffreymckenzie

Here is the java version, pretty much the same...

our GO method is simply a call to
The Navigate to starting URL method
So let's take a look at the base page.

```
package com.mcknz.wolfram.web;

import org.openqa.selenium.*;
import org.openqa.selenium.interactions.Actions;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

abstract public class WebPage {

    private final WebDriver driver;
    private final Settings settings;
    private final String url;

    protected WebPage(String url) {
        driver = Driver.get();
        settings = Driver.getSettings();
        this.url = url;
    }
}
```

In the base page we are setting three things –
The Selenium WebDriver, A settings object, And the url.

```
1 reference | Jeff McKenzie, 14 days ago | 1 author, 1 change  
public void NavigateToStartingUrl()  
{  
    driver.Navigate().GoToUrl(url);  
}
```

So once we are in the base page,
That call to NavigateToStartingURL
Is what actually makes the call to Selenium,
It's just encapsulated.

```

using OpenQA.Selenium;
using OpenQA.Selenium.Interactions;
using OpenQA.Selenium.Support.UI;
using SeleniumExtras.WaitHelpers;
using System;

namespace WolframNet.Web
{
    2 references | Jeff McKenzie, 8 days ago | 1 author, 4 changes
    public abstract class WebPage
    {
        private readonly IWebDriver driver;
        private readonly Settings settings;
        private readonly string url;

        1 reference | Jeff McKenzie, 8 days ago | 1 author, 1 change
        public WebPage(string url)
        {
            driver = Driver.Get();
            settings = Driver.GetSettings();
            this.url = url;
        }
    }
}

```

Insight 

@jeffreymckenzie

Here is the .NET version...

You'll notice that the driver and settings
 Are static instances, and both of these give us the flexibility
 To work with UI testing both locally and in the CI.
 So, let's look at the Driver class.

```

using OpenQA.Selenium;
using System.Diagnostics;

namespace WolframNet.Web {
    4 references | Jeff McKenzie, 6 days ago | 1 author, 11 changes
    public class Driver
    {
        private static readonly WebDriverFactory factory = new WebDriverFactory();
        private static IWebDriver driver;
        private static readonly Settings settings;

        0 references | Jeff McKenzie, 13 days ago | 1 author, 1 change
        static Driver()
        {
            settings = new Settings();
        }
    }
}

```

In the driver class, we grab an instance of
 A webdriver factory,
 And in the static constructor we get a new instance
 Of the settings class.
 So let's look at the factory, then the settings,
 And then we'll recap what we have.

```

2 references | Jeff McKenzie, 6 days ago | 1 author, 9 changes
class WebDriverFactory
{
    1 reference | Jeff McKenzie, 13 days ago | 1 author, 1 change
    internal IWebDriver GetDriver(Settings settings)
    {
        return GetWebDriver(settings);
    }

    1 reference | Jeff McKenzie, 6 days ago | 1 author, 3 changes
    private IWebDriver GetWebDriver(Settings settings)
    {
        switch (settings.DriverType)
        {
            case WebDriverType.Chrome:
                return GetChromeDriver(settings, false);
            case WebDriverType.ChromeHeadless:
                return GetChromeDriver(settings, true);
        }
        throw new ArgumentException("Unable to create Driver");
    }
}

```

Insight

@jeffreymckenzie

Here's the factory, which essentially exposes a method to return a WebDriver of a specific type. I am only using Chrome here, so this could be expanded to include IE, Firefox, or whatever other WebDriver implementations you want.

The idea is to decouple creation of the driver.

The important thing is to be able to specify headless or not. What is a headless browser?

[2 references](#) | Jeff McKenzie, 6 days ago | 1 author, 8 changes

```
private IWebDriver GetChromeDriver(Settings settings,
                                   bool isHeadless)
{
    ChromeOptions options = new ChromeOptions();
    if (isHeadless)
    {
        options.AddArguments("headless", "window-size=1920,1080");
    }

    return new ChromeDriver(settings.DriverPath, options);
}
```

Insight 

@jeffreymckenzie

Here is the specific method that returns the ChromeDriver,
To get a headless instance we simply pass in a couple options.
For headless in a UI I find that setting the window size
To a high resolution helps avoid issues with finding specific elements.
So this returns an instance of chromedriver,
Given a path to the driver as specified in the settings class.

```
private WebDriver getChromeDriver(Settings settings,
                                   boolean isHeadless) {
    ChromeOptions options = new ChromeOptions();
    if(isHeadless) {
        options.addArguments("--headless", "--window-size=1920,1080");
    }
    System.setProperty("webdriver.chrome.driver", settings.getDriverPathAndName());
    return new ChromeDriver(options);
}
```

Insight

@jeffreymckenzie

Here is the java method – one difference is that in java
You set the system property webdriver.chrome.driver
To the path and name of the chromedriver.


```
static WebDriver get() {  
    if(driver == null) {  
        driver = factory.getDriver(settings);  
    }  
    return driver;  
}
```

Then in both .NET and java there's one static instance
That exists for the entirety of the test run

```
public class Settings
{
    private readonly IConfiguration config;
    private WebDriverType driverType;
    private TimeSpan? pageTimeout;
    private string driverPath;

    1 reference | Jeff McKenzie, 8 days ago | 1 author, 3 changes
    public Settings() {
        config = new ConfigurationBuilder()
            .SetBasePath(Directory.GetCurrentDirectory())
            .AddJsonFile("appsettings.json", true)
            .AddEnvironmentVariables()
            .Build();
    }
}
```

Insight

@jeffreymckenzie

The settings class is really just a wrapper around configuration.

We have our private members, and in the constructor,

We create a dotnet core configuration builder.

Two main things = bringing in appsettings.json

-- allows us to use this locally

And the add environment variables

-- which allows us to use this in CI

App settings....

```
private void loadResourceSettings(ClassLoader loader) {
    Properties p = new Properties();
    try {
        final Enumeration<URL> systemResources =
            (loader == null ? ClassLoader.getSystemClassLoader() : loader)
                .getResources(name: "settings.properties");
        while (systemResources.hasMoreElements()) {
            p.load(systemResources.nextElement().openStream());
        }
    } catch (IOException ex) {
        // ignore if running as part of CI process
        // because all properties will be supplied externally
    }
    props.putAll(p);
}
```

Insight

@jeffreymckenzie

Same thing in java, except we are loading a properties file that exists
In the resources directory called "settings.properties"



```
appsettings.json* ×  
Schema: http://json.schemastore.org/appsettings  
{  
  "driverType": "Chrome",  
  "pageTimeout": 60,  
  "driverPath": "c:\\selenium\\"  
}
```

These are our local settings


In the appsettings.json file

Driver Type is chrome b/c we want to see what happens

Page timeout = 60 seconds to wait

Driver path, location of the driver executable

```
wolfram-java [~/projects/wolfram-java] ~ .../wolfram-java-web/src/main/resources/settings.properties {  
1  #local environment properties  
2  driverType=CHROME  
3  pageTimeout=60  
4  driverPathAndName=/usr/local/bin/chromedriver
```

Insight  @jeffreymckenzie

Similar settings in our java example...

WolframPage
- WebPage
- ChromeDriver
- Settings
- appsettings.json
or
- settings.properties

 Insight.

To recap, this is what we have...

We just walked up the instantiation or initialization chain

For all our objects.

Main thing we have is a page object with settings
that can be automated with Chrome



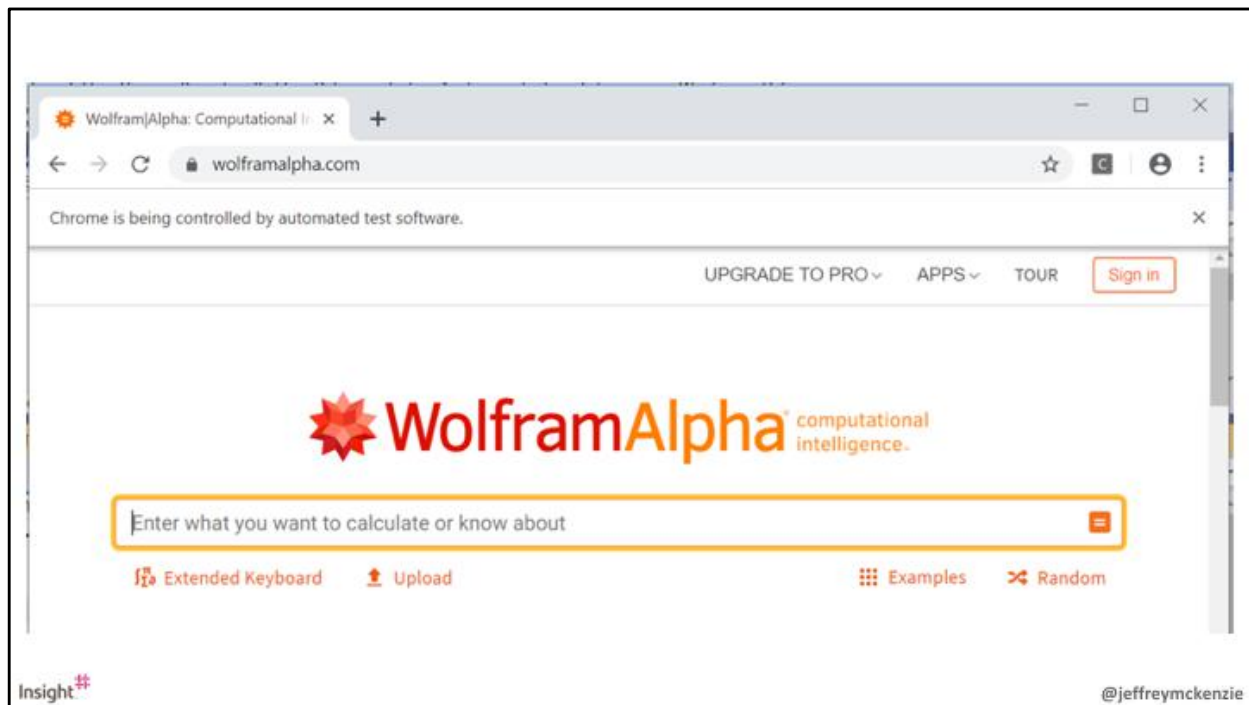
So when our page instantiates, we get an instance of the driver, which kicks off chrome....
"this is automated"

```
using WolframNet.Web;

namespace WolframNet.Pages
{
    3 references | Jeff McKenzie, 6 days ago | 1 author, 5 changes
    public class WolframPage : WebPage
    {
        1 reference | Jeff McKenzie, 13 days ago | 1 author, 1 change
        public WolframPage() : base("https://www.wolframalpha.com/")
        {
        }

        1 reference | Jeff McKenzie, 13 days ago | 1 author, 1 change
        public void Go()
        {
            NavigateToStartingUrl();
        }
    }
}
```

Then in our Go method we have
Navigate to starting URL...



Which brings up the wolfram alpha page...
[notice that placeholder text]:
“Enter what you want to know about”

Feature: wolfram Trivia

Scenario: wolfram Knows Movies

Given I navigate to wolfram Alpha

when I ask "what movies star both
Billy Dee Williams and John Ratzenberger"

Then wolfram Alpha answers "?????????"

Our next line is "when I ask"

And if you recall that maps to the ask method of the page...

```
When( expression: "^I ask \"([^\"]*)\"$", page::ask);
```

[When]

0 references | Jeff McKenzie, 8 days ago | 1 author, 2 changes

```
| public void When_I_ask_QUESTION(string question) {  
|     page.Ask(question);  
| }
```

The steps look like this...
java on top, .net on the bottom

1 reference | Jeff McKenzie, 8 days ago | 1 author, 1 change

```
public void Ask(string question)
{
    string questionXPath =
        "//input[@placeholder='Enter what you want" +
        "to calculate or know about']";
    string questionButtonXPath = "//button[@type='submit']";

    EnterTextByXPath(questionXPath, question);
    ClickByXPath(questionButtonXPath);
}
```

Insight 

@jeffreymckenzie

[The page object code looks about the same in both .NET and java
So I will use the .NET example here]
so we are grabbing the question input text box using Xpath,
Finding the element that has the specific placeholder text,
And entering our question in that input text box
The page doesn't know anything about Selenium,
So the base class handles the actual call in
The EnterTextByXpath method.....

[1 reference](#) | Jeff McKenzie, 8 days ago | 1 author, 1 change

```
protected void EnterTextByXPath(string xpath,
                                string text)
{
    WaitUntilElementClickableByXPath(xpath)
        .SendKeys(text);
}
```

one note...
on this method called WaitUntilElementClickableByXPath,
Which generates a web element based on the
WebDriverWait object....

```

4 references | Jeff McKenzie, 8 days ago | 1 author, 1 change
private IWebElement WaitUntilElementClickableByXPath(string xpath) {
    return WaitUntilElementClickable(By.XPath(xpath));
}

1 reference | Jeff McKenzie, 8 days ago | 1 author, 2 changes
private IWebElement WaitUntilElementClickable(By locator) {
    return GetWait().Until(
        ExpectedConditions.ElementToBeClickable(locator));
}

1 reference | Jeff McKenzie, 8 days ago | 1 author, 3 changes
private WebDriverWait GetWait() {
    WebDriverWait wait =
        new WebDriverWait(driver, settings.PageTimeout);
    wait.IgnoreExceptionTypes(
        typeof(StaleElementReferenceException),
        typeof(NoSuchElementException)
    );
    return wait;
}

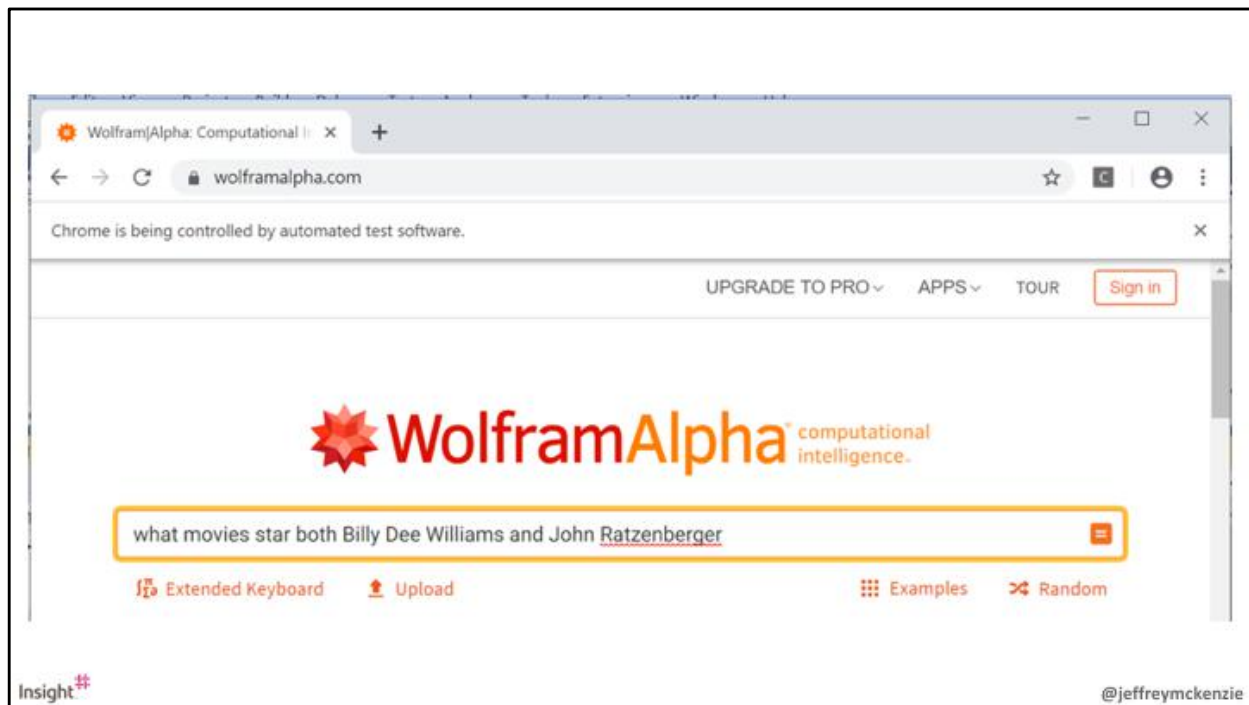
```

Insight

@jeffreymckenzie

The method at the top, which returns a web element,
calls the method Underneath it,
And that method in turn calls the GetWait method.

Not to get too into the weeds, but this is an
Implicit wait in selenium, which essentially
Tells the code to wait until the element is ready
To be interacted with.



So that enters our text in the input box...

1 reference | Jeff McKenzie, 8 days ago | 1 author, 1 change

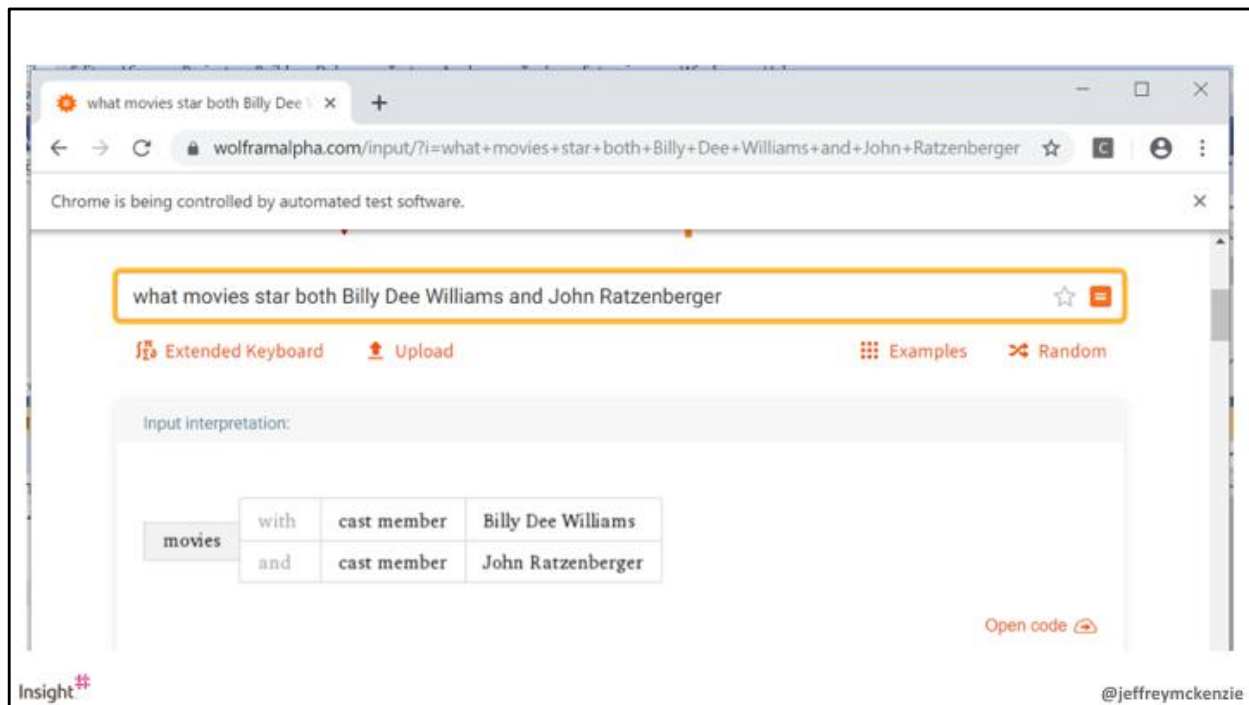
```
public void Ask(string question)
{
    string questionTextXPath =
        "//input[@placeholder='Enter what you want" +
        "to calculate or know about']";
    string questionButtonXPath = "//button[@type='submit']";

    EnterTextByXPath(questionTextXPath, question);
    ClickByXPath(questionButtonXPath);
}
```

Insight 

@jeffreymckenzie

Next we find the submit button and perform
A click on it, using a very similar pattern
Of calling the base page and using a wait....



That starts Wolfram Alpha doing it's thing...
Processing the statement....

Feature: wolfram Trivia

Scenario: wolfram Knows Movies

Given I navigate to wolfram Alpha

when I ask "what movies star both

Billy Dee Williams and John Ratzenberger"

Then wolfram Alpha answers "?????????"

Then our last line is..

Then Wolfram Answers

Of course in order for this to be a valid UI test
(to get our test to pass)

we have to know the answer in advance,

But that's not much fun for trivia purposes....

```
Then( expression: "^Wolfram Alpha answers \"([^\"]*)\"$", (String answer) -> {  
    assertThat(page.getAnswer(), containsString(answer));  
});
```

[Then]

0 references | Jeff McKenzie, 6 days ago | 1 author, 4 changes

```
public void Then_Wolfram_Alpha_answers_ANSWER(  
    string answer) {  
    Assert.IsTrue(page.GetAnswer().Contains(answer));  
}
```

So we will finish up with the Gherkin “Then” statement,
[again, java on top, .NET on bottom]
Which gets the answer from the page object,
And parses it to see if it matches what we want....

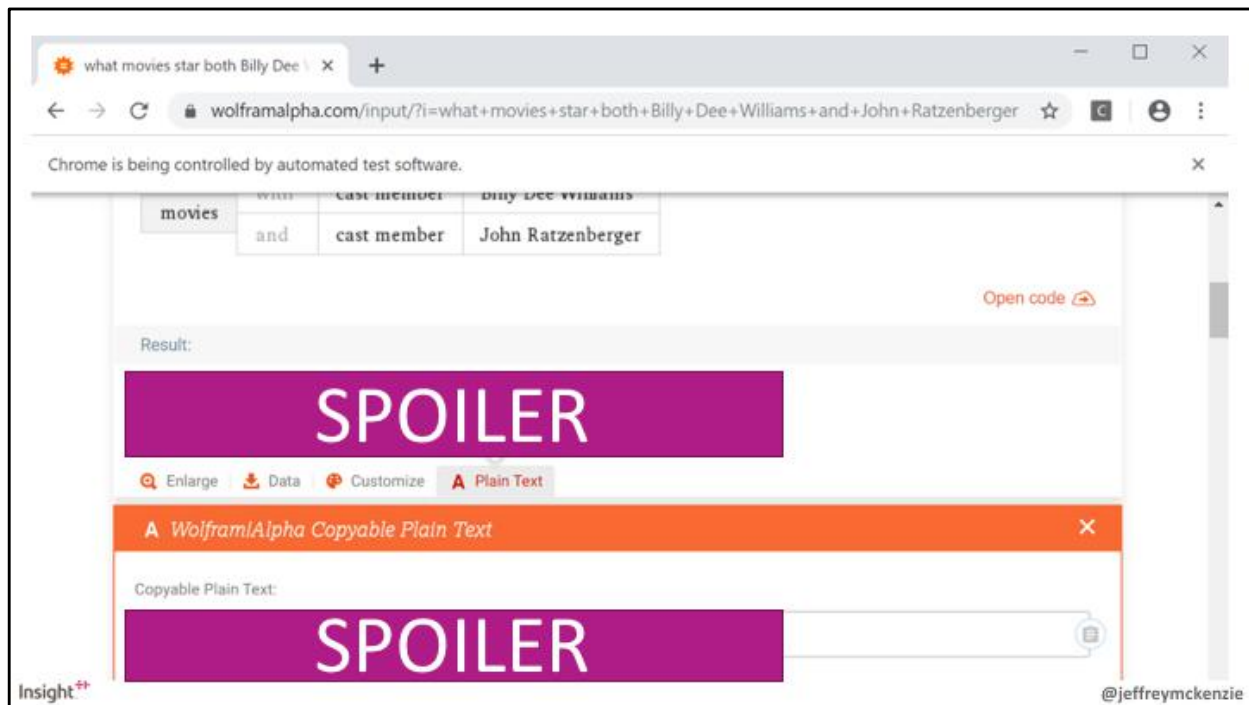
1 reference | Jeff McKenzie, 8 days ago | 1 author, 3 changes

```
public string GetAnswer()
{
    string answerDivXPath =
        "//*[@id=\"root\"]/div/div/main/div[3]" +
        "/div/section[1]/section[2]/div[2]";
    string answerButtonXPath =
        "//span[text() = 'Plaintext']";
    string answerTextXPath =
        "//div[@aria-describedby='tooltip5']" +
        "/button/span";
    MouseoverByXPath(answerDivXPath);
    ClickByXPath(answerButtonXPath);
    return GetTextByXPath(answerTextXPath);
}
```

Insight

@jeffreymckenzie

In the object we are interacting with 3 elements
To do this, a mouseover,
A click, and then returning the text value.
[a little ugly]



Drum roll please....

Result:

Star Wars: Episode V – The Empire Strikes Back

 Enlarge |  Data |  Customize | **A Plain Text**

A *Wolfram|Alpha Copyable Plain Text*

Copyable Plain Text:

Star Wars: Episode V - The Empire Strikes Back



There he is, Major Bren Derlin on Hoth
Does anybody know any of his lines?

<https://starwarsblog.starwars.com/wp-content/uploads/2015/02/Major-Derlin-Headshot-1536x864-441008088411.jpg>

Your Highness, there's nothing
more we can do tonight.

The shield doors must be closed.

Insight 

@jeffreymckenzie

Your Highness, there's nothing more we can do tonight. The shield doors must be closed.

Now that we have our project working locally,
Let's get it into CI...

Q8: What is the possibility of successfully navigating an asteroid field?

Insight.

No phones –

But first, I have one more question for you...

What is the possibility of successfully navigating

An asteroid field?

Anyone?

Where is this quote from?

Again,



Star wars....

[https://commons.wikimedia.org/wiki/File:SWCA_-](https://commons.wikimedia.org/wiki/File:SWCA_-_Long_night_in_the_Millennium_Falcon_(17015135868).jpg)

[_Long_night_in_the_Millennium_Falcon_\(17015135868\).jpg](https://commons.wikimedia.org/wiki/File:SWCA_-_Long_night_in_the_Millennium_Falcon_(17015135868).jpg)

[https://upload.wikimedia.org/wikipedia/commons/8/8d/SWCA_-](https://upload.wikimedia.org/wikipedia/commons/8/8d/SWCA_-_Long_night_in_the_Millennium_Falcon_%2817015135868%29.jpg)

[_Long_night_in_the_Millennium_Falcon_%2817015135868%29.jpg](https://upload.wikimedia.org/wikipedia/commons/8/8d/SWCA_-_Long_night_in_the_Millennium_Falcon_%2817015135868%29.jpg)

William Tung from USA [CC BY-SA 2.0 (<https://creativecommons.org/licenses/by-sa/2.0/>)]

Feature: Wolfram Trivia

Background:

Given I navigate to Wolfram Alpha

Scenario: Wolfram Alpha Knows Star Wars

When I ask "the possibility of successfully navigating an asteroid field"

Then Wolfram Alpha answers "????????????????"

Scenario: Wolfram Alpha Knows Movies

When I ask "what movies star both Billy Dee Williams and John Ratzenberger"

Then Wolfram Alpha answers "Star Wars: Episode V – The Empire Strikes Back"

Insight

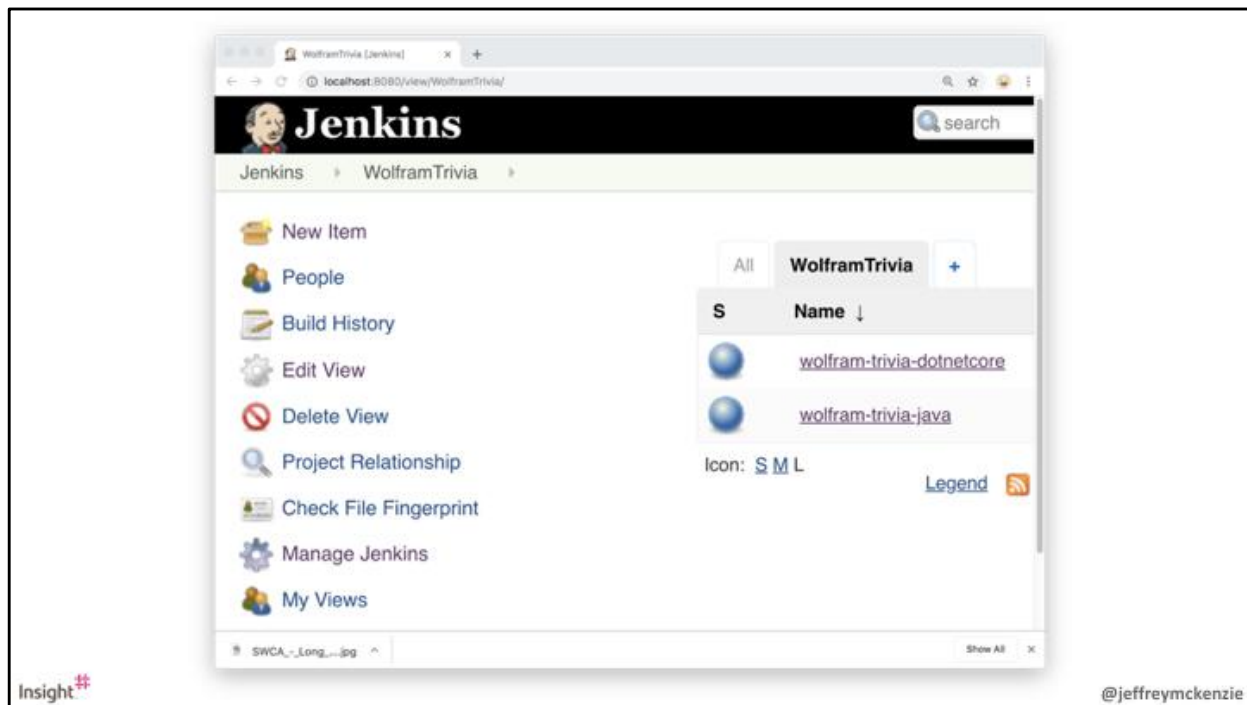
@jeffreymckenzie

For this one, we are going to use our CI process,
But we will need to change up our Gherkin a little...

We already know the answer to our first question,
So we will add a scenario to cover the second,
And also move our Given up to background,
So we don't have to repeat it.

So lets construct a pipeline to get the answer
To our question...

[Azure Devops users here? Jenkins?]

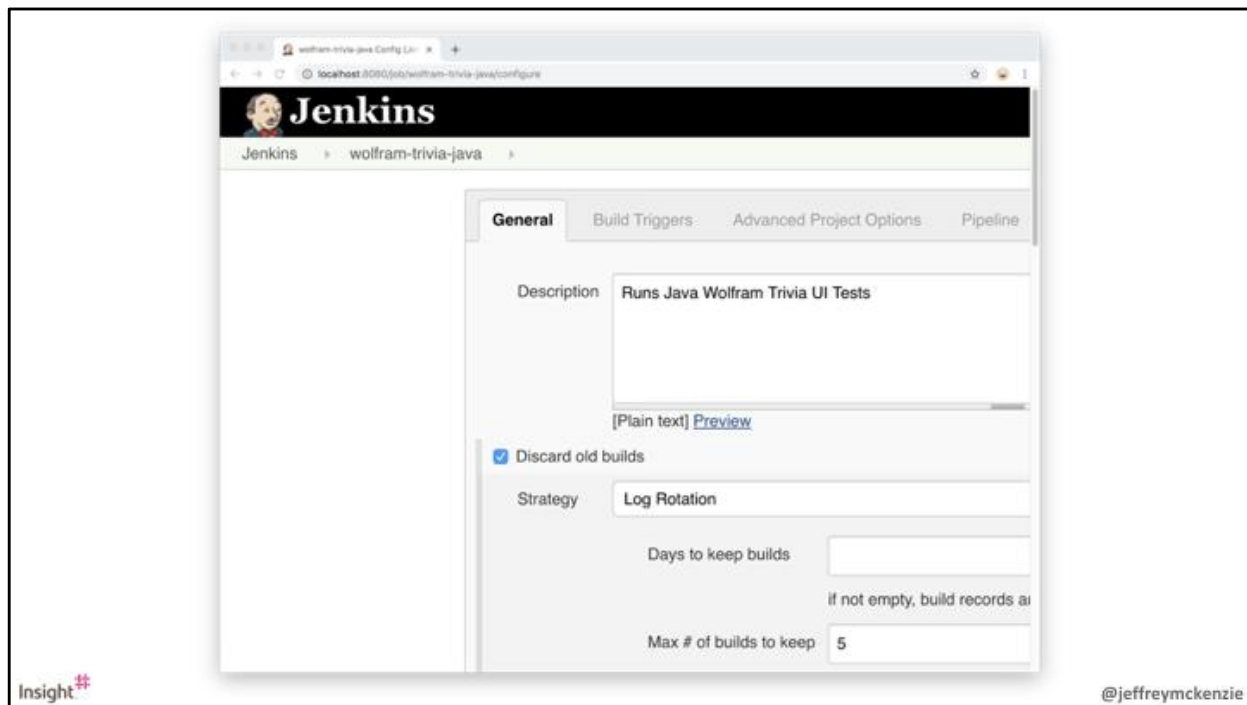


Start with Jenkins....

I have two jobs here, One for .NET and one for Java.

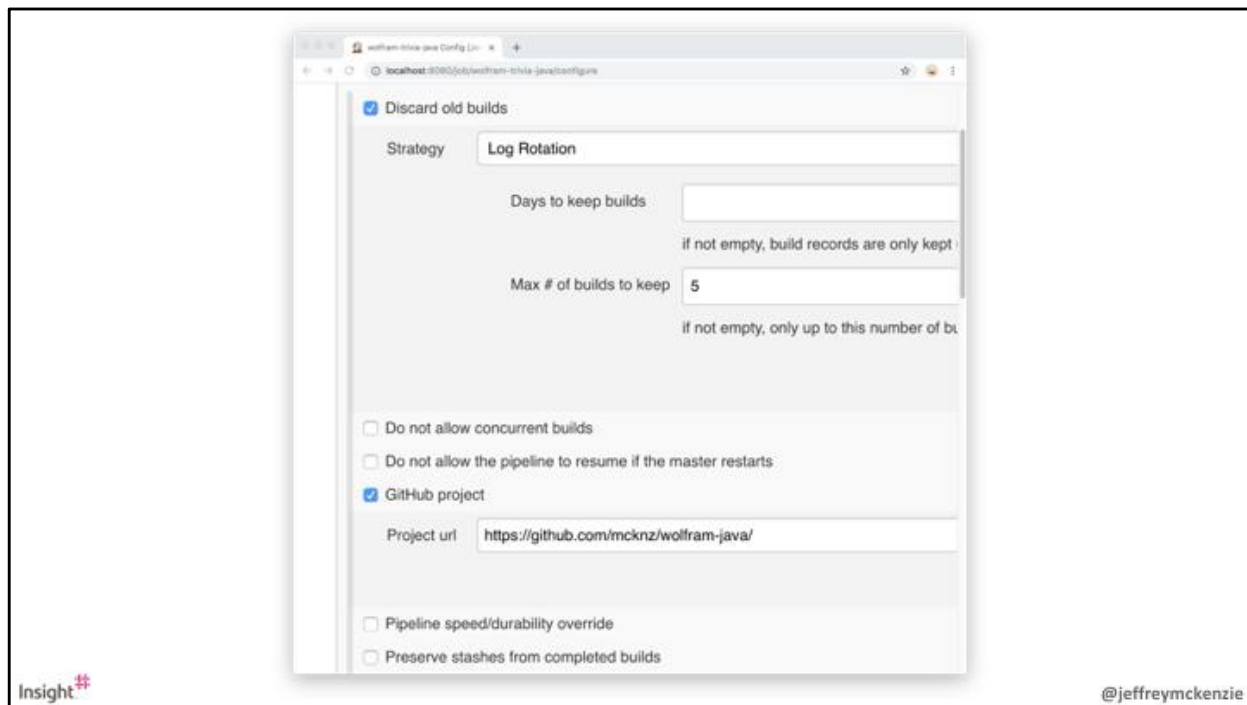
I wanted to contrast this with Azure Devops,
which is an online service,

So this is running locally on my Mac laptop
[could as easily stand this up on AWS or Azure]

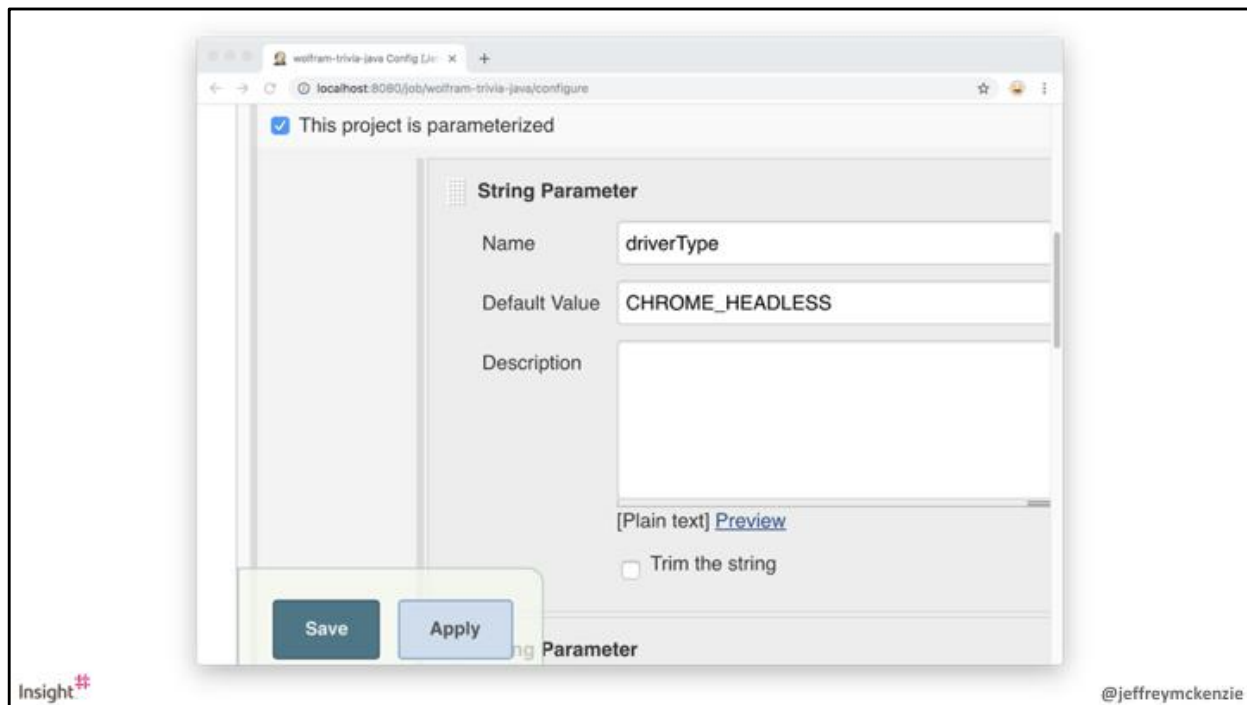


Give it a description, run java tests,

Keep the last 5 builds....



We can specify the github project here as well



Then we need to add parameters --
And this is where it starts to get different from running locally.
Remember our settings file?

Driver type we want to set to headless....

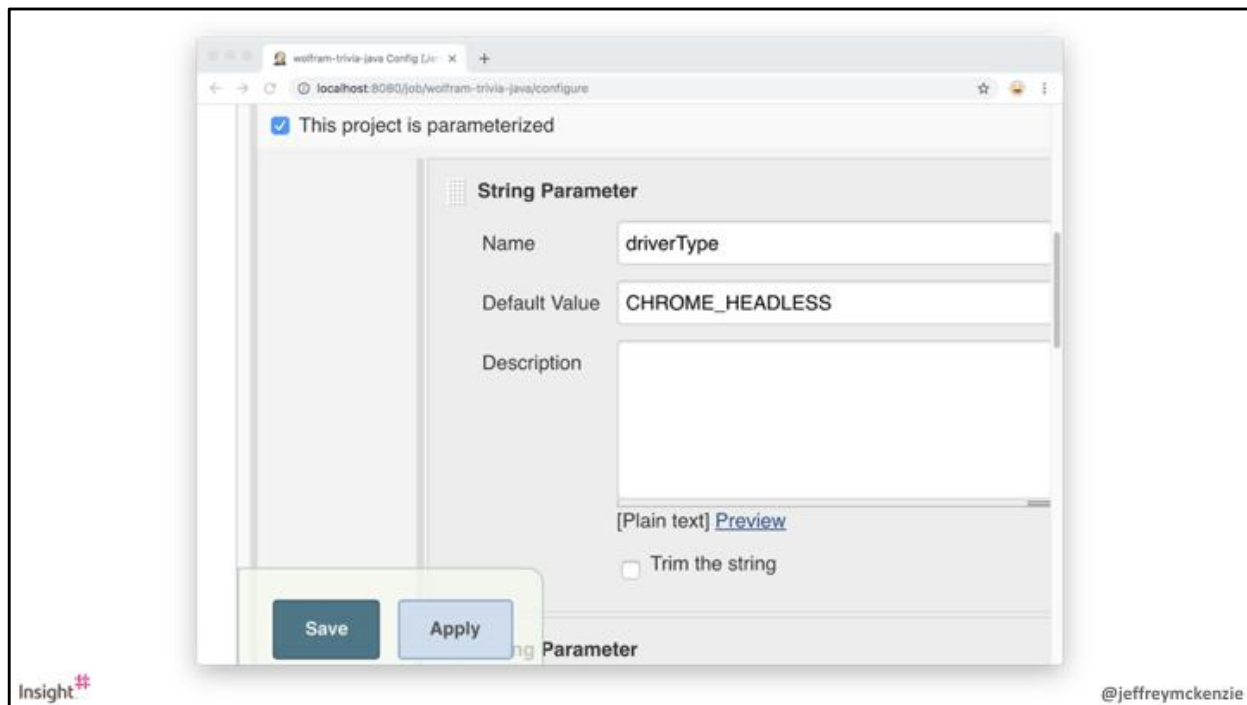
```
WebDriverType getDriverType() {  
    if(driverType == null) {  
        driverType = WebDriverType.valueOf(getProperty("driverType"));  
    }  
    return driverType;  
}  
  
private String getProperty(String key) {  
    String property = System.getProperty(key);  
    if(property == null) {  
        return props.getProperty(key);  
    }  
    return property;  
}
```

Insight 

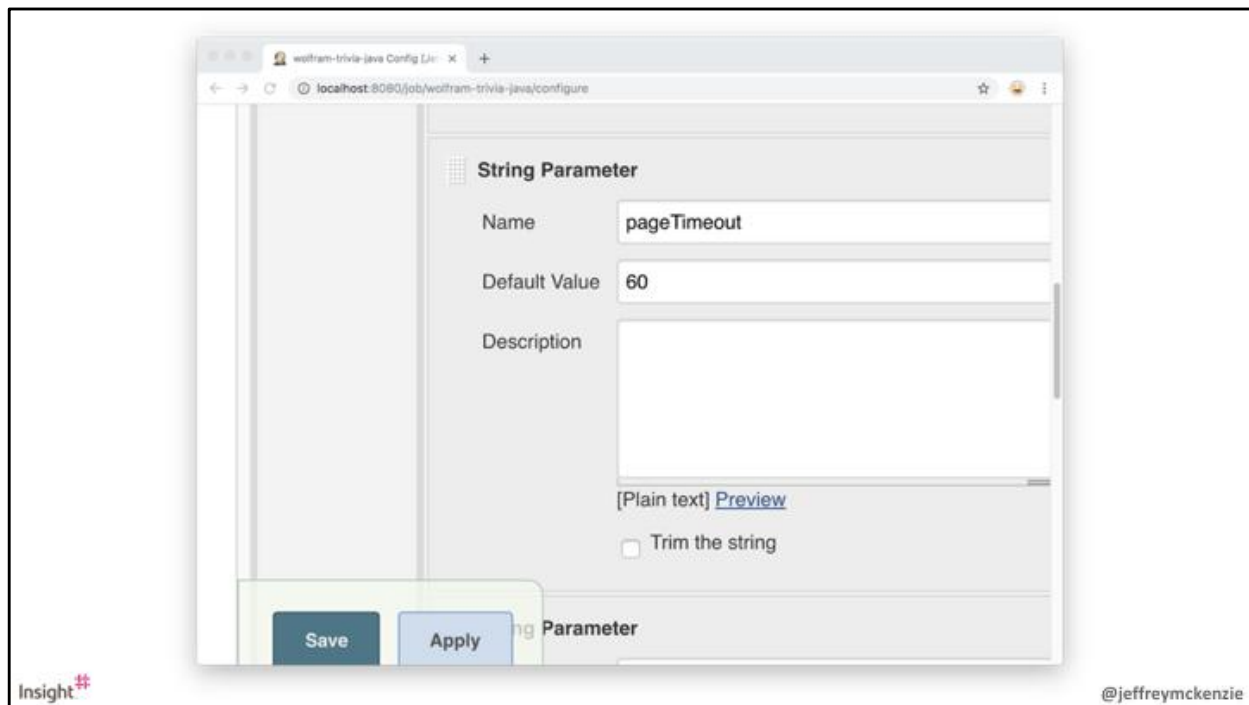
@jeffreymckenzie

So in java, we have the setter for GetDriverType,
Which returns which WebDriver instance we want,
Like Chrome or Chrome Headless

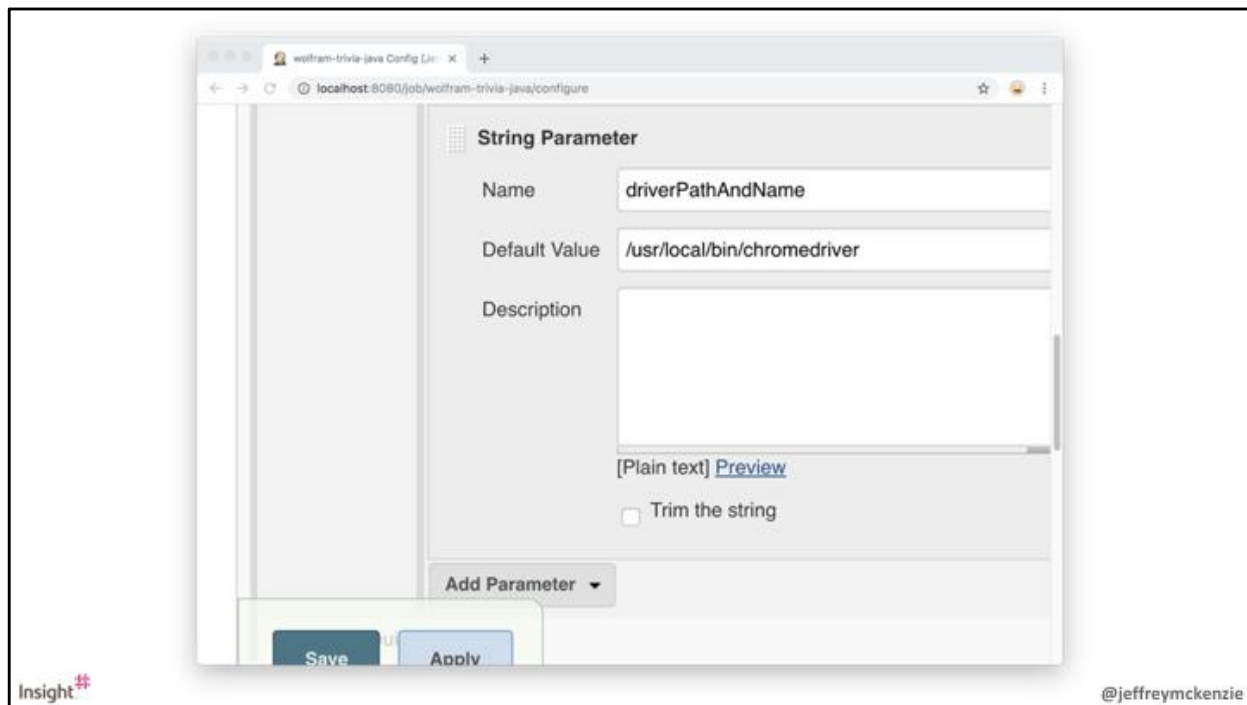
The GetProperty method first checks the system properties,
And if the system property does not exist
Then it uses the local property.



So this works well for our CI, so we can put in these values
As part of the Jenkins job,
The driver type...

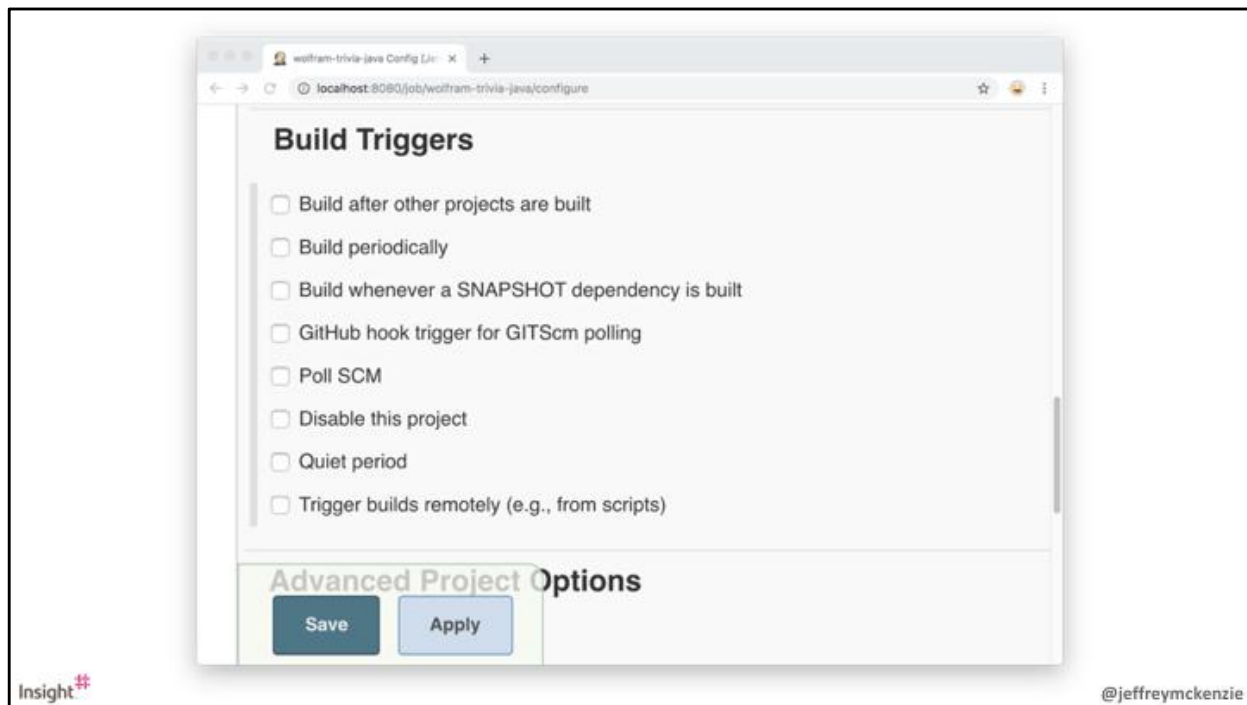


The page timeout

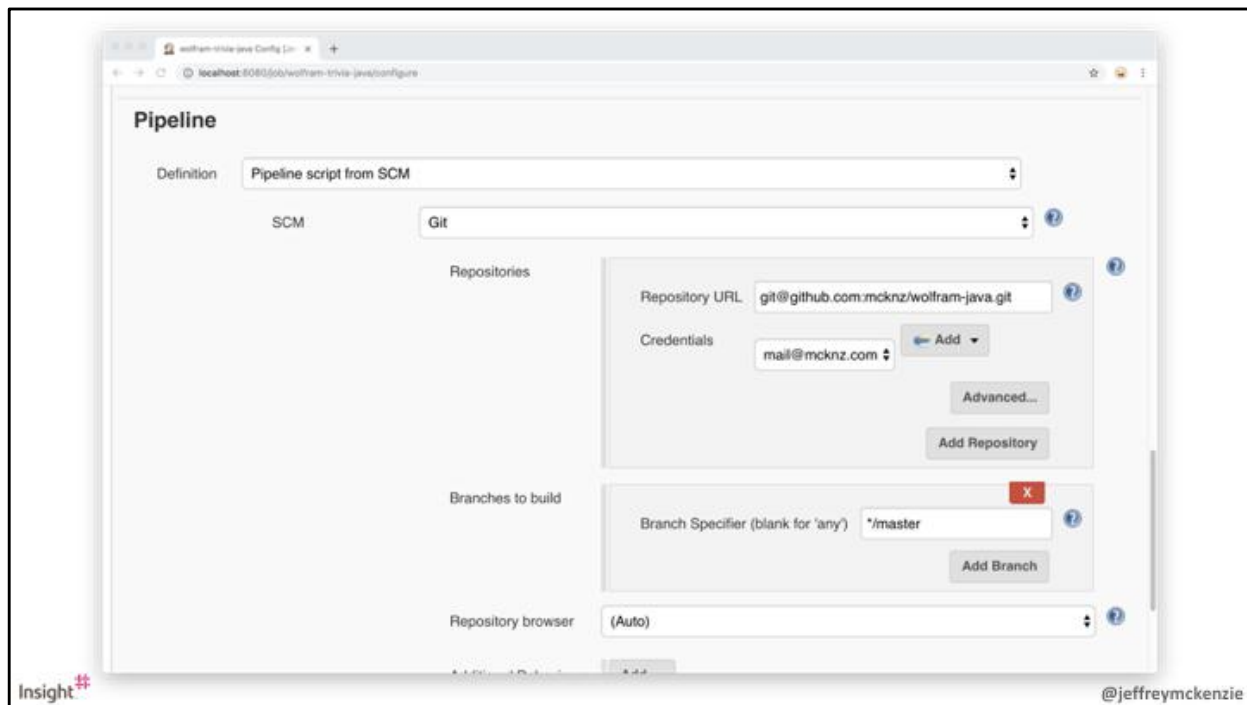


And the driver name and location...

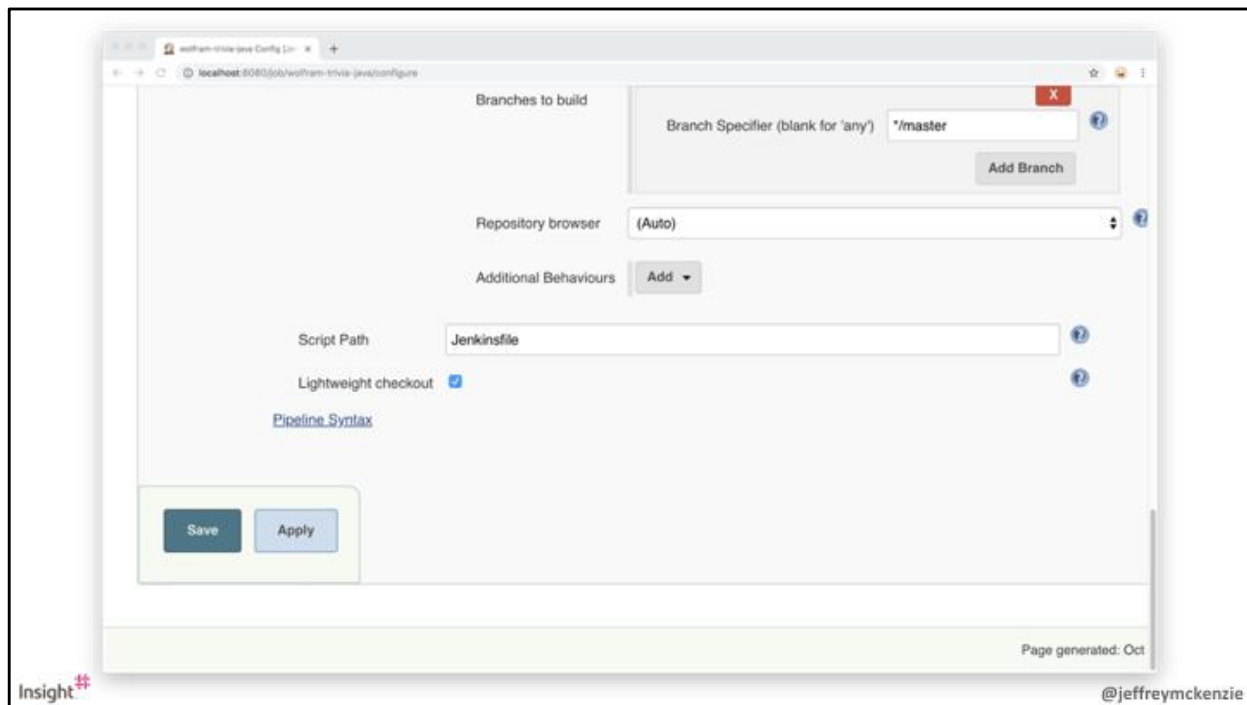
Now because I'm running these locally
The settings happen to be the same,
But if my environment were different
I could pass in different values.



There is the build triggers section—
I am building this manually for demo purposes
But you could set a trigger on this to run
Overnight, or whatever makes sense for your process.



As far as the actual pipeline script itself,
We are going to be grabbing that from source control,
Our wolfram-java repo
You can see we have specified that option,
Selected the repo.
I have also included credentials for my github account –
Those I can enter into Jenkins directly using
The credentials plugin



And you can see in the Script Path box
We specify Jenkinsfile as the script we want to pull –
That's a naming convention.

So let's look at that script

```

pipeline {
  agent any
  tools {
    maven 'Maven-3.6.0'
    jdk 'OpenJDK-11'
  }
  stages {
    stage ('Checkout code') {
      steps {
        git credentialsId: 'mcknz-ssh', url: 'git@github.com:mcknz/wolfram-java.git'
      }
    }
    stage ('Run Tests') {
      steps {
        sh '''mvn \
          -q \
          -DdriverType=${driverType} \
          -DpageTimeout=${pageTimeout} \
          -DdriverPathAndName=${driverPathAndName} \
          test'''
      }
    }
    stage ('Generate Reports') {
      steps {
        cucumber buildStatus: 'UNSTABLE',
          fileIncludePattern: '**/*.json',
          trendsLimit: 10,
          classifications: [
            [
              'key': 'Browser',
              'value': 'Chrome'
            ]
          ]
      }
    }
  }
}

```

Insight 

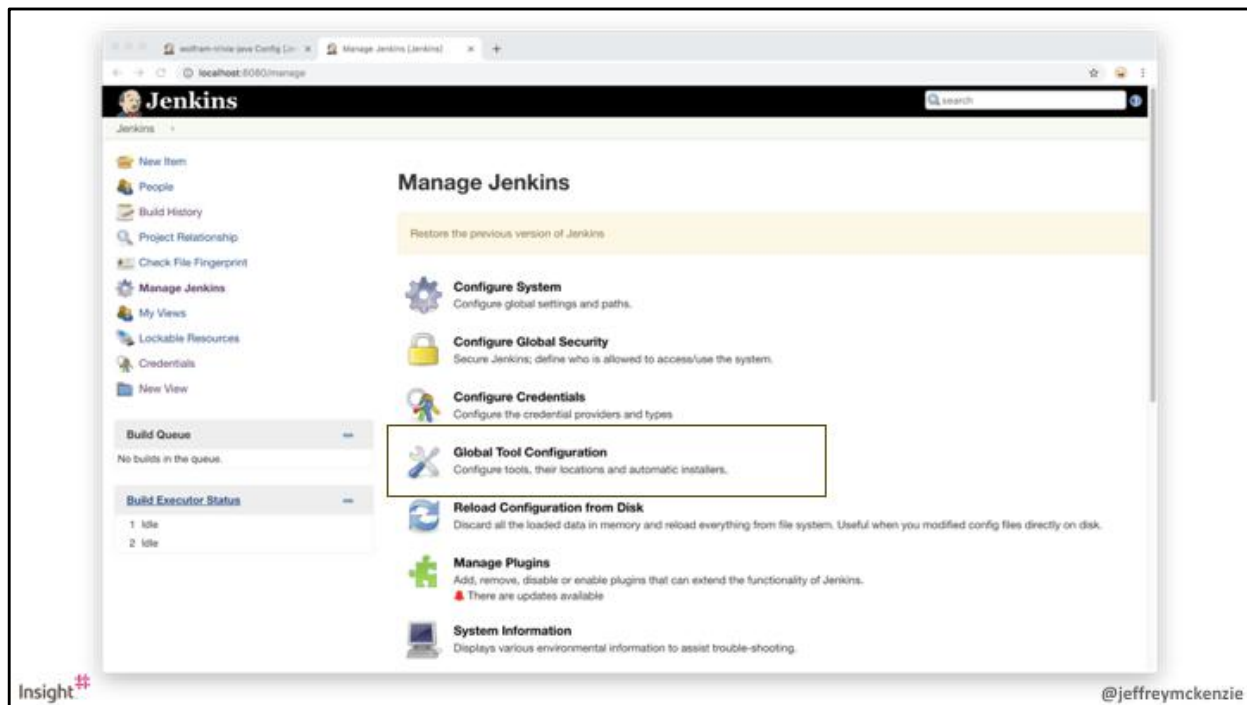
@jeffreymckenzie

Here it is, let's break that down little by little here...

```
pipeline {  
  agent any  
  tools {  
    maven 'Maven-3.6.0'  
    jdk 'OpenJDK-11'  
  }  
  . . .  
}
```

First we have our pipeline block,
Which is required...
We specify the agent we want to run this on,
In this case we say “any agent” since we are running local

Then we have our tools section –
For this we are using java 11 and maven 3.6



Those get set in Jenkins in the Global Tool Configuration
Of the Manage Jenkins page...

JDK

JDK installations

Add JDK

JDK

Name

OpenJDK-11

JAVA_HOME

/Library/Java/JavaVirtualMachines/adoptopenjdk-11.jdk/Contents/Home

☐ Install automatically

Add JDK

List of JDK installations on this system

Insight

@jeffreymckenzie

Maven

Maven installations

Add Maven

Maven

Name

Maven-3.6.0

MAVEN_HOME

/usr/local/Cellar/maven/3.6.0/libexec

☐ Install automatically

Insight

@jeffreymckenzie

```
pipeline {  
  agent any  
  tools {  
    maven 'Maven-3.6.0'  
    jdk 'OpenJDK-11'  
  }  
  . . .  
}
```

You can see those names correspond to the tools configured

```
pipeline {
  agent any
  tools { . . . }
  stages {
    stage('Checkout code') {
      steps {
        git credentialsId: 'mcknz-ssh',
          url: 'git@github.com:mcknz/wolfram-java.git'
      }
    }
    . . .
  }
}
```

Insight 

@jeffreymckenzie

Next we have the stages section, which is a way to group steps:
Each stage can have one or more steps.
Our checkout code stage has just one, which is to pull
The source code from our git repo
Using the same credentials

```

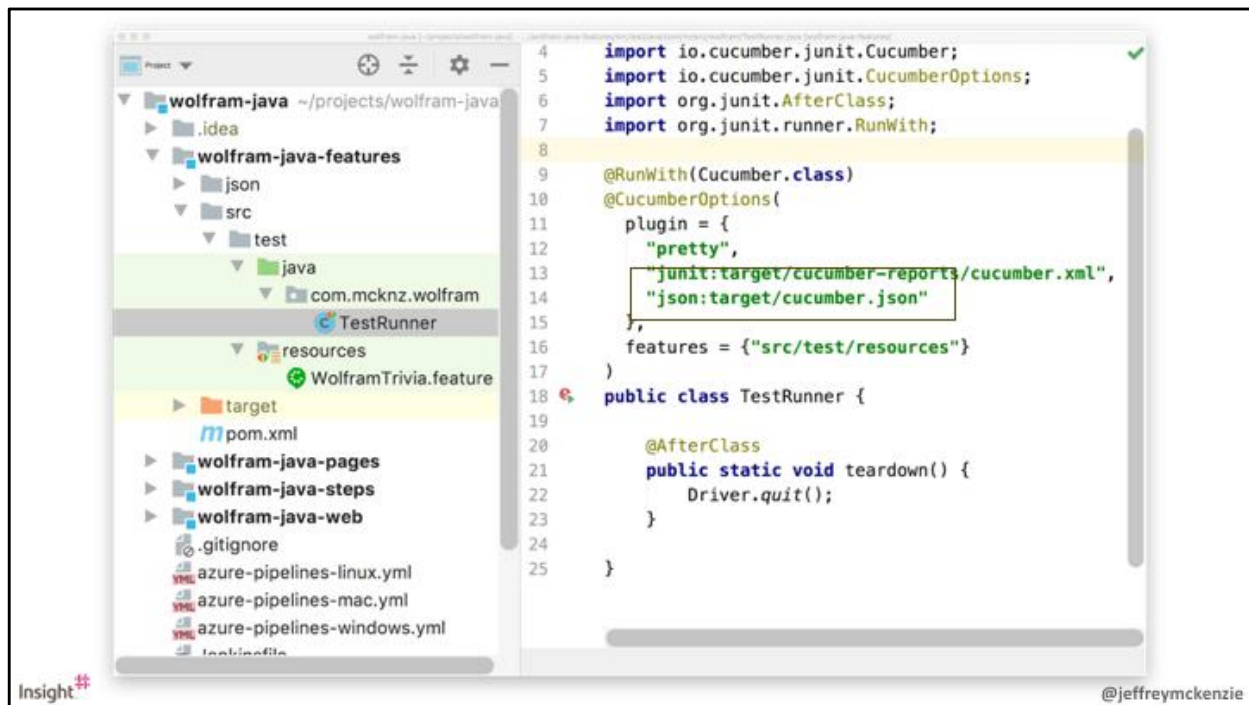
pipeline {
  stages {
    stage ('Generate Reports') {
      steps {
        cucumber buildStatus: 'UNSTABLE',
                  fileIncludePattern: '**/*.json',
                  trendsLimit: 10,
                  classifications: [[
                    'key': 'Browser',
                    'value': 'Chrome']]
      }
    }
  }
}

```

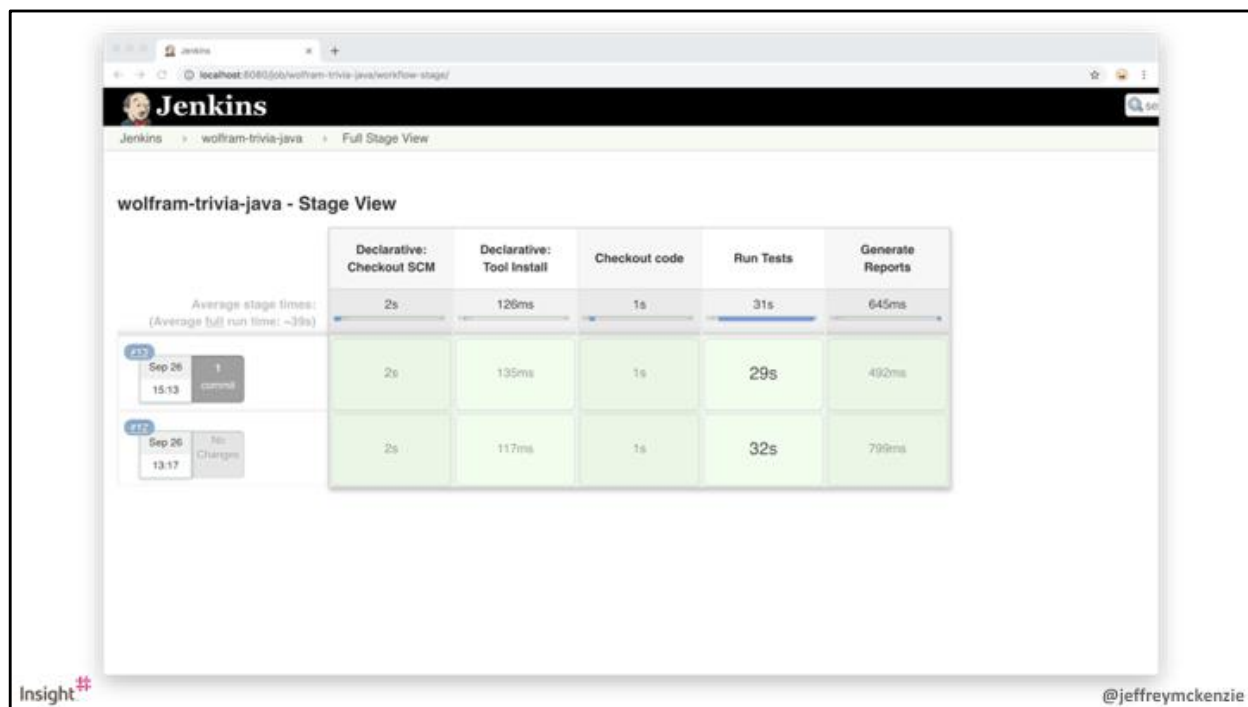
Insight

@jeffreymckenzie

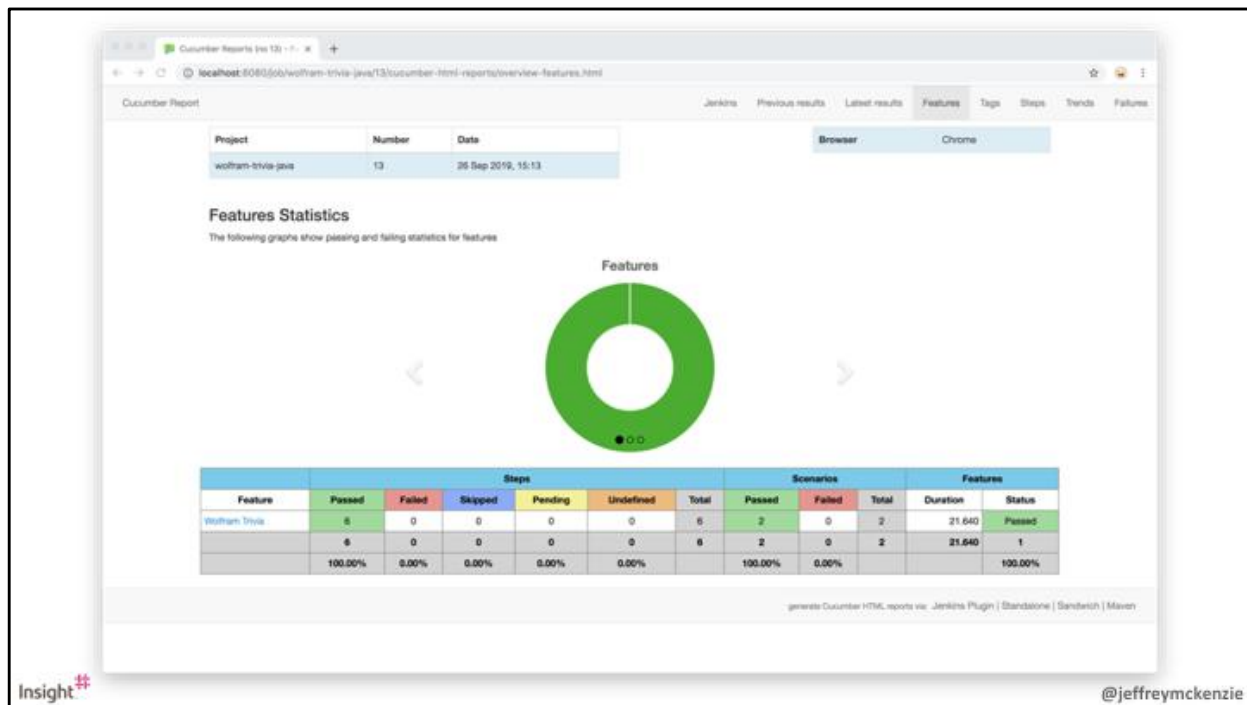
Our final stage is report generation –
This is going to look for that json file named in the testrunner class...



In cucumber options...



When we execute this, we can look in the stage view
That shows the success or failure of those different stages,
And how long they took to generate.



And we get some nice HTML reports
Showing the status of the various features and steps.

Feature Report

Feature	Steps					Total	Passed
	Passed	Failed	Skipped	Pending	Undefined		
Wolfram Trivia	6	0	0	0	0	6	2

Feature Wolfram Trivia

Background >

Scenario Wolfram Alpha Knows Star Wars ▾

Steps >


Background >

Scenario Wolfram Alpha Knows Movies ▾

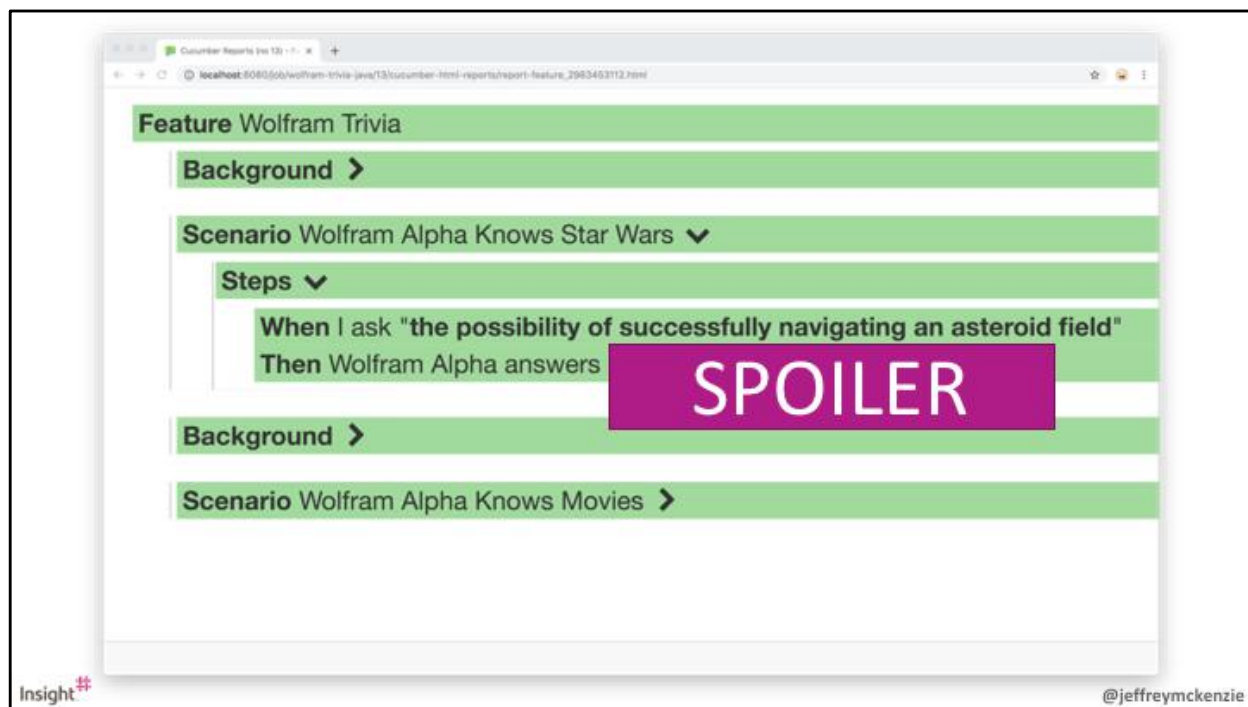
Steps ▾

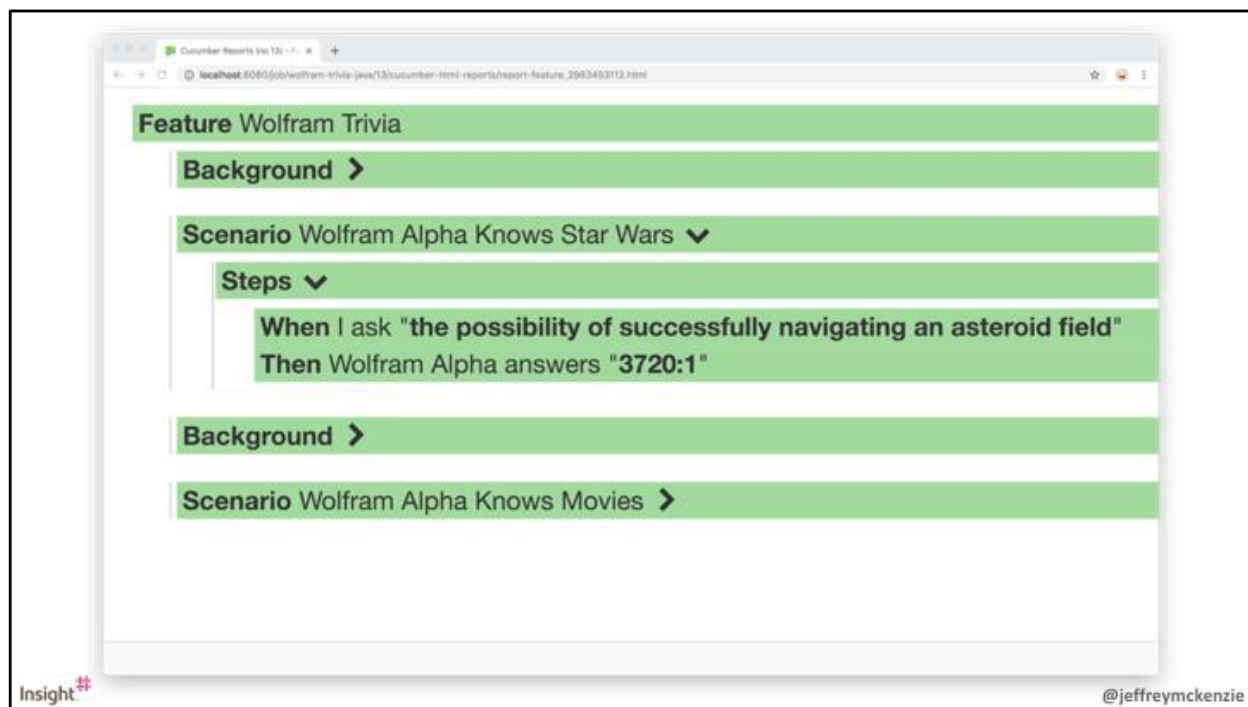
When I ask "what movies star both Billy Dee Williams and John Ratzenberger"

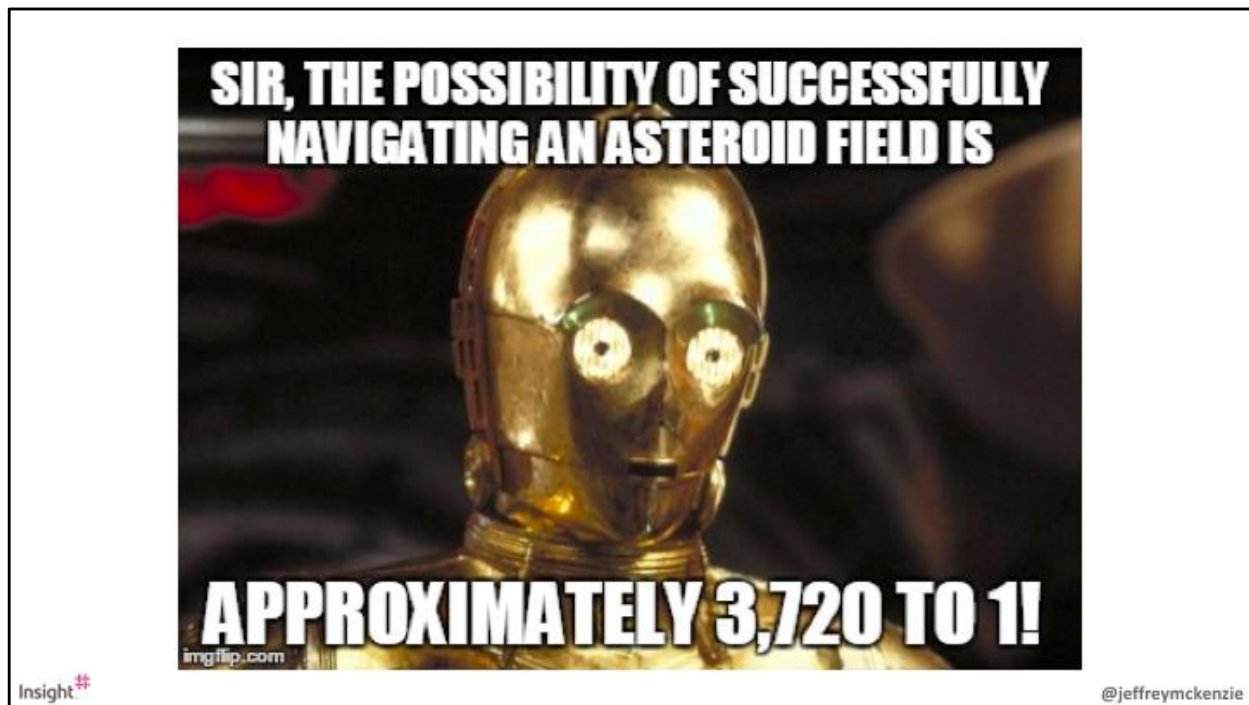
Then Wolfram Alpha answers "Star Wars: Episode V - The Empire Strikes Back"

Insight  @jeffreymckenzie

And we can drill down into the individual steps,
So drum roll please, the answer to







And... your obligatory meme.

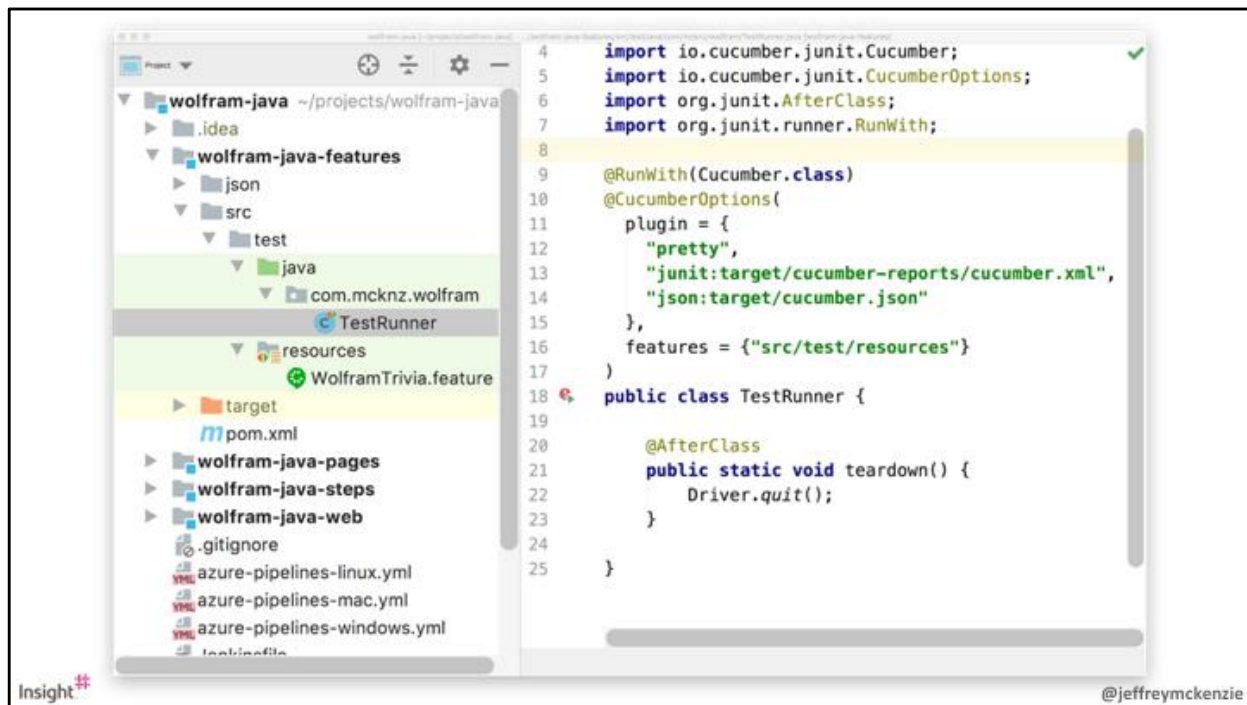
```

using TechTalk.SpecFlow;
using WolframNet.Web;

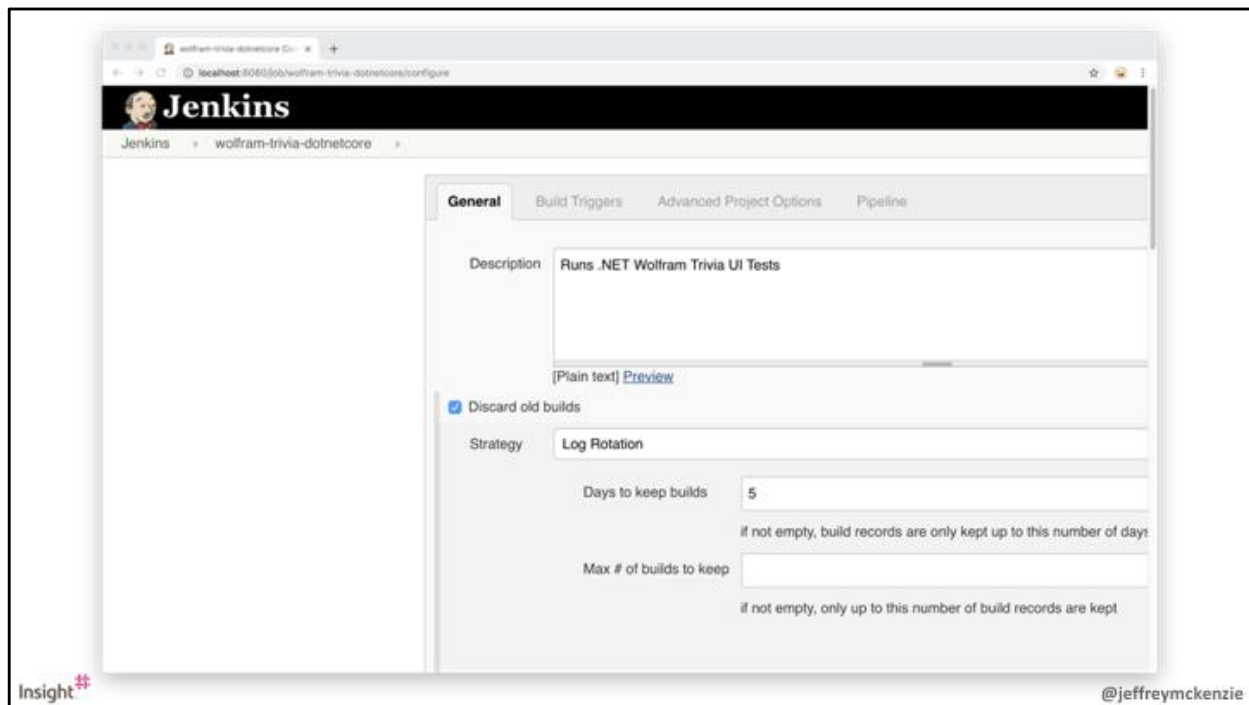
namespace WolframNet.Features
{
    [Binding]
    0 references | Jeff McKenzie, 7 days ago | 1 author, 7 changes
    public sealed class WolframHooks
    {
        [AfterTestRun]
        0 references | Jeff McKenzie, 7 days ago | 1 author, 3 changes
        public static void AfterTestRun()
        {
            Driver.Quit();
        }
    }
}

```

We were able to add another test to our gherkin
 And reused the code,
 But because we are using the same instance of the driver
 Throughout to improve performance,
 We need to close it out when we are done.
 In specflow we can add an AfterTestRun hook...



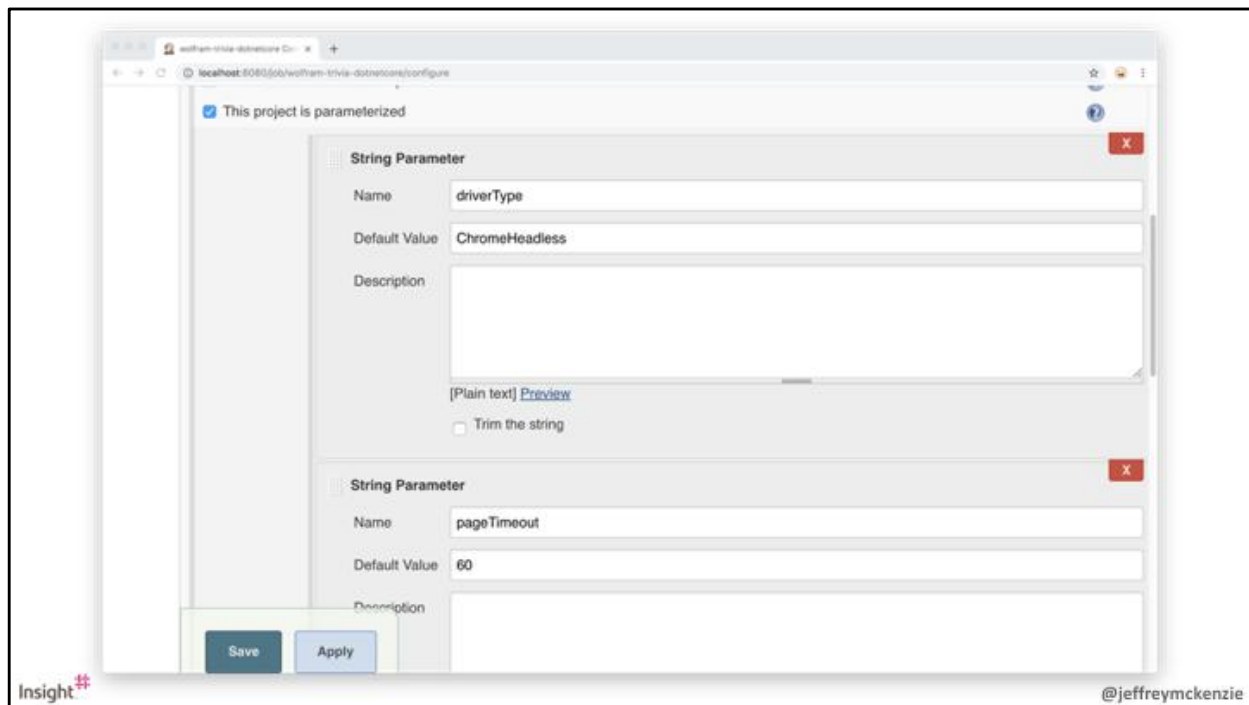
And in java we can use the AfterClass
JUnit attribute of the Test Runner class.



So let's look at running the dot net core project on Jenkins.

Since it's dotnet core it will work just fine on my Mac..

It is set up just about the same



The parameters are similar,

3 references | Jeff McKenzie, 9 days ago | 1 author, 2 changes

```
private string GetVariable(String key)
{
    string variable = Environment.GetEnvironmentVariable(key);
    if (variable == null)
    {
        return config[key];
    }
    return variable;
}
```

And just as in the Java code, in the .NET code we are first checking
To see if the environment variable is set,
If that doesn't exist then we default to the local setting.
So the main difference in the .NET Jenkins job
is in the code we are pulling
And the pipeline script itself...
So lets look at that..

```

pipeline {
  agent any
  stages {
    stage('Clean workspace') {
      steps {
        cleanws()
      }
    }
    stage('Checkout code') {
      steps {
        git credentialsId: 'mcknz-ssh', url: 'git@github.com:mcknz/wolfram-net.git'
      }
    }
    stage('Run tests') {
      steps {
        sh '/usr/local/share/dotnet/dotnet test WolframNet --logger trx
          --results-directory ${WORKSPACE}'
        mstest testResultsFile:"**/*.trx", keepLongStdio: true
      }
    }
  }
}

```

Insight 

@jeffreymckenzie

This one is even simpler.

```
pipeline {  
  agent any  
  stages {  
    stage('Clean workspace') {  
      steps {  
        cleanWs()  
      }  
    }  
    . . .  
  }  
}
```

Insight 

@jeffreymckenzie

Our first stage cleans out our Jenkins workspace
To make sure we don't have any old results in there
from previous builds

```
pipeline {
  agent any
  stages {
    stage('Clean workspace') { . . . }
    stage('Checkout code') {
      steps {
        git
        credentialsId: 'mcknz-ssh',
        url: 'git@github.com:mcknz/wolfram-net.git'
      }
    }
  }
}
```

Then we do our checkout, same as before,
except we are pulling our .NET project.

```

pipeline {
  agent any
  stages {
    stage('Clean workspace') { . . . }
    stage('Checkout code') { . . . }
    stage('Run tests') {
      steps {
        sh '/usr/local/share/dotnet/dotnet test wolframNet
          --logger trx --results-directory ${WORKSPACE}'
        mstest testResultsFile:"**/*.trx",
          keepLongStdio: true
      }
    }
  }
}

```

Insight 

@jeffreymckenzie

And in the run tests, we have 2 steps:
 Run the dotnet cli test command against the project,
 Publishing the mstest results to the workspace,
 And then run the mstest plugin to process the results
 and generate the report
 The plugin converts the mstest results to junit format

Jenkins

Jenkins » wolfram-trivia-dotnetcore » Full Stage View

wolfram-trivia-dotnetcore - Stage View

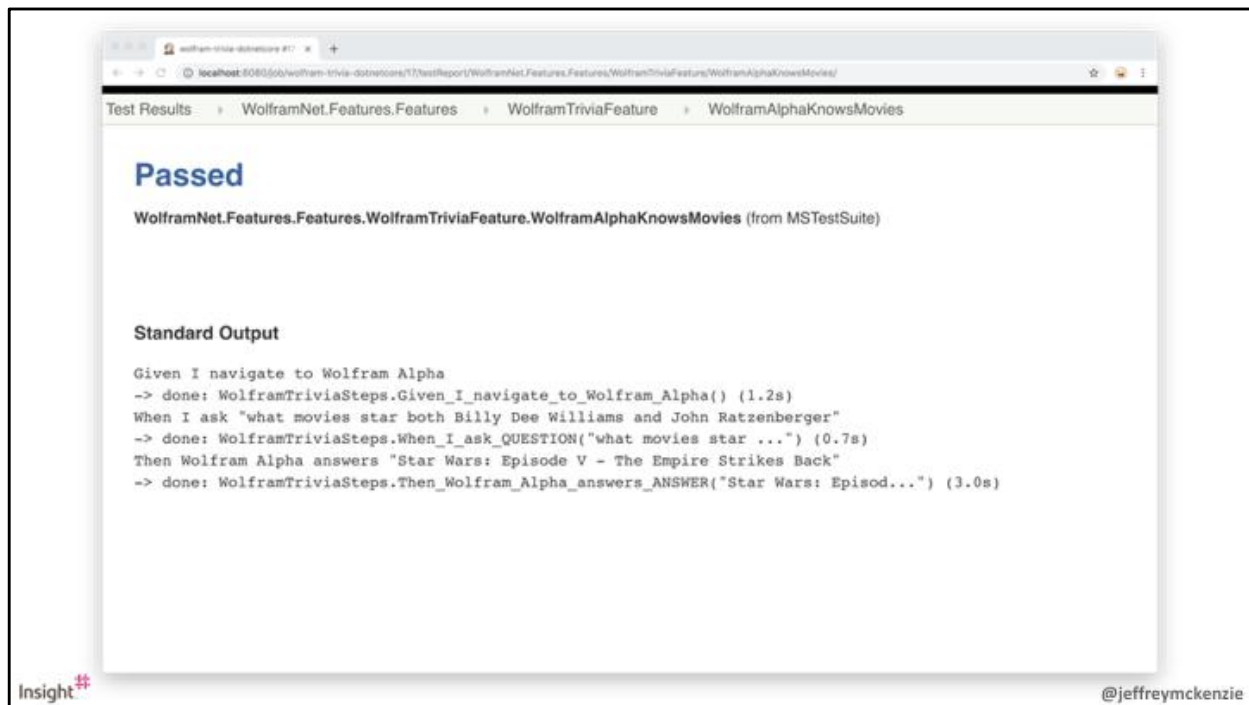
Average stage times:
(Average full run time: ~31s)

	Declarative: Checkout SCM	Clean workspace	Checkout code	Run tests
Average stage times: (Average full run time: ~31s)	1s	65ms	1s	25s
#17 Sep 26 15:14 1 commit	1s	65ms	1s	25s

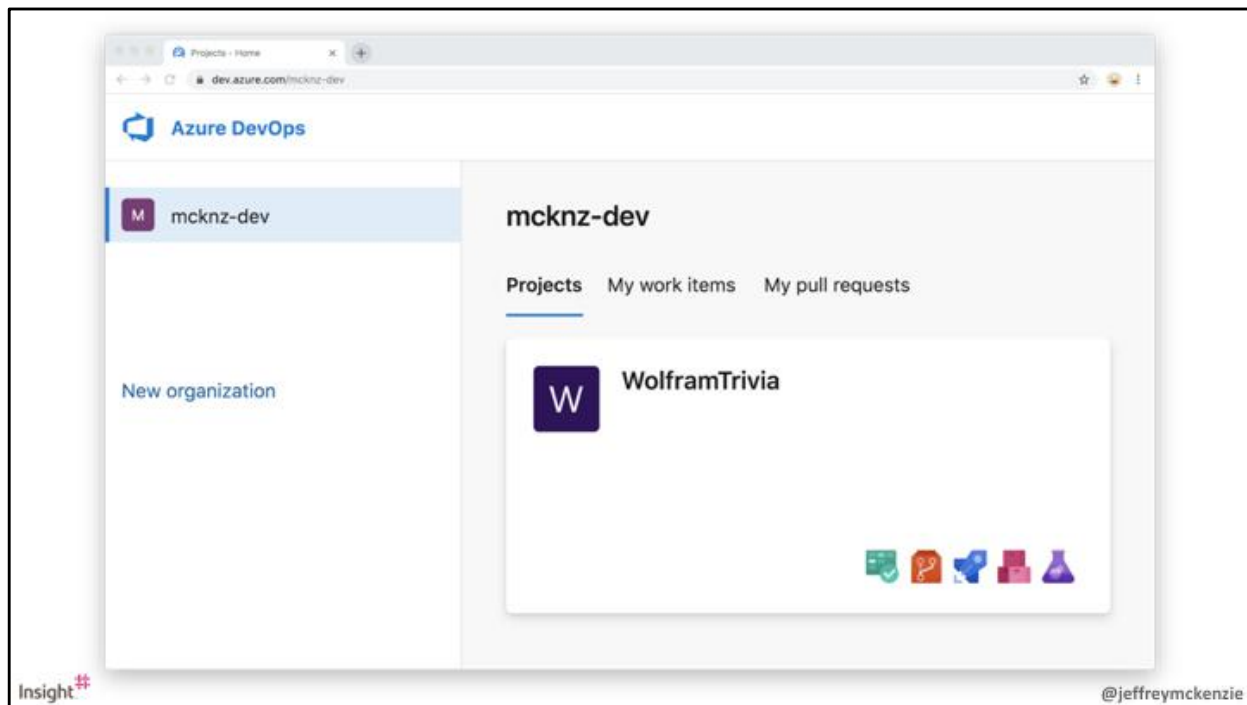
Insight

@jeffreymckenzie

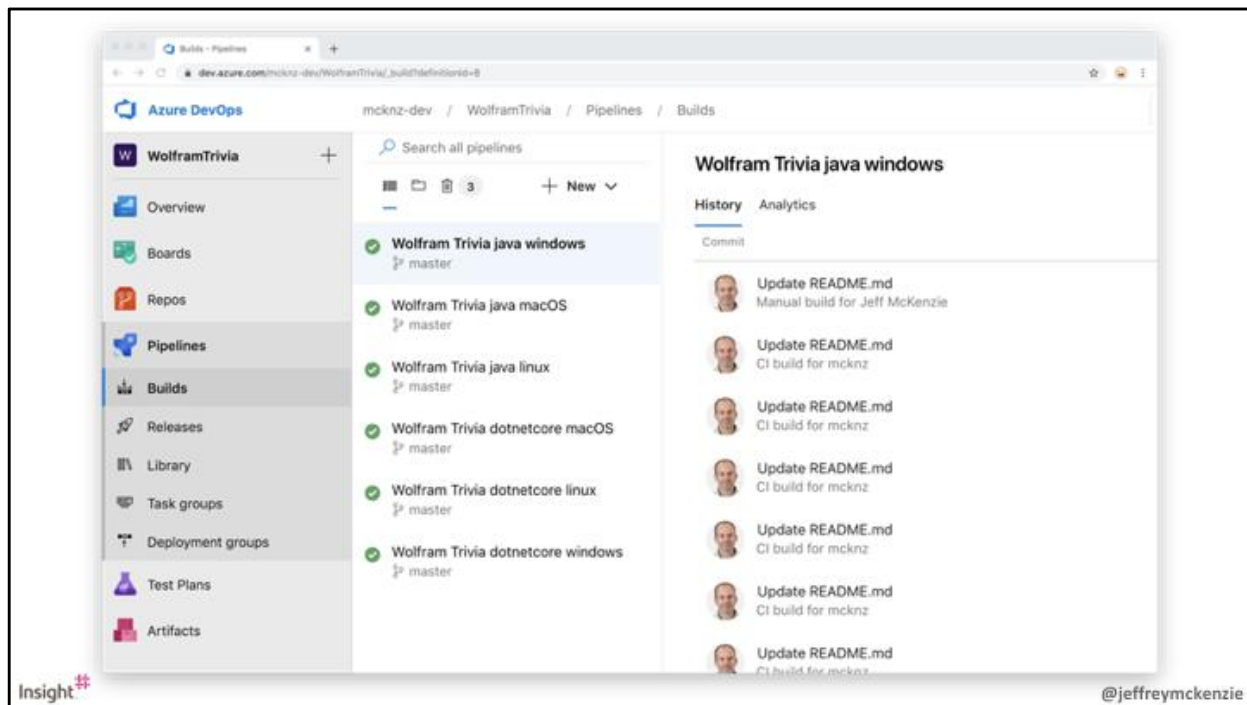
Again when we run this we see the stage view
With everything as success...



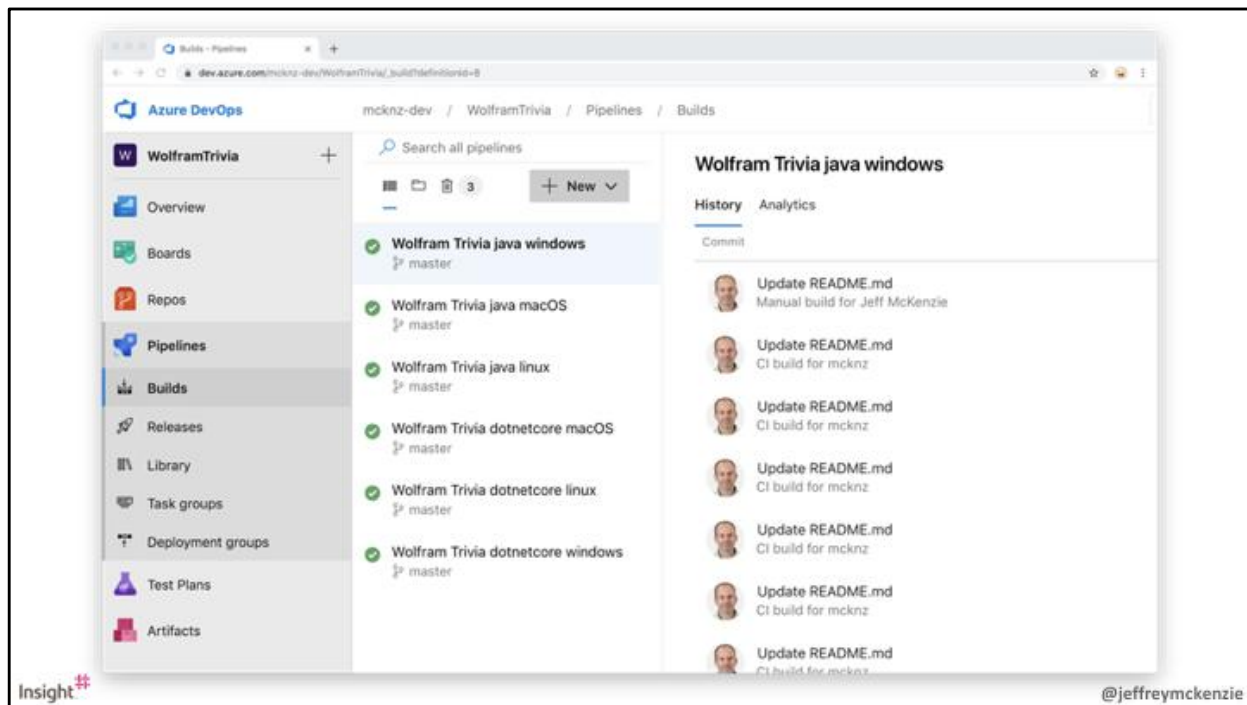
So the results aren't as nice as the java cucumber
But we still do get something we can inspect.



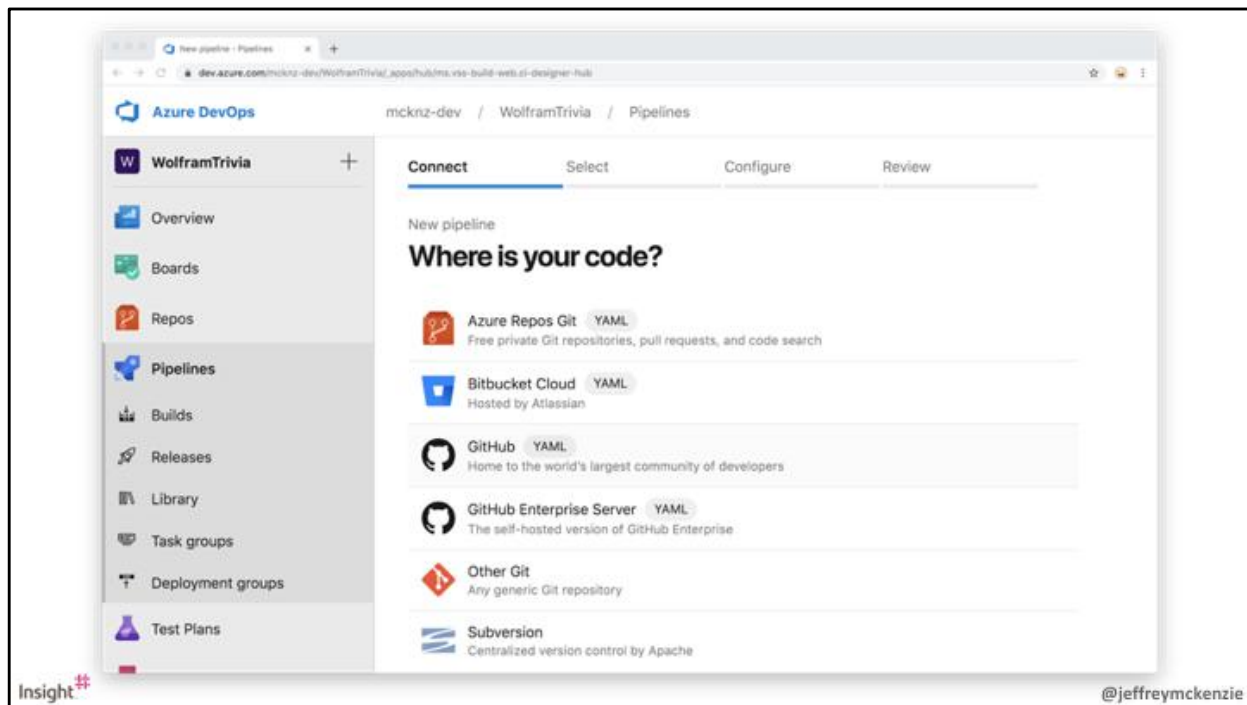
Let's switch over to Azure devops.
So I have a Wolfram Trivia project
in my mcknz-dev organization.
We are going to look at the pipelines



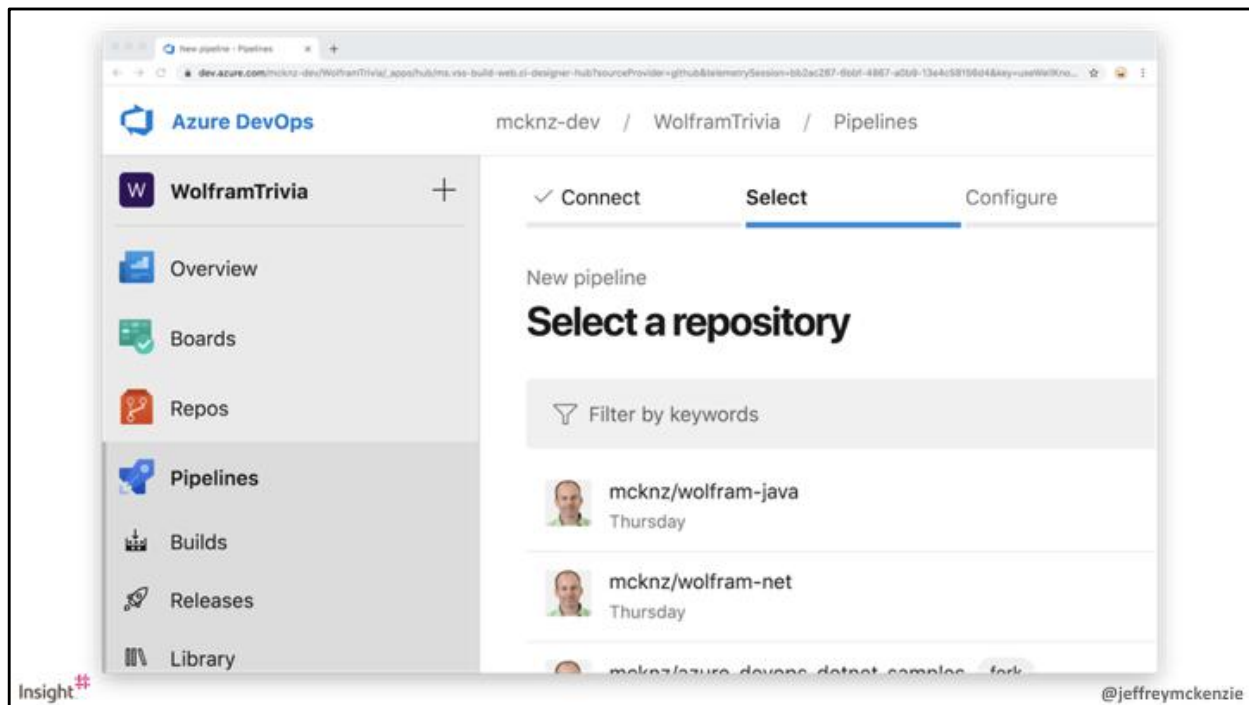
Now because I am just running tests
I can use the Builds pipeline, and specifically
The yaml pipelines, which are text based
Scripts very similar to the ones
We have been looking at in Jenkins
These are hosted agents, and you can see
I have 6 pipelines, one for java and one for dotnet
in Windows, Mac, and Linux



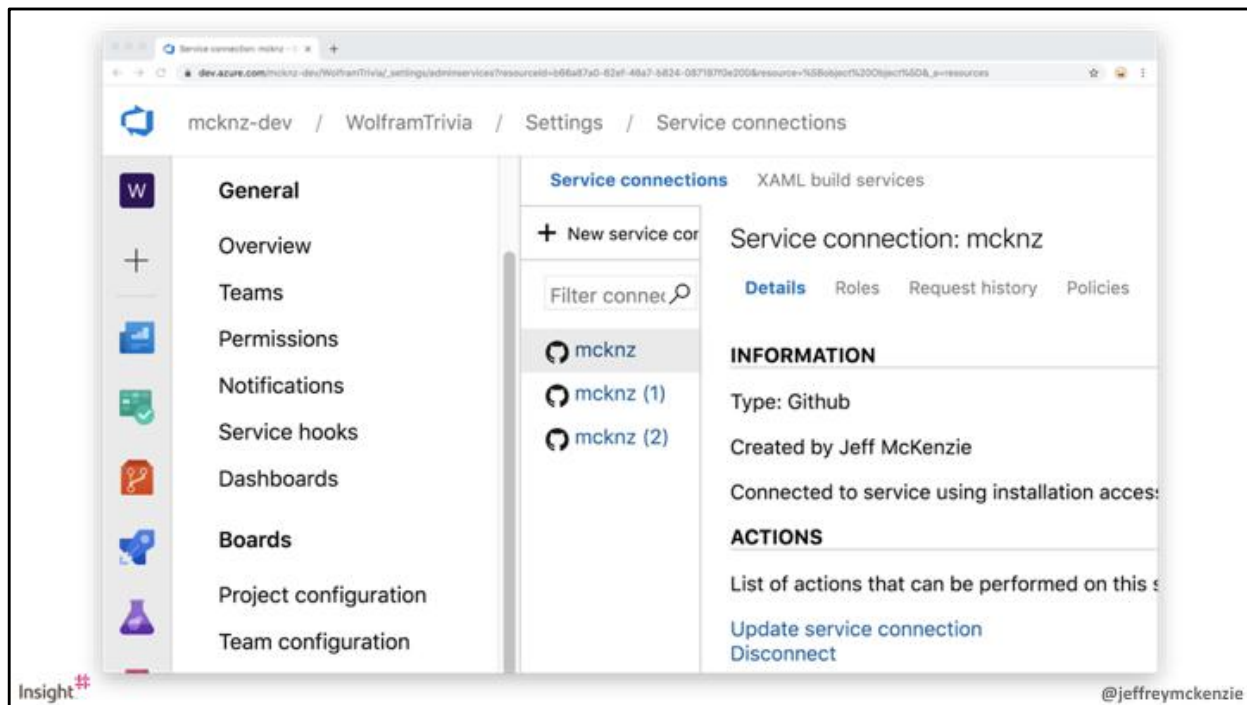
To create a pipeline, you simply click that new button above the pipeline list



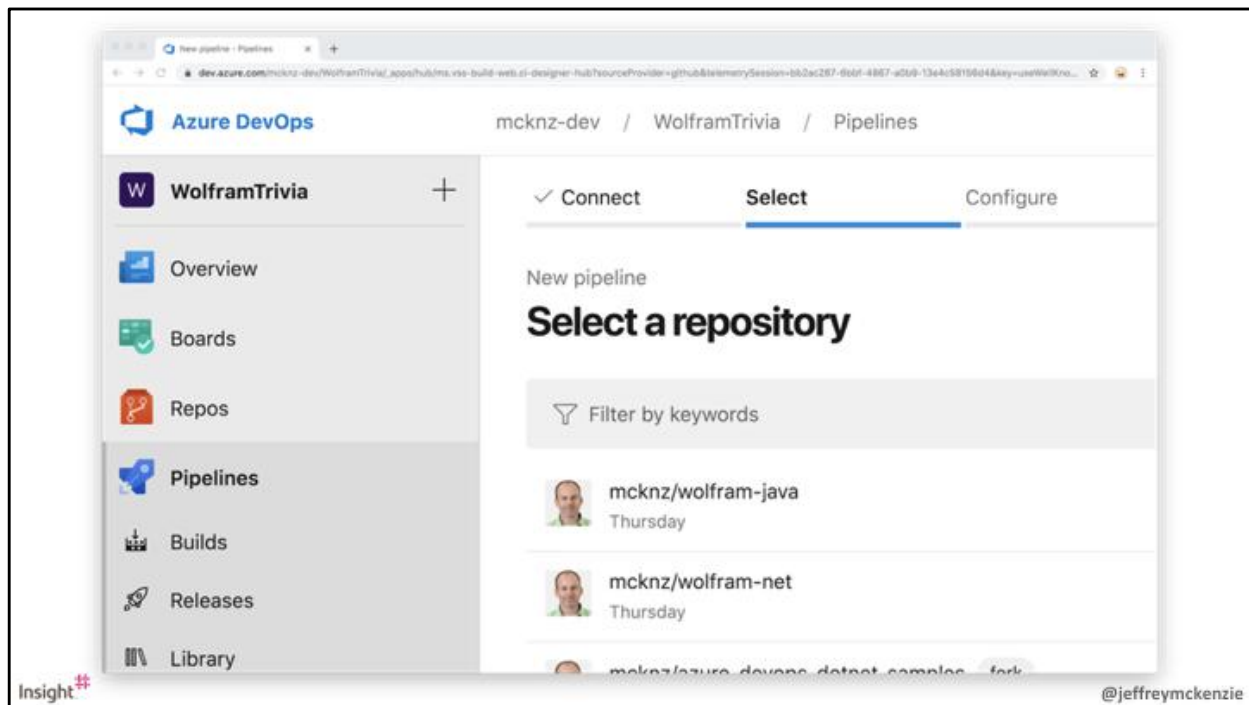
And then select where your code lives.
It might be in Azure devops itself –
Mine happens to be in Github so I will select that



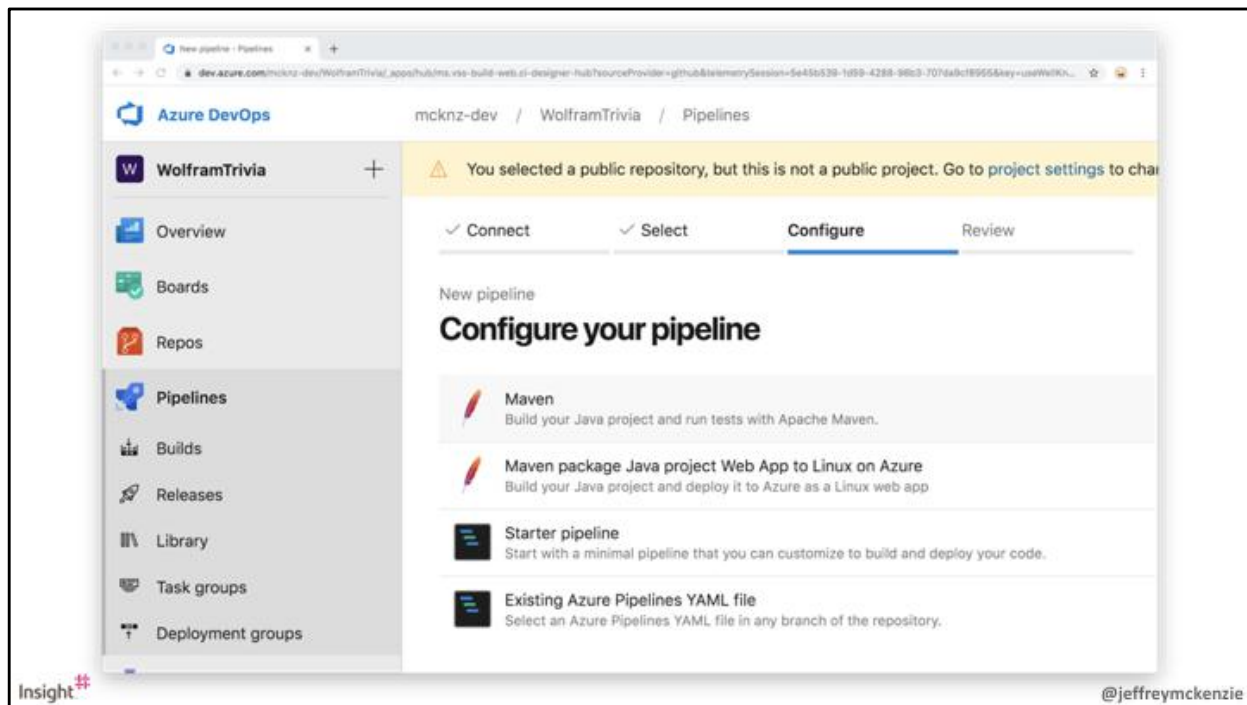
And then I can pick which repo I want to build.



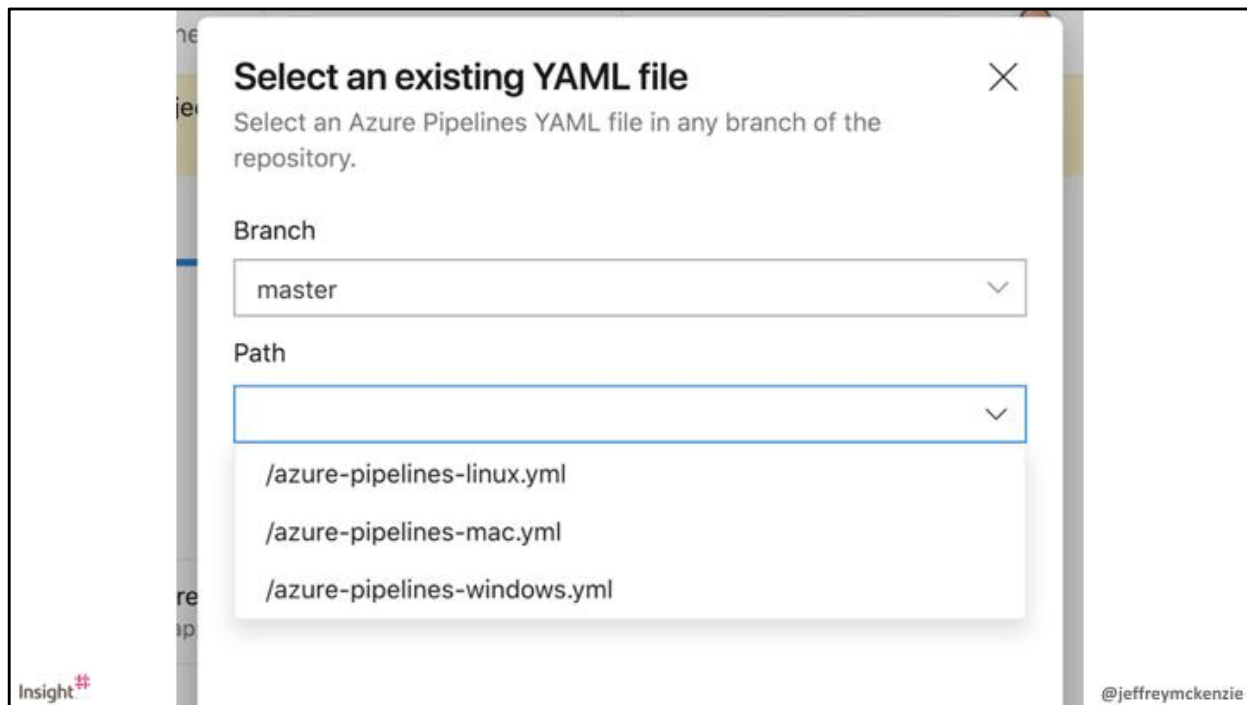
I have already created a service connection for GitHub –
If you are doing this for the first time you will need
To authenticate to GitHub using your credentials.



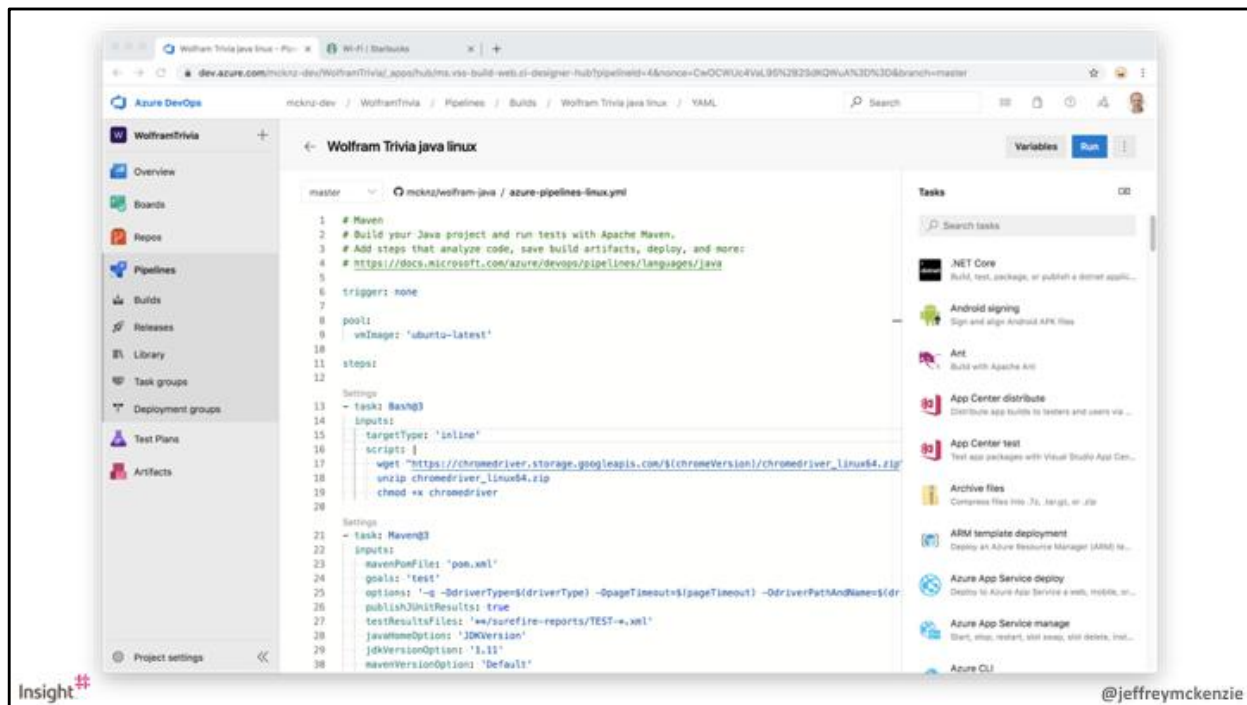
So I have my repos, and let's say I pick wolfram-java...



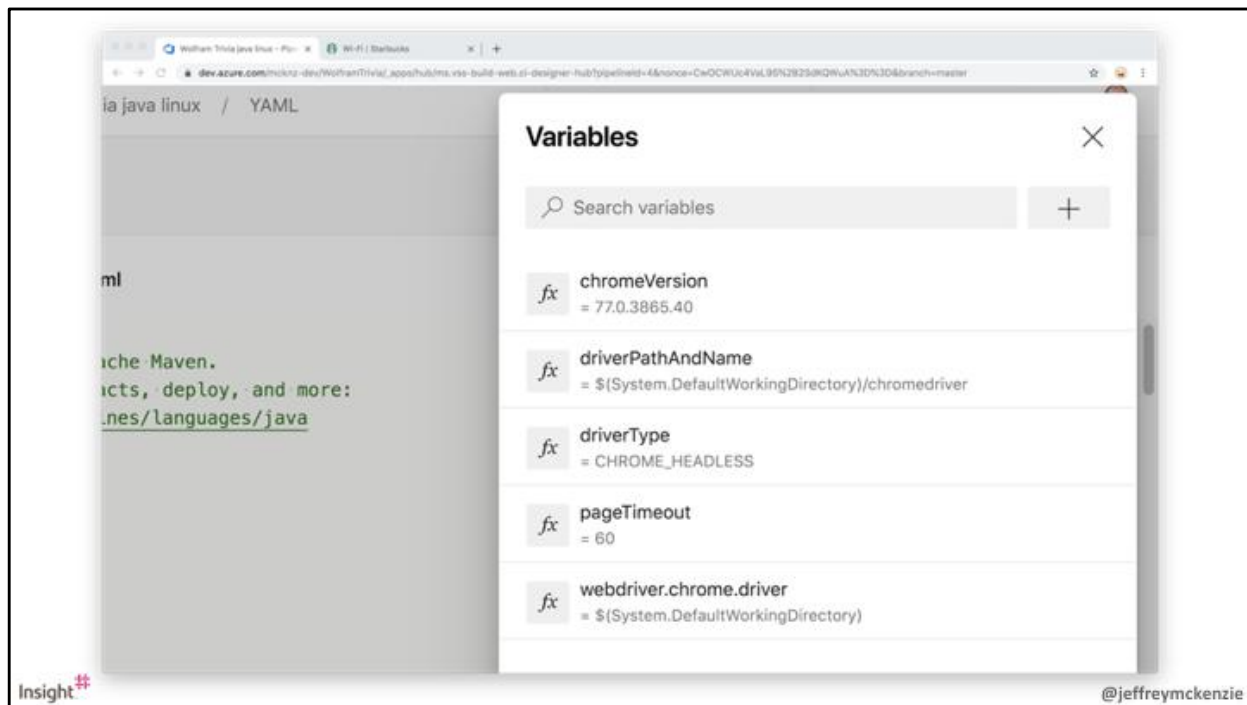
Azure DevOps recognizes that it's a Maven project
And gives me a couple of options to create a maven pipeline.
I could start from scratch, but I already have
An azure pipeline yaml file in there, 3 in fact,
So I could select one of those...



So my repo has yaml pipelines for linux, mac, and windows.
Let's take a look at those.



Here is the linux build for java,
What it looks like in the editor.
In the top right corner we have a variables button



So this is where we set the environment variables that
The project needs to run.
You can see we specify the driver Type as headless,
And the driverPath to the workspace..
We are also setting the chrome version
And that webdriver.chrome.driver system variable

```

# Maven
# Build your Java project and run tests with Apache Maven.
# Add steps that analyze code, save build artifacts, deploy, more:
# https://docs.microsoft.com/azure/devops/pipelines/languages/java

trigger: none

pool:
  vmImage: 'ubuntu-latest'

steps:
- task: Bash@3
  inputs:
    targetType: 'inline'
    script: |
      wget "https://chromedriver.storage.googleapis.com/
        $(chromeVersion)/chromedriver_linux64.zip"
      unzip chromedriver_linux64.zip
      chmod +x chromedriver

```

Insight 

@jeffreymckenzie

Let's look at the script.

First I could specify a trigger for this build,

Again I am running these manually so I say "trigger none"

Pool/VMImage allows me to select which agent

I want to run this on.. I select the latest version of ubuntu.

Then I specify my steps,

And my first step is a bash task...

The latest image does not contain

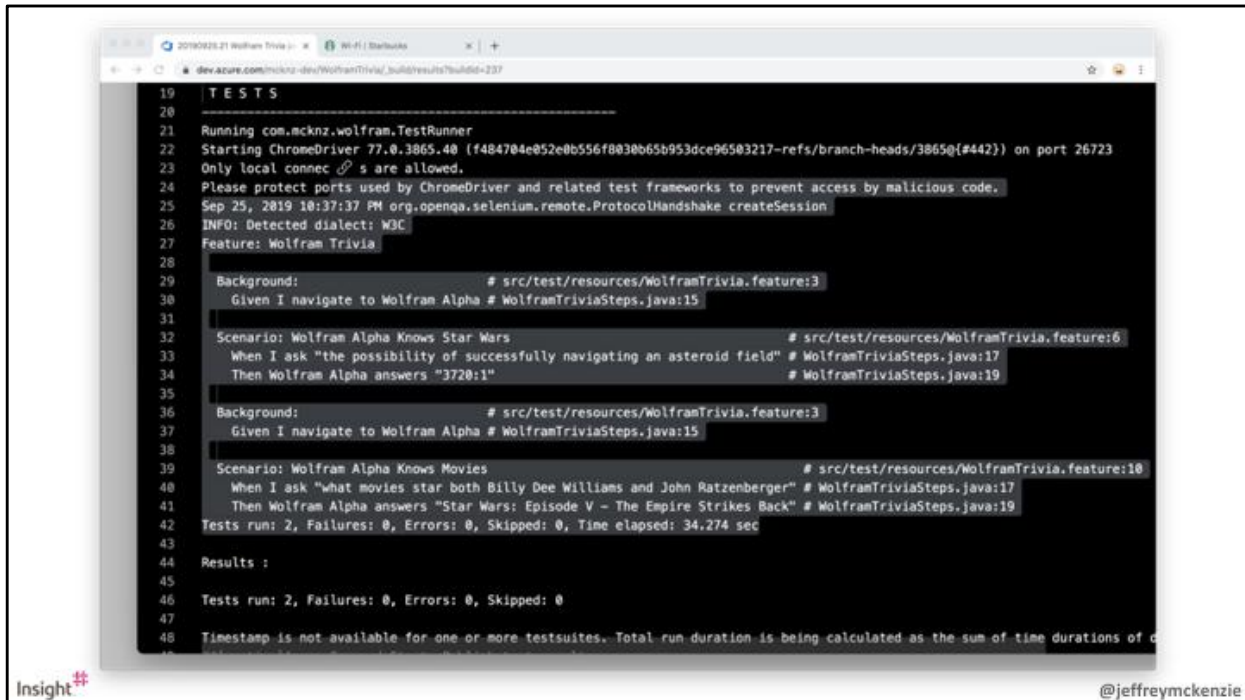
The chrome driver, so I need to download the zip file

And expand it in the workspace so my project

Has access to it.


```
- task: Maven@3
  inputs:
    mavenPomFile: 'pom.xml'
    goals: 'test'
    options: '-q
      -DdriverType=$(driverType)
      -DpageTimeout=$(pageTimeout)
      -DdriverPathAndName=$(driverPathAndName)'
    publishJUnitResults: true
    testResultsFiles: '**/surefire-reports/TEST-*.xml'
    javaHomeOption: 'JDKVersion'
    jdkVersionOption: '1.11'
    mavenVersionOption: 'Default'
```

Then there's a built-in maven task that again
Just runs the test goal, and passes in those parameters.

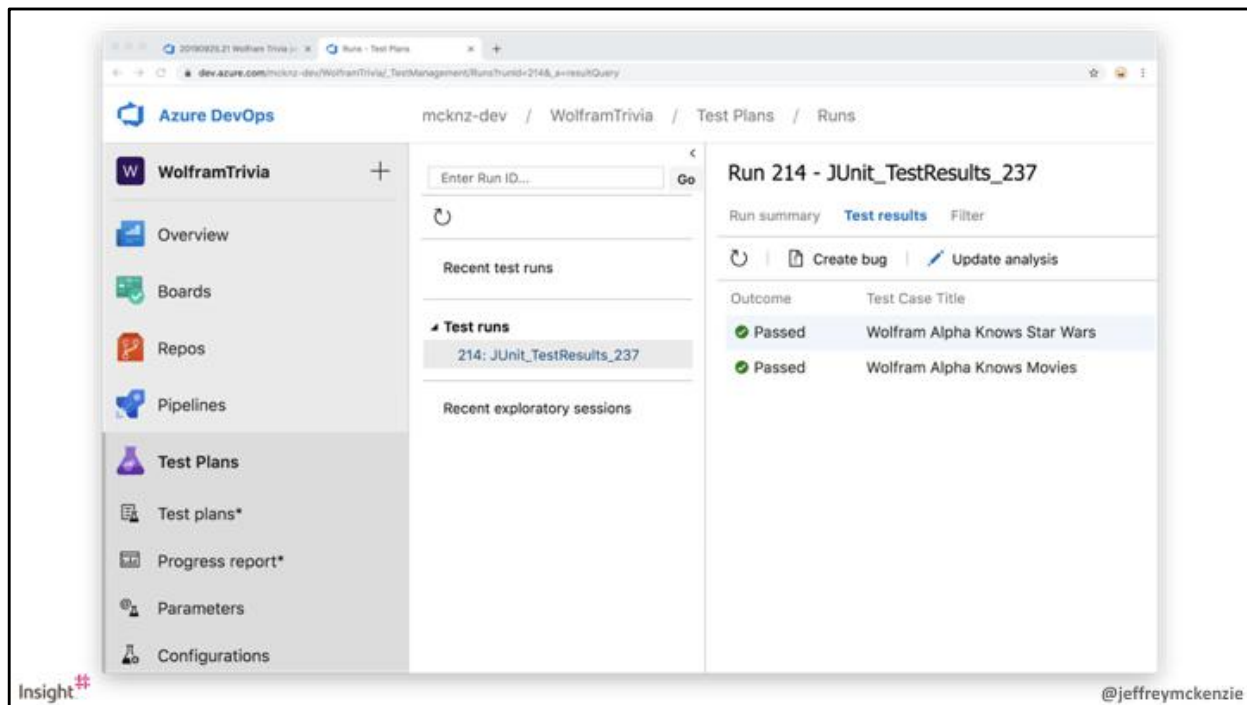


The screenshot shows a web browser window with the address bar displaying a URL from dev.azure.com. The main content area shows the output of a Selenium test run. The output is formatted with line numbers on the left and test details on the right. It includes a 'TESTS' section, a 'Running' message, a 'Starting ChromeDriver' message, a warning about local connections, a 'Please protect ports' warning, a 'createSession' message, an 'INFO' message about the detected dialect, a 'Feature' message, a 'Background' message, a 'Given' message, a 'Scenario' message, a 'When' message, a 'Then' message, a 'Tests run' summary, a 'Results' section, and a 'Timestamp' message.

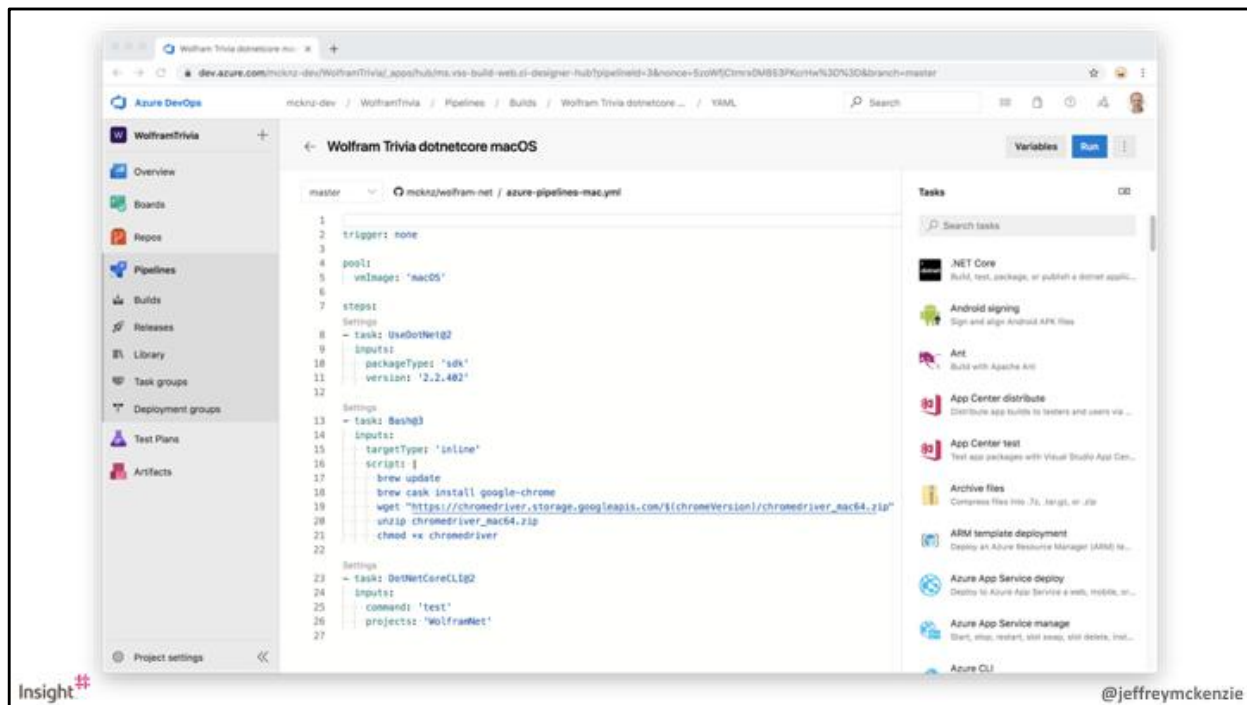
```
19 TESTS
20 -----
21 Running com.mcknz.wolfram.TestRunner
22 Starting ChromeDriver 77.0.3865.40 (f484704e052e0b556f8030b65b953dce96503217-refs/branch-heads/3865@{#442}) on port 26723
23 Only local connections are allowed.
24 Please protect ports used by ChromeDriver and related test frameworks to prevent access by malicious code.
25 Sep 25, 2019 10:37:37 PM org.openqa.selenium.remote.ProtocolHandshake createSession
26 INFO: Detected dialect: W3C
27 Feature: Wolfram Trivia
28
29 Background: # src/test/resources/WolframTrivia.feature:3
30 Given I navigate to Wolfram Alpha # WolframTriviaSteps.java:15
31
32 Scenario: Wolfram Alpha Knows Star Wars # src/test/resources/WolframTrivia.feature:6
33 When I ask "the possibility of successfully navigating an asteroid field" # WolframTriviaSteps.java:17
34 Then Wolfram Alpha answers "3720:1" # WolframTriviaSteps.java:19
35
36 Background: # src/test/resources/WolframTrivia.feature:3
37 Given I navigate to Wolfram Alpha # WolframTriviaSteps.java:15
38
39 Scenario: Wolfram Alpha Knows Movies # src/test/resources/WolframTrivia.feature:10
40 When I ask "what movies star both Billy Dee Williams and John Ratzenberger" # WolframTriviaSteps.java:17
41 Then Wolfram Alpha answers "Star Wars: Episode V - The Empire Strikes Back" # WolframTriviaSteps.java:19
42 Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 34.274 sec
43
44 Results :
45
46 Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
47
48 Timestamp is not available for one or more testsuites. Total run duration is being calculated as the sum of time durations of d
```

Insight  @jeffreymckenzie

When I run this you can see the gherkin executed
In the console..



And you also get a Junit results test run summary...



Not going to look at all of these
But want to hit on the highlights to show
The different things you can and need to do
In a hosted environment.
Here is dotnetcore on MacOS


```
pool:
  vmImage: 'macOS'

steps:
- task: UseDotNet@2
  inputs:
    packageType: 'sdk'
    version: '2.2.402'

- task: Bash@3
  inputs:
    targetType: 'inline'
    script: |
      brew update
      brew cask install google-chrome
```

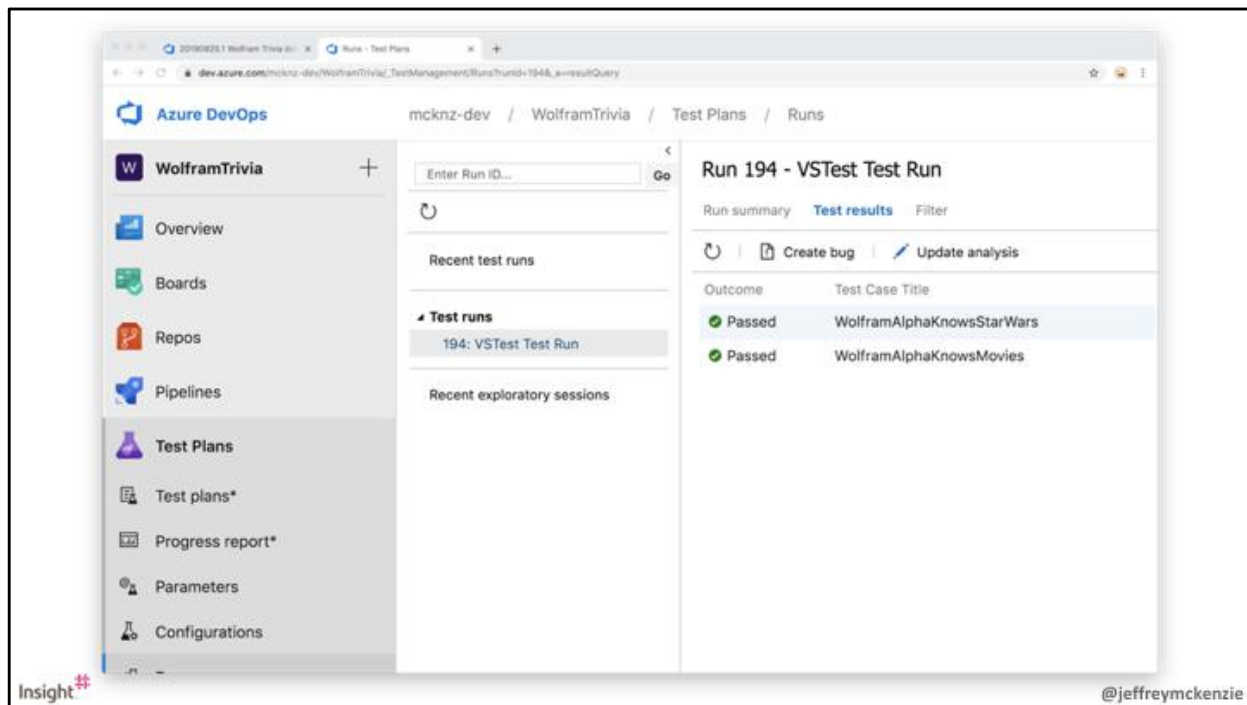
Again I specify the image,
I pull the version of .NET core I want to use (2.2.402)
The macOS image doesn't come with Chrome
Or Chrome Driver, so I have to run brew cask install chrome...

```
- task: Bash@3
  inputs:
    targetType: 'inline'
    script: |
      brew update
      brew cask install google-chrome

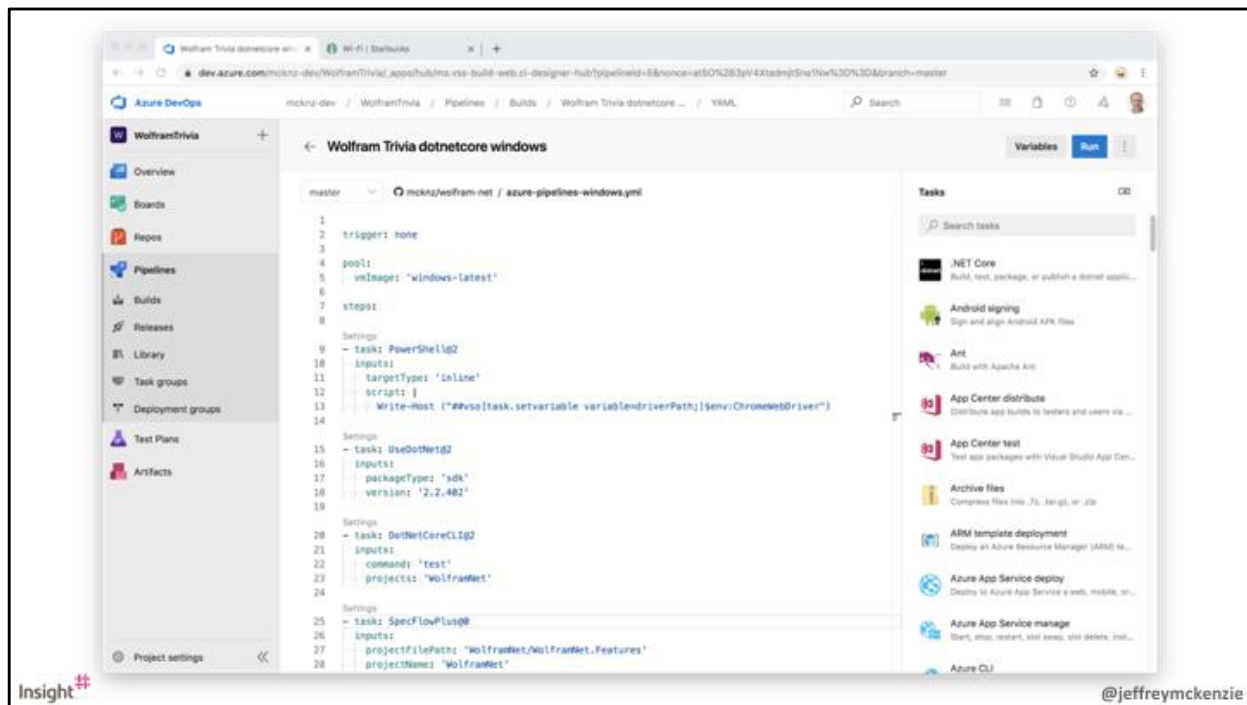
      wget "https://chromedriver.storage.googleapis.com/
        $(chromeVersion)/chromedriver_mac64.zip"
      unzip chromedriver_mac64.zip
      chmod +x chromedriver

- task: DotNetCoreCLI@2
  inputs:
    command: 'test'
    projects: 'WolframNet'
```

Then download the driver, unzip, and set permissions...
Then I can run the cli test command.



I do get test results for that as well



Last one we will look at is the dotnet core for windows,
It's a little interesting

```
trigger: none

pool:
  vmImage: 'windows-latest'

steps:
- task: PowerShell@2
  inputs:
    targetType: 'inline'
    script: |
      write-Host ("##vso[task.setvariable
        variable=driverPath;]$env:ChromeWebDriver")
```

Insight 

@jeffreymckenzie

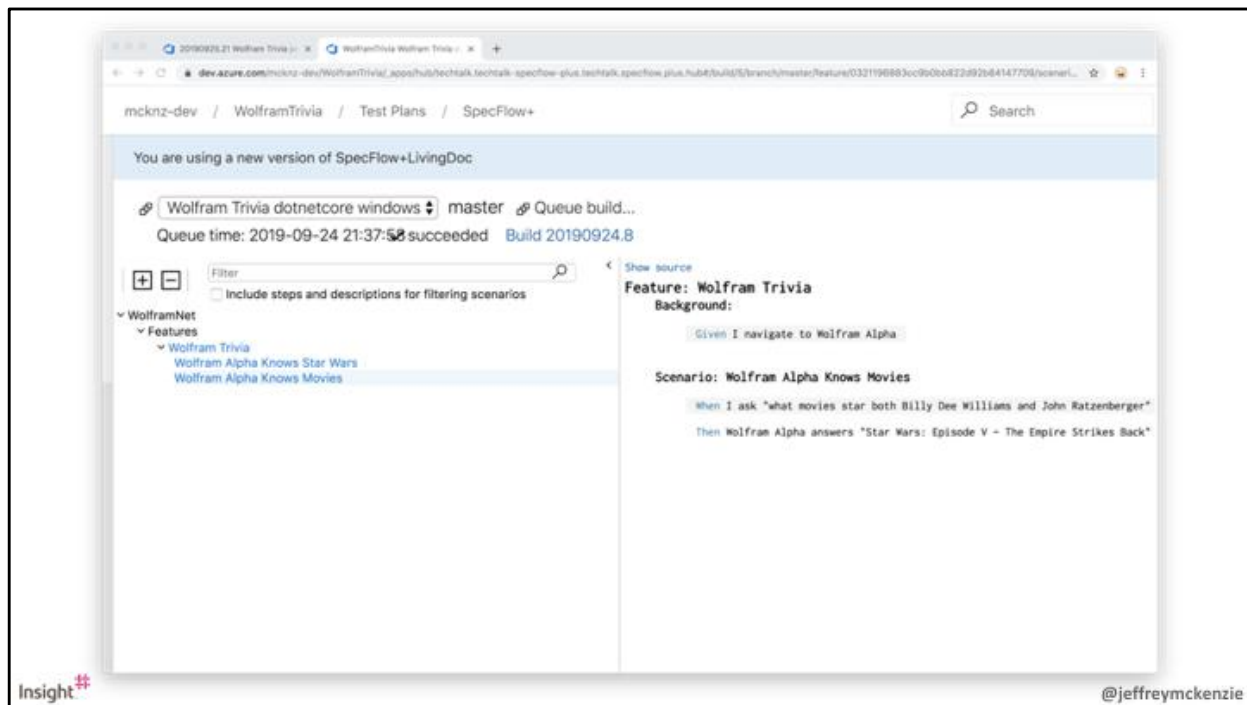
The latest version of Windows does come with ChromeDriver,
But the path to ChromeDriver is in an environment variable
On the agent.
But I need to pass that to my project.
One trick for that is to use PowerShell WriteHost—
Essentially here we are getting the ChromeWebDriver value
From the host, and writing to our driverPath pipeline variable.

```
- task: UseDotNet@2
  inputs:
    packageType: 'sdk'
    version: '2.2.402'

- task: DotNetCoreCLI@2
  inputs:
    command: 'test'
    projects: 'WolframNet'

- task: SpecFlowPlus@0
  inputs:
    projectFilePath: 'WolframNet/WolframNet.Features'
    projectName: 'WolframNet'
    projectLanguage: 'en'
```

Then again we specify our version of dotnetcore,
And Run the cli test command.
SpecFlow actually has an Azure DevOps Pipeline task
That runs on windows agents we can use



So there it is....

It's not great, but it's there.

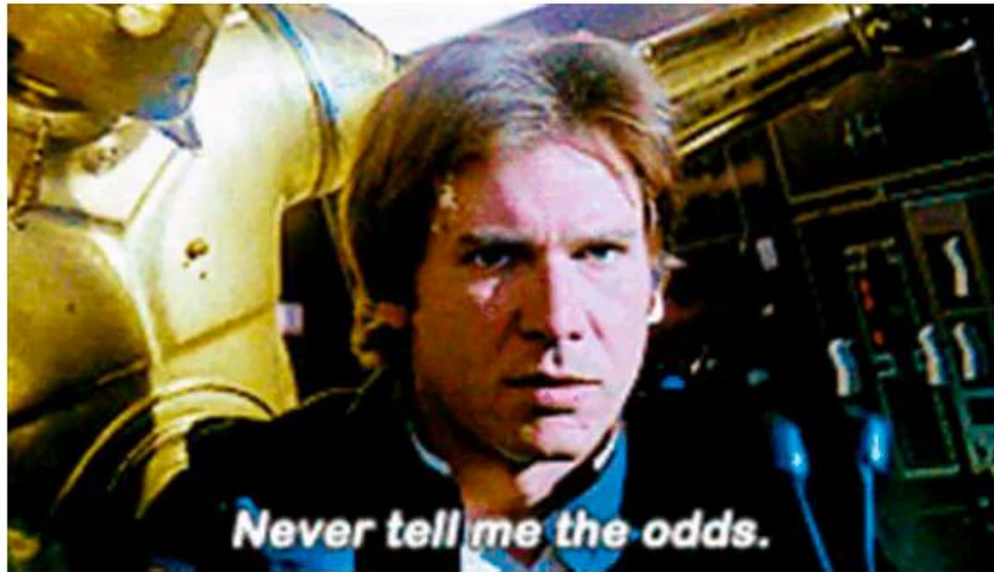
It's also not free -- there is a limit of like 10 tests
before you need to pay for a license

@jeffreymckenzie

Ci(ui)

Insight.

So that's what I got –
There's a lot here, and it can be tricky to get it all
To play together, but...



Insight

@jeffreymckenzie

Don't ever let anyone tell you the odds.

<https://github.com/mcknz/DogFoodCon-2019>

Get Some UI In Your CI Trivia Edition

Jeff McKenzie • jeff.mckenzie@insight.com • [@jeffreymckenzie](https://twitter.com/jeffreymckenzie)

Good morning/afternoon everyone –
Thanks for being here...

Today – UI testing, specifically within
A continuous Integration environment.
This is the special Trivia Edition,
Which we will get into a little later.

My name is Jeff McKenzie,
And I am a Practice Manager for App Dev and Infrastructure
At Insight Digital Innovation in Columbus Ohio