Class 11

1.

- Enter https://dgotlieb.github.io/Navigation/Navigation.html
- Print iFrame text.

2.

Create a TestNG test with the following:

- Enter https://dgotlieb.github.io/Actions
- Take a screenshot of the box element
- Drag wheel into box.
- Double click the text “Double-click…” check what happened and create assertion on result.
- Make a mouse hover on X image.
- Select two items from food list.
- Upload a file.
- Scroll down to “click me button” try both scroll to element and scroll to location.

3.

- Create XML file with a key name URL and a value with any website URL (e.g. https://www.....com).
- Build a TestNG test with the following:
  - Read the URL value from the XML file.
  - Open the browser with the given URL.
  - Change the URL in the XML file (manually) to test the effect

4. Create a TestNG test with the following:
   - Setup Extent Reports.
   - Log every step in your program into the report (@BeforeClass, @Test, etc.).
   - Go to Google Translate.
   - Take a screen shot and it to report.
   - Press on translation field (the one you enter the text to be translated).
   - Add your company name to system info.

5. Write all test logs (all System.out.print…) into a .log file

6.

- Enter https://dgotlieb.github.io/Navigation/Navigation.html
- Press on “Show alert” button and print alert text.
- Press on “Show prompt” button, fill your name and assert result.
- Press on “Show confirm box” button, press on one button and assert result accordingly (assert the text shown after pressing).
- Press “open new tab” button and go back to main tab.
- Press “open new window” button and go back to main window

CHALLENGES:

7.

- Open Google in first tab
- Open YouTube on the second tab
- Open Google translate in the third tab.
- From translate go to Google and from Google go to YouTube.

8. Create a JSON file with a website URL (any).
    - Create a test which will read the URL from the JSON file (using any library) and navigate there.

9. Read about GSON library
    - Create a class named Config and deserialize the (above) JSON file content into it.

- In the deserialized class create a getter for URL (getURL)
- Use getURL to run your test