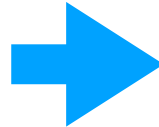


Automated UI Testing

CPSC 310

How you create tests in D3?

```
{  
  "WHERE": {  
    "GT": {  
      "courses_avg": 97  
    }  
  },  
  "OPTIONS": {  
    "COLUMNS": [  
      "courses_dept",  
      "courses_avg"  
    ],  
    "ORDER": "courses_avg"  
  }  
}
```



Campus Explorer Plain Courses Rooms

Conditions
☒ All of the following ☐ Any of the following ☐ None of the following

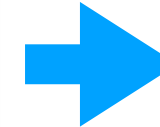
Columns
☐ Audit ☐ Average ☐ Department ☐ Fall
☐ ID ☐ Instructor ☐ Pass ☐ Title
☐ UIDP ☐ Year

Order
☐ Descending

Groups
☐ Audit ☐ Average ☐ Department ☐ Fall
☐ ID ☐ Instructor ☐ Pass ☐ Title
☐ UIDP ☐ Year

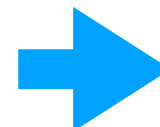
Transformations

Copy HTML Submit



```
<form data-type="rooms">  
  <div class="form-group conditions">  
    <legend>Conditions</legend>  
    <div class="control-group condition-type">  
      <div class="control conditions-all-radio">  
        <input checked="" id="rooms-conditiontype-all" type="radio"/>  
        <label for="rooms-conditiontype-all">All</label>  
      </div>  
      <div class="control conditions-any-radio">  
        <input id="rooms-conditiontype-any" type="radio"/>  
        <label for="rooms-conditiontype-any">Any</label>  
      </div>  
    </div>  
  </div>  
</form>
```

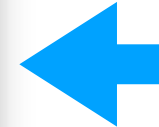
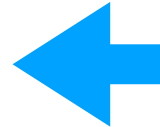
Interact with the UI



Save the current
states

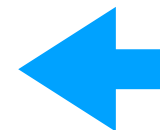
How we run your tests in D3?

```
{
  "WHERE": {
    "GT": {
      "courses_avg": 97
    }
  },
  "OPTIONS": {
    "COLUMNS": [
      "courses_dept",
      "courses_avg"
    ],
    "ORDER": "courses_avg"
  }
}
```



```
<form data-type="rooms">
  <div class="form-group conditions">
    <legend>Conditions</legend>
    <div class="control-group condition-type">
      <div class="control conditions-all-radio">
        <input checked="" id="rooms-conditiontype-all" type="radio"/>
        <label for="rooms-conditiontype-all">All</label>
      </div>
      <div class="control conditions-any-radio">
        <input id="rooms-conditiontype-any" type="radio"/>
        <label for="rooms-conditiontype-any">Any</label>
      </div>
    </div>
  </div>
</form>
```

Simulate the
“submit” click and
expect something



Setup a fixed
environment from
the saved states

What if...

CampusExplorer Plain Courses Rooms

Conditions
☒ All of the following ☐ Any of the following ☐ None of the following

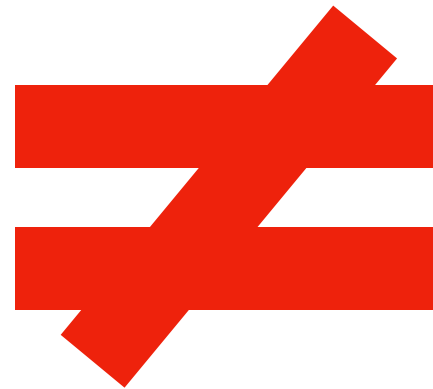
Columns
☐ Audit ☐ Average ☐ Department ☐ Fail
☐ ID ☐ Instructor ☐ Pass ☐ Title
☐ UUID ☐ Year

Order
Audit
Average
Department
Fail
ID
Instructor
Pass
☐ Descending

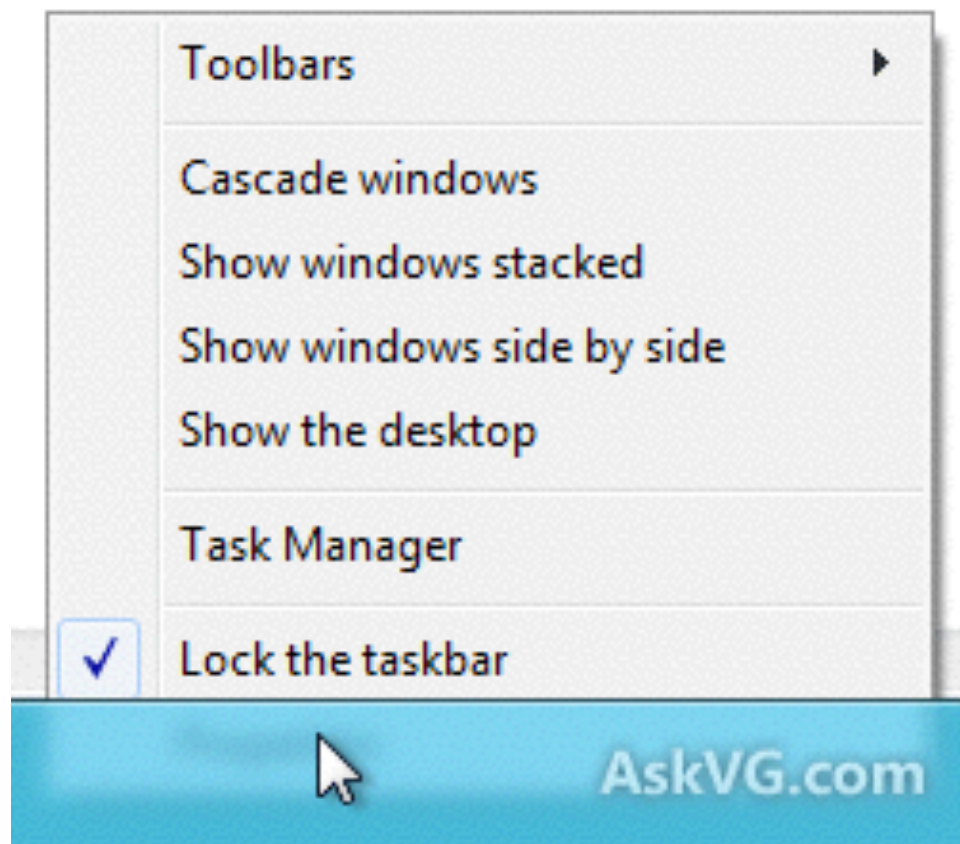
Groups
☐ Audit ☐ Average ☐ Department ☐ Fail
☐ ID ☐ Instructor ☐ Pass ☐ Title
☐ UUID ☐ Year

Transformations

Copy HTML Submit



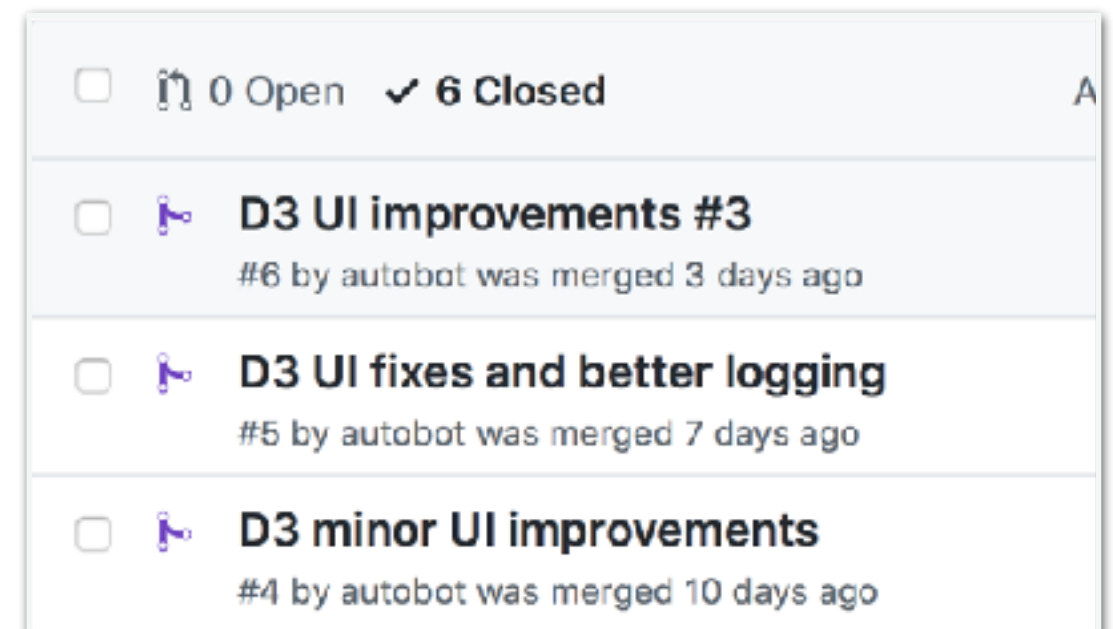
```
<form data-type="rooms">  
  <div class="form-group conditions">  
    <legend>Conditions</legend>  
    <div class="control-group condition-type">  
      <div class="control conditions-all-radio">  
        <input checked="" id="rooms-conditiontype-all" type="radio"/>  
        <label for="rooms-conditiontype-all">All</label>  
      </div>  
      <div class="control conditions-any-radio">  
        <input id="rooms-conditiontype-any" type="radio"/>  
        <label for="rooms-conditiontype-any">Any</label>  
      </div>  
    </div>  
  </div>  
</form>
```



Some glitches in the reference UI

1. Unchecking the options in columns or groups doesn't work. The copied html file will still have these options checked. The page has to be reloaded.
 2. In transformations, if you want use the COUNT operator, you have to select something else then select COUNT back, otherwise nothing will be selected in the html file.
- Also, it seems like when I use rooms dataset, the global variable document is not updated. Every time I test a rooms query, buildQuery returns the last courses query's result.

d3



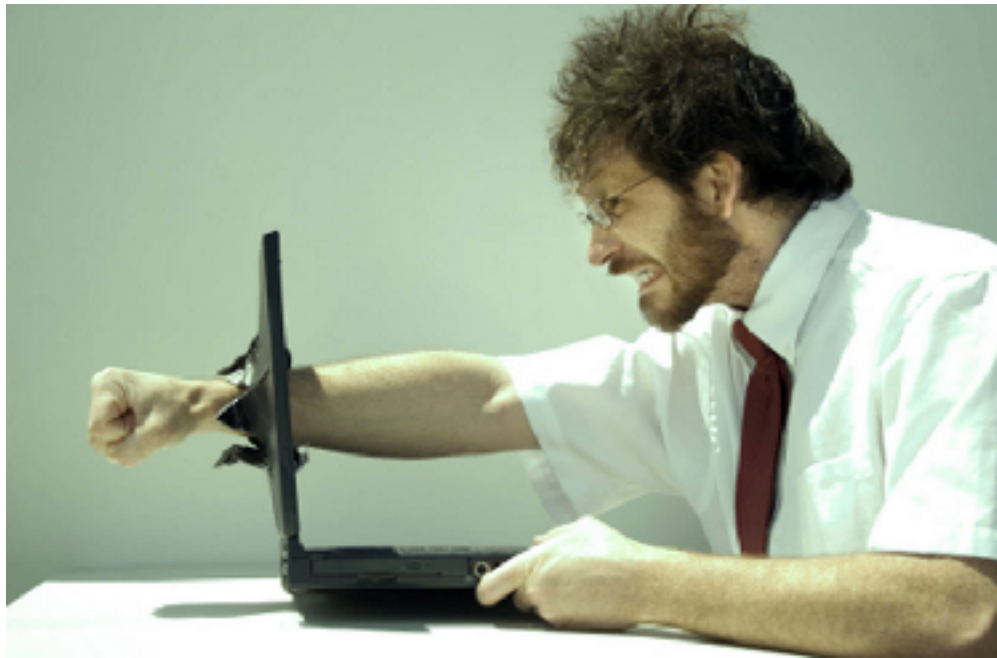
“UI can be buggy”

Maybe we need to test it

UI Testing

“In software engineering, graphical user interface testing is the process of testing a product's graphical user interface to ensure it meets its specifications. This is normally done through the use of a variety of test cases.”

Common Approaches



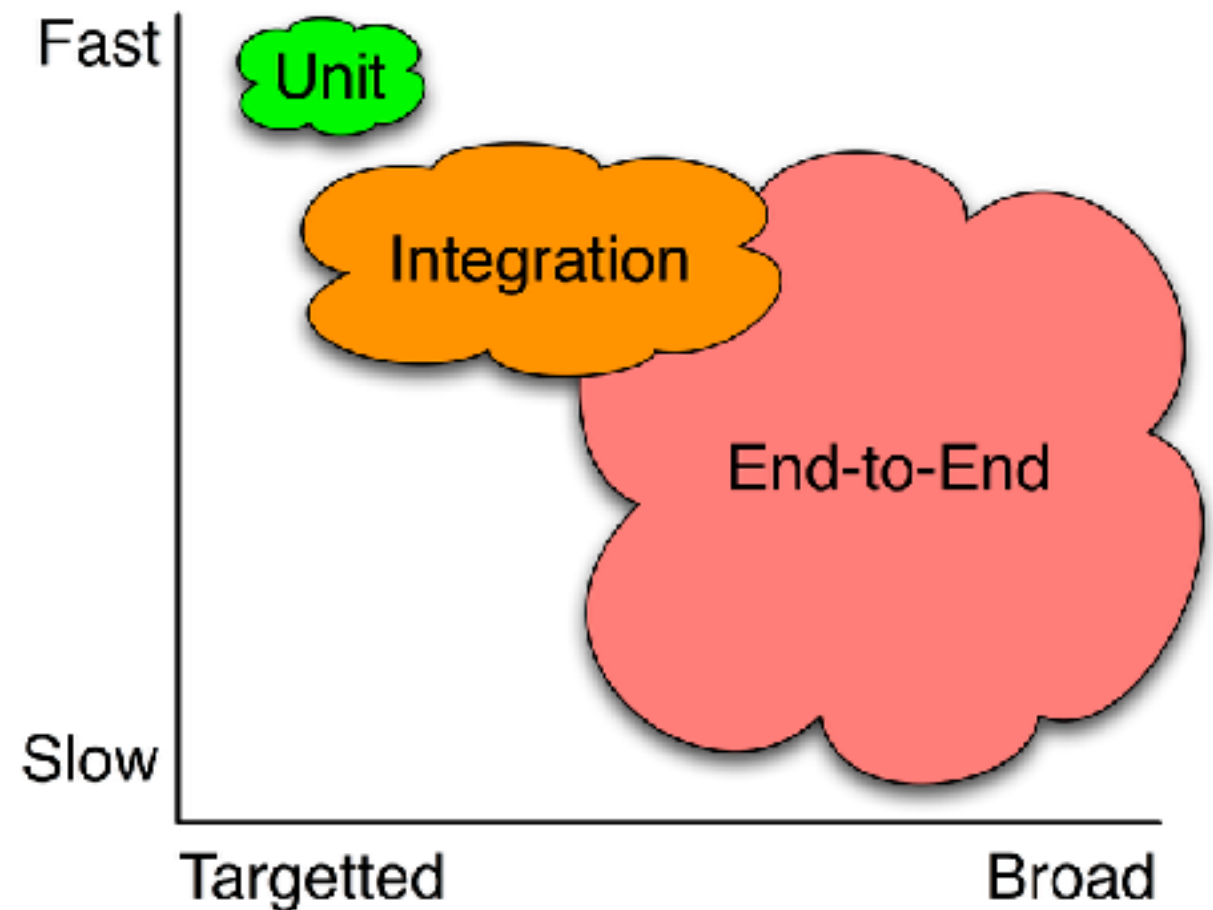
Manual Test



Automated Test

Automated UI Testing

- Unit tests are great
- Unit tests only focus on small parts
- Automated UI tests are integration tests even End-to-End tests



Automated UI Testing

1. UI automation
2. Testing check
 1. Screen render properly
 2. Navigations work
 3. ...

Tools









- A test runner
- Spawn a web server
- Execute source code against test code
- Examine/display the results



A test framework

```
var assert = require('assert');
describe('Array', function() {
  describe('#indexOf()', function() {
    it('should return -1 when the value is not present',
      function() {
        assert.equal([1,2,3].indexOf(4), -1);
      });
  });
});
```

<input type="checkbox"/>		0 Open	<input checked="" type="checkbox"/> 6 Closed	A
<input type="checkbox"/>		D3 UI improvements #3	#6 by autobot was merged 3 days ago	
<input type="checkbox"/>		D3 UI fixes and better logging	#5 by autobot was merged 7 days ago	
<input type="checkbox"/>		D3 minor UI improvements	#4 by autobot was merged 10 days ago	

“Hands on time”

A case study: UI bugs in reference UI

Related Topic

- UI Automation
 - Web crawls (obtain information)
 - Batch tasks (repetitive boring tasks)
 - Cron jobs (ticket booking bots)
 - ...