

# By OnlineInterviewQuestions.com

## Protractor Interview Questions

### Q1. How to use Cucumber with Protractor?

To use **Cucumber.js**, you should update the framework and add a framework path, and then add a few cucumberOpts that specify where to find the step definition files such that it will now run the protractor cucumber.

### Q2. What is difference between WebDriver click() and JavaScript click()?

WebbDriver click() simulates real user interaction with the UI whereas JavaScript click() executes JavaScript in the currently selected frame or window.

### Q3. How to select an option from dropdown in Protractor e2e testing?

You can select an option from the dropdown in Protractor e2e testing with the following step:

- Find the dropdown using the element.
- Click the dropdown.
- Click the option.

### Q4. How to set default browser window size in Protractor/WebdriverJS?

You can use the following code to set the default window size in Protractor/WebdriverJs:

```
browser.executeScript('window.moveTo(0,0);' + 'window.resizeTo(screen.width, screen.height);');
```

### Q5. How to getText on an input in protractor?

To get a text on an input in the protractor you can use *getText()* function.

### Q6. How to locate elements that have a specific class name.in Protractor?

**webdriver.By.className** is used in Protractor to locates elements that have a specific class name.

### Q7. How to create promises in Protractor?

With the help of following syntax you can create a promise in protractor, the code snippet is:

```
var deferred = protractor.promise.defer();  
var promise = deferred.promise;
```

### Q8. How to install Protractor for?

Use npm to install Protractor globally with ***npm install -g protractor***. This will install two command-line tools, **protractor** and **webdriver-manager**. By running ***protractor --version*** you can insure it's working.

### Q9. How can I get the current URL using a protractor?

You can get the current URL using protractor by using **browser.getCurrentUrl()** as follows:

```
expect(browser.getCurrentUrl()).toEqual("expectedUrl");
```

### Q10. What is browser.ignoreSynchronization in protractor?

**browser.ignoreSynchronization** is the driver method used directly to find the elements such that the system will try to find them without waiting for any ongoing \$http requests to finish, It can be done by setting ***browser.ignoreSynchronization = true*** .

### Q11. List the different mouse actions that can be performed using protractor?

**Mouse actions** are the demonstration of mouse activities (such as hover, clicking, drag, and drop) that can be easily simulated in Selenium Protractor with the predefined methods for mouse movement, clicking, and others.

Different mouse actions that can be performed using protractor are as follows:

1. `mouseMove()`
2. `dragAndDrop( source , target )`
3. `click()`
4. `doubleClick()`
5. `mouseUp()`
6. `mouseDown()`

7. contextClick()
8. clickAndHold()
9. dragAndDropBy(source, xOffset, yOffset)
10. moveByOffset(xOffset, yOffset)
11. moveToElement(toElement)
12. release()
13. mouseMove()
14. mouseHover()

**Q12. List the major differences between Protractor vs Selenium?**

Major differences between Protractor and Selenium are as follows:

## **Selenium WebDriver**

You can automate the web applications using the selenium WebDriver with Java or any programming language of choice.

The languages supported by Selenium are Java, Python Kotlin, C, C#, PHP. Supports many frameworks of unit testing, such as TestNg and Junit.

You need a lesser number of code lines for testing with Selenium. Selenium's performance is faster for a non-Angular web application.

It is open-source software.

Enough options for locators.

Debugging is easy owing to Java-based design.

Technical skills mandatory to work with this tool are Java and TestNG.

Licensed under Eclipse.

The framework is proven and widely used.

Compatible with Windows and Linux.

Works well with Angular pages too; however, page synchronization issues may crop up.

Proven, widely used for automation, well-extended support, and stable API.

You can work in popular programming languages such as Python, Ruby, Java, Perl, C, and C##.

Offers comprehensive support for all sorts of testing needs for web applications.

## **Protractor**

Prefer Protractor when you are not comfortable with angular waits using fluent wait in Selenium.

All js related languages are supported.

JavaScript.

Protractor works well with Jasmine.

Code length is longer when you use promises.

return promises, so you have to solve them.

When working with Angular applications.

It is also open-source software.

Lesser option for locators.

Debugging needs expertise and time.

One needs to know JavaScript and Node.js.

Licensed under Webstorm7.0.

The framework is budding and hard to maintain.

This is also compatible with Windows and Linux.

Designed specifically for Angular pages. It may face some synchronization issues.

New entrée in the market and needs more time to mature.

Available only for JavaScript.

This is just a wrapper developed over Selenium. All browser automation are developed using Selenium.

Cannot trace and work with angular directives in a code.

Rich in angular specific locator strat