04/28/2018  
----------

Abdullah:

States  
------  
Alaska  
Alamabama  
......

TX  
CA  
..  
==============

(SELECT STATE FROM STATES  
WHERE STATE IN ('TX','CA')  
ORDER BY STATE DESC)  
UNION ALL  
(SELECT STATE FROM STATES  
WHERE NOT STATE IN ('TX','CA'));  
===============

MAY 2 - 2018  
last official class.

MAY 5 - MAY 12 -> REVIEW SESSION DAILY(BOOTCAMP)  
11AM - 5PM EST DAILY  
===============

API DAY 6:  
---------

Cucumber scenario with API and UI  
AND DB  
---------------------------------

AUTHENTICATION METHODS IN REST API:  
-> GOOGLE MAPS API  
==================================

END TO END TESTING SCENARIOS:

END TO END TESTING -> INVOLVING FUNCTIONALITY  
END TO END TESTING -> INVOLVING FUNCTIONALITY PLUS  
EACH LAYER OF APPLICATION

UI, API, DB

1) GOTO UI -> ADD AN EMPLOYEE  
EXPECTED:  
1) GOTO DATABASE AND VERIFY IF EMPLOYEE IS ADDED SUCCESSFULLY AND ALL DATA IS MATCHING  
2) API -> GET REQUEST AND VERIFY IF EMPLOYEE IS ADDED SUCCESSFULLY AND ALL DATA IS MATCHING

-> Makes changes in Front end and verify in database and Rest API.

2) GOTO UI -> ADD AN EMPLOYEE:  
CHECK IN UI SEARCH PAGE.   
----------------

3) POST AN EMPLOYEE USING REST API:  
EXPECTED:  
1) SEND A GET REQUEST WITH API AND VERIFY  
2) GOTO DATABASE AND VERIFY IF EMPLOYEE IS ADDED SUCCESSFULLY AND ALL DATA IS MATCHING  
3) GOTO TO FRONT END(WEBSITE) AND VERIFY THAT DATA POSTED IS DISPLAYED

-> Makes changes using REST API then verify in Database and UI  
--------------------------

4) INSERT AN EMPLOYEE INTO DATABASE:  
EXPECTED:  
1) RUN SELECT STATEMENT IN DB AND VERIFY WHATEVER YOU INSERTED IS THERE IN TABLES  
2) SEND API GET REQUEST AND VERIFY JSON IS MATCHING DATA YOU INSERTED TO DATABASE  
3) GOTO TO FRONT END(WEBSITE) AND VERIFY THAT DATA INSERTED TO DB IS DISPLAYED

-> Make changes in Database using sql and verify in REST API and FRONT END  
=========================

TELL ME ABOUT YOUR PROJECT?  
-> web application  
-> benefit/purpose of application  
-> some useful functionality  
-> some name of the application

WHERE IN YOUR PROJECT YOU HAVE REST API?  
WHY DEVELOPED A REST API?  
-> Application in the current project needs to be integrated to other internal and external applications. For the integration our team developed RESTful API.

PROJECT/APP A ----> PROJECT/APP B IN SAME COMPANY  
INTEGRATION WITH REST API

APP A ---> EXTERNAL APPLICATION  
INTEGRATION USING REST API  
================================

REST API TESTING:  
How do you test rest api?

I verify if each REST API endpoint is working as expected.  
I send POST,PUT,GET, DETELE type of requests and verify reponse status code and response body, header.  
I also do positive and negative testing of API.  
When I do positive testing, I send valid request parameters , valid headers, valid request json body and verify that response status code is 200 successful and Json response body data is also matching the expected.

When I do negative testing, I send invalid request parameters , or invalid headers, or invalid request json body and verify that response status code is not 200 and Json response body contains error message.  
========================

DOCUMENTATION OF API: SWAGGER UI  
MANUAL TESTING OF API: POSTMAN,SOAP UI  
AUTOMATED TESTING OF API: REST-ASSURED LIBRARY IN JAVA.   
HOW API WAS DEVELOPED? ORACLE ORDS FRAMEWORK  
.IT IS A JAVA APPLICATION AND WAS DEVELOPED USING SPRING FRAMEWORK  
========================

When do you get StaleElementException?  
After initially finding the element, page or element was refreshed then Selenium cannot work with that element any more.  
------------------------

CHALLENGES WHILE DOING END-TO-END AUTOMATION  
THAT INCLUDES UI, API, DATABASE LAYERS.

1) Data validation:  
UI data is in HTML format  
API data is in JSON format  
Database data is in ResultSet object format

Solution:  
1) Bring all data from 3 sources into same java data structure format. We used mostly maps because it is easy to work with K,Vs.  
THEN COMPARE/VALIDATE/ASSERT

HTML -> SELENIUM > MAP  
JSON -> GSON -> MAP  
RESULTSET -> MAP

2) Create POJOs/ Custom classes to match our data and build objects using data in different sources.

HTML -> SELENIUM > POJO/BEAN  
JSON -> GSON -> POJO/BEAN  
RESULTSET -> POJO/BEAN

3) Direct comparison - will end up with diffucult to maintain and read code - spagetti code.  
==============================

AUTHENTICATION IN REST API  
--------------------------

1) Using username and password  
You pass them as query parameters.

2) AUTH Token

AIzaSyABLuS48NJ5zFhCkP\_4eRyONGejoRQA51Q

Given Accept type is JSON  
And query parameters are following:  
origin : 7925 Jones Branch Drive, McLean VA 22102  
destination : 75 9th Ave, New York, NY  
mode : driving  
key : AIzaSyABLuS48NJ5zFhCkP\_4eRyONGejoRQA51Q  
When I perform GET request to Google maps API url:  
https://maps.googleapis.com/maps/api/directions/json  
Then status code should be 200  
And distance and duration should be in response JSON  
-------------------------

Summary RestFul API:  
-> Manual test using Postman  
-> Http request types: POST , PUT, GET, DELETE  
-> Http status codes  
-> Headers, Parameters   
-> JSON , validate data in json  
-> Response body > de-serialize it  
-> serializing, de-serializing  
-> POJOs  
-> RESTAssured library in Java  
-> END TO END testing automation  
FULL STACK AUTOMATION ENGINEER