**Name: Daniel Shako Kodiemoka**

**Company: ENTERSEKT**

**Agent: Bonita Young**

QE Analyst Practical

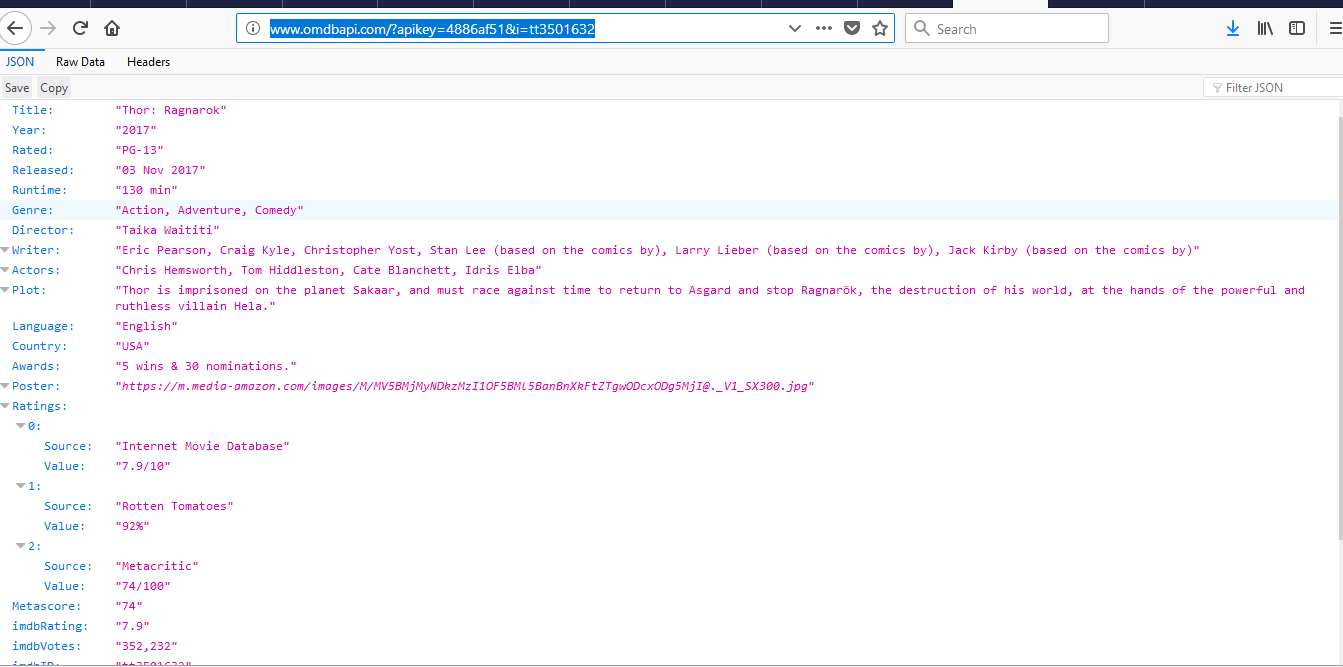
Assessment

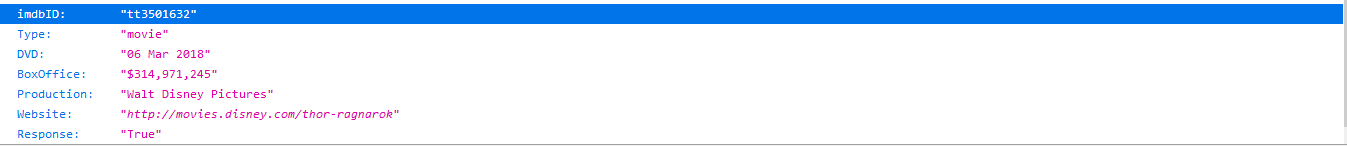
**Practical Exercise**

1. API Key: 4886af51 provided
2. <http://www.omdbapi.com/#usage> API Usage
3. Query the API for the film “Thor: Ragnarok” using only the api key and the “i” parameter in your query string

<http://www.omdbapi.com/?apikey=4886af51&i=tt3501632>

Results:





1. Write Test Case for Step 3:

Test Description: Thor: Ragnarok

|  |  |  |
| --- | --- | --- |
| Steps | Action | Expect |
| 1 | Access the API Usage link: http://www.omdbapi.com/#usage | The API Link is displayed on the web |
| 2 | In the API Usage link [http://www.omdbapi.com/?apikey=[yourkey]&](http://www.omdbapi.com/?apikey=%5byourkey%5d&)  enter the following details:   * API Key: 4886af51 & * “I” parameter “should be the movie IMDb ID” | The API Key and “I” parameter are entered in the link |
| 3 | Run the API Link on web browser address bar:  <http://www.omdbapi.com/?apikey=4886af51&i=tt3501632> | The Link is ran on the address bar in the web browser |
| 4 | Thor: Ragnarok movie JSON description or results should be displayed | The JSON contains the movie description |

1. automated test script for step 3.

“test\_steps” : [{

“description”: “Step 1 – Access the API Usage link”,

“expected”: “login page opens, obviously”,

},

{

“description”: “Step 2 – <http://www.omdbapi.com/?apikey=4886af51&i=tt3501632>”,

“expected”: “user is redirected to the queried page”,

},

{

“description”: “Step 3 – Thor: Ragnarok results “,

“expected”: “The page displays Thor: Ragnarok description”,

“attachments”: []

}]

**Theoretical Exercise**

1. Difference between a Test Case and a User Story?

Test Case - is a set of conditions or variables or steps which determines whether the software satisfies requirements and functions properly.

User Story - A user story is a tool used in Agile software development get a description of a software feature from an end-user perspective. The user story describes the type of user, what they want and why.

1. Considering a Continuous Integration environment, when should execution of automated tests occur?

* I would say depending on scenarios in a continuous integration environment, 100% functionality correct scripts would be needed for an end-to-end testing. At this point, automation coverage matters a lot as there is no time allocated in the middle of the integration for scripts to be fixed. In logic we have to COMMIT the changes to a code repository, BUILD the software test coverage then EXECUTE the automated tests.

1. What type of testing is at an API level?

* Functional Testing – Testing the end-to-end functionality of broader scenarios
* Unit Testing – Testing the units of an application.
* Integration Testing – Test integration between different components and software.
* Performance Testing - This level of testing is performed after we know that all the functionality is properly working but we also want to understand how the API would behave under high load.
* Runtime error detection - Monitoring an application the execution of automated or manual tests to expose problems such as race conditions, exceptions, and resource leaks.

1. When is the most effective time to involve a software quality engineer in the SDLC?

* In the Software Development Life Cycle (SDLC) the software quality engineer should be involved in the CODING and IMPLEMENTATION or BUILD phase. This phase includes the actual engineering and writing of the application — while attempting to meet all the requirements established during the planning phase.

1. For the practical exercise, what percentage of automation vs manual testing would be ideal?

Manual Testing percentage – 40% due to human/testers logical skills, analytical as well as usability in terms of manual approach to measure how testing is progressing.

Automation Testing – 60% due to repeated execution of this task, load testing for completing the testing efficiently, regression testing for frequent changes and performance testing for simulation of concurrent scripts or users.