Implement framework and auto tests, based on provided testcase.

Task basics

- You'll need to implement basic framework, which should include the next parts:
    1. BaseForm class
    2. BaseElement class and child classes for elements
    3. Singleton & BrowserFactory (Factory method) to organize process of getting driver instance
    4. Utility class to work with driver
    5. ConfigManager class

- All of the assertions should be implemented by using special testing libraries (Java – TestNG, C# - NUnit, Python – PyTest, JS - mocha+chai)

  Selenium WebDriver should be used for working with browser, pages' elements, etc.

  Tests should be working on both Chrome and Firefox browsers.

Requirements

1. You should implement Singleton и FactoryMethod patterns in your solution.
2. There should be an option in your solution to choose a browser which will be used in test - Chrome or Firefox.
3. Page Object pattern should be used in your solution.
4. BaseForm Class should be implemented (and all created pages/forms should be inherited from it.
5. Pre- and postconditions should be used in your solution.
6. Logger should be used in your solution.
7. One of created autotests should be using DDT approach and be parametrized.
8. Test data and configurational data should be stored in separate files.
9. Solution structure should be organized: different folders, namespaces/packages/etc. - it should be clear which class go where.

Site

https://demoqa.com/

Test scenarios

#1. Alerts

#	Step	Expected result
1	Navigate to main page	Main page is open
2	Click on **Alerts, Frame & Windows** button. In a menu click **Alerts** button.	Alerts form has appeared on page
3	Click on **Click Button to see alert** button	Alert with text "You clicked a button" is open
4	Click **OK** button	Alert has closed
5	Click on **On button click, confirm box will appear** button	Alert with text "Do you confirm action?" is open
6	Click on **OK** button	Alert has closed Text "You selected Ok" has appeared on page
7	Click on **On button click, prompt box will appear** button	Alert with text "Please enter your name" is open
8	Enter **randomly generated** text, click **OK** button	Alert has closed Appeared text equals to the one you've entered before

#2. Iframe

#	Step	Expected result
1	Navigate to main page	Main page is open
	Click on **Alerts, Frame & Windows** button	Page with Nested Frames form is open.

#2. Iframe

#	Step	Expected result
1	Navigate to main page	Main page is open
2	Click on **Alerts, Frame & Windows** button In a menu click **Nested Frames** button	Page with Nested Frames form is open. There are messages "Parent frame" & "Child Iframe" present on page
3	Select **Frames** option in a left menu	Page with Frames form is open. Message from upper frame is equal to the message from lower frame

#3. Tables

User №	First Name	Last Name	Email	Age	Salary	Department
1	Jon	Snow	knownothing@gmail.com	30	3000	alpha
2	Buttercup	Cumbersnatch	BudapestCandygram@mail.io	41	2000	beta

#	Step	Expected result
1	Navigate to main page	Main page is open
2	Click on **Elements** button In the menu click a **Web Tables** button	Page with **Web Tables** form is open
3	Click on **Add** button	**Registration Form** has appeared on page
4	Enter data for **User №** from the table and then click **Submit** button	Registration form has closed. Data of **User №** has appeared in a table

5	Click **Delete** button in a row which contains data of **User №**	Number of records in table has changed Data of **User №** has been deleted from table

#4. Handles

#	Step	Expected result
1	Navigate to main page	Main page is open
2	Click on **Alerts, Frame & Windows** button In the menu click a **Browser Windows** button	Page with Browser Windows form is open
3	Click on **New Tab** button	New tab with sample page is open
4	Close current tab	Page with Browser Windows form is open
5	In the menu on the left click **Elements → Links** button	Page with Links form is open
6	Click on **Home** link	New tab with main page is open
7	Resume to previous tab	Page with Links form is open