# Exam Tasks from Telerik QA Track 2013

## Problem 1. Bug Reports

You are given two **videos showing bugs** in various functionalities. Your task is to watch them, navigate to the URL shown in the video and **prepare bug reports**.

* Identify the bug
* Reproduce it
* Write bug report

### ASP AJAX control bug

Watch <http://tinyurl.com/qa-video-1> and <http://tinyurl.com/qa-video-2>.

Go to <http://www.telerik.com/products/aspnet-ajax.aspx> and try to reproduce them.

Prepare a **bug report** containing all **necessary information** for logging the report in a bug tracking system. Try to **isolate** the minimum steps required for reproducing the bug, removing any steps that are not really influencing the behavior.

The **reports** should be presented in a **table in word** document.

Hint: Be careful, there may be more than one bug on a video! Consider an appropriate test oracle.

## Problem 2. Yahoo Test

**Selenium IDE**

Make functional web tests using Selenium IDE.

The tests should navigate to the following URL: <http://www.yahoo.com/>.

Prerequisite: Create an account in Yahoo which will be used for tests.

Record the following tests:

1. Log into a valid Yahoo account and verify that the **log in** was successful and the correct user is logged in
2. While you are logged in make a test that **sends email** to you. In the message subject write the following text "QA Exam #1"
3. Make sure that the message is sent using the information about **Sent** messages

* Verify that the field **To** contains your email address
* Verify that the field **Subject** has title "QA Exam #1"

1. Take the test above and instead of writting "QA Exam #1", create it with an unique subject using JavaScript. (Tip: You can generate an unique name in separate variable)

## Problem 3. jqGrid Test

**Selenium IDE**

Make functional web tests using Selenium IDE.

Navigate to the following URL: <http://www.trirand.net/demo/aspnet/webforms/jqgrid/> and select Hierarchy -> Sub Grid (2 nested levels) from the left menu. That will open the jqGrid that we are going to use. (Note that you should follow these steps every time you make a test).

1. Verify that the table headers are exactly (ID, CompanyName, ContactName, Phone and City)
2. Expand the row with ID="ANTON" and verify that the CompanyName value from the expaned row is equal to ShipName value in the the first row of the inner jqGrid.

**Selenium WebDriver**

Convert the test cases (A, B) from the task to C# and test if they work properly. Run the tests a couple of times to verify that they work correctly.

Make functional web test using Selenium WebDriver to test paging functionality of the jqGrid. From the combo box select **consequently 10, 20 and 30** and verify that changing the page count is working correctly i.e. when 10 is selected, the rows in the **main table** should be exactly 10 etc. Be careful to count only these rows that are visible in the table.

## Problem 4. Test Studio Stand Alone

Make functional web tests using Test Studio.

Download the project [here](http://tinyurl.com/qa-academy-test-studio):

Run the project following this steps:

1. Unzip the project
2. Open Visual Studio and choose File -> Open -> Web Site
3. Navigate to the unzipped project
4. Run the project

Once you run the project record the following tests:

* Create a new record.
* Add a data source to the test above for Product Name and Unit Price value. The data source should look like this:

|  |  |
| --- | --- |
| Product Name | Unit Price |
| Product1 | 1 |
| Product2 | 2 |
| Product3 | 3 |

* Filter the products by Product Name *Contains* “Product”.

Use coded step for the folowing validations:

* + Vetify that the **Count** in the footer shows 3.
  + Verify that the **First** element in the footer displays the first row’s Product name.
  + Verify the products are filtered by checking if the sum of the unit prices of the three products equals the **Sum** of the footer.
* Verify that the editing and deleting is working correctly. Use coded step(s) for clicking (locating) the elements and validations.

Hint: Using **Reset Date to its initial state** button might be useful.