HOW TO RUN AND VERIFY RESULTS?

KARMA START

```
→ js-tests git:(task-11710) ✗ karma start
28 06 2017 12:47:59.709:WARN [karma]: No captured browser, open http://localhost:9876/
28 06 2017 12:47:59.728:INFO [karma]: Karma v0.13.22 server started at http://localhost:9876/
28 06 2017 12:47:59.743:INFO [launcher]: Starting browser PhantomJS
28 06 2017 12:48:02.220:INFO [PhantomJS 1.9.8 (Mac OS X 0.0.0)]: Connected on socket 6KKt9wGHtvCeO_rFAAAA with id 36241519
PhantomJS 1.9.8 (Mac OS X 0.0.0) LOG: 'JQMIGRATE: Logging is active'
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
.....................................................................
ERROR: 'LOGISMATA: Error: "Disability for client 1" scenario impossible to be calculated.'
.....................................................................
.....................................................................
.....................................................................
.....................................................................
...........................................................
PhantomJS 1.9.8 (Mac OS X 0.0.0): Executed 1082 of 1095 (skipped 13) SUCCESS (4.075 secs / -301669 mins 38.421 secs)
```

ENV KARMA_SPECS="SPEC/ONLINE-MORTGAGE/ONLINE-MORTGAGE-STRATEGY-CONTROLLER.JS" KARMA START

Success

```
→ js-tests git:(task-11710) ✗ env KARMA_SPECS="spec/online-mortgage/online-mortgage-strategy-controller.js" karma start
28 06 2017 12:53:48.669:WARN [karma]: No captured browser, open http://localhost:9876/
28 06 2017 12:53:48.687:INFO [karma]: Karma v0.13.22 server started at http://localhost:9876/
28 06 2017 12:53:48.702:INFO [launcher]: Starting browser PhantomJS
28 06 2017 12:53:50.983:INFO [PhantomJS 1.9.8 (Mac OS X 0.0.0)]: Connected on socket ruTVBKKUPGl3kGZvAAAA with id 91572973
PhantomJS 1.9.8 (Mac OS X 0.0.0) LOG: 'JQMIGRATE: Logging is active'
.........
PhantomJS 1.9.8 (Mac OS X 0.0.0): Executed 9 of 9 SUCCESS (0.119 secs / 0.078 secs)
```

Failed

```
→ js-tests git:(task-11710) ✗ env KARMA_SPECS="spec/online-mortgage/online-mortgage-strategy-controller.js" karma start
28 06 2017 12:59:27.012:WARN [karma]: No captured browser, open http://localhost:9876/
28 06 2017 12:59:27.030:INFO [karma]: Karma v0.13.22 server started at http://localhost:9876/
28 06 2017 12:59:27.045:INFO [launcher]: Starting browser PhantomJS
28 06 2017 12:59:29.330:INFO [PhantomJS 1.9.8 (Mac OS X 0.0.0)]: Connected on socket wAyZyw55osOfMhBTAAAA with id 51131856
PhantomJS 1.9.8 (Mac OS X 0.0.0) LOG: 'JQMIGRATE: Logging is active'
PhantomJS 1.9.8 (Mac OS X 0.0.0) Controller: OnlineMortgage.StrategyCtrl addEmptyTranche test FAILED
        Expected [ Object({ kind: 2, duration: null, amount: null, date_of_payment: null }) ] to equal [ Object({ kind: null, duration: nu
ll, amount: null, date_of_payment: null }) ].
            at /Users/user/projects/mmp/src/mmp/static/js-tests/spec/online-mortgage/online-mortgage-strategy-controller.js:41
...
PhantomJS 1.9.8 (Mac OS X 0.0.0) Controller: OnlineMortgage.StrategyCtrl deleteTranche test (remove last tranche) FAILED
        Expected [ Object({ kind: 2, duration: null, amount: null, date_of_payment: null }) ] to equal [ Object({ kind: null, duration: nu
ll, amount: null, date_of_payment: null }) ].
            at /Users/user/projects/mmp/src/mmp/static/js-tests/spec/online-mortgage/online-mortgage-strategy-controller.js:126
....
PhantomJS 1.9.8 (Mac OS X 0.0.0): Executed 9 of 9 (2 FAILED) (0.087 secs / 0.08 secs)
```

```javascript
describe('Controller: OnlineMortgage.StrategyCtrl', function() {

    // Instantiate a new version of module before each test
    beforeEach(module('OnlineMortgage.StrategyApp'));

    var scope, onlineMortgageStrategyCtrl, onlineMortgageMock;

    // Before each unit test, instantiate a new instance of the
    // controller
    beforeEach(inject(function($rootScope, $controller) {
        // define the mock for onlineMortgage service
        onlineMortgageMock = {
            tranches: []
        };
        scope = $rootScope.$new();
        scope.onlineMortgage = onlineMortgageMock;
        onlineMortgageStrategyCtrl =
        $controller('OnlineMortgage.StrategyCtrl', {
            $scope: scope
        });
    }));

    it('remove justAddedTranche tranche', function() {
        var justAddedTranche = {
            kind: 'variable',
            duration: 4,
            amount: 300000
        };
        scope.onlineMortgage.tranches = [justAddedTranche];
        expect(scope.onlineMortgage.tranches.length).toEqual(2);
        expect(justAddedTranche.id).not.toBeDefined();
        var justAddedTrancheIndex =
        scope.onlineMortgage.tranches.indexOf(
            justAddedTranche
        );
        scope.deleteTranche(justAddedTrancheIndex);
        expect(scope.onlineMortgage.tranches.length).toEqual(1);
        expect(scope.onlineMortgage.tranches)
        .not.toContain(justAddedTranche);
    });
```

JASMINE: SPEC STYLE OF TESTING

Jasmine Syntax
➤ describe
➤ beforeEach
➤ it
➤ afterEach
➤ expect

Useful Jasmine Matchers
➤ toEqual
➤ toBe
➤ toBeTruthy
➤ toBeFalsy
➤ toBeDefined
➤ toBeUndefined
➤ toBeNull
➤ toContain
➤ toMatch