Psychometric Preparation for Digital Transition

- Psychometric Training 2016

- Ou Zhang

Today's topic

- History: Paper-Pencil Era, variable map
- Digital Transition
- Content Ingestion Tool (CIT)
- Project Rex (Q-i simulator on PC)
- Using JSON output to verify the variable list

History

- **Paper-Pencil Era**

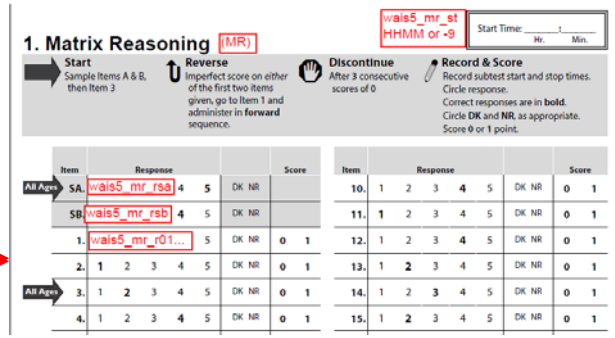
- Materials Needed
  - Manual
  - Stimulus Books
  - Record form
  - Variable list (created by Psychometrics)
  - Variable map (connects the record form)

- What you need to do:
  - Get familiar with the test
  - Understand the administration rules
  - Learn the content in detail
  - Know the Score flow

History (cont.)

- **Paper-Pencil Era (Variable map)**

- record form



Variable Map

SAS Variables	Description	Type Of Data	Acceptable Value
front page variables			
wais5_teid	Test event ID	numeric	0-999999
wais5_eeid	Consent form ID (cfid)	numeric	0-999999
wais5_cl	Clinical Case	numeric	0,1,-9
wais5_sex	Sex	string	M,F, -9
wais5_hand	Handedness	numeric	1-3, -9
wais5_doty	Test date year	numeric	2016
wais5_dotm	Test date month	numeric	1-12
wais5_dotd	Test date day	numeric	1-31
wais5_doby	Birth date year	numeric	1906-2016
wais5_dobm	Birth date month	numeric	1-12
wais5_dobd	Birth date day	numeric	1-31
wais5_pl	Prescription lenses	numeric	0, 1
wais5_plw	Prescription lenses wearing	numeric	0, 1
wais5_ld	Assisted Listening device	numeric	0, 1
wais5_ldw	Assisted Listening device wearing	numeric	0, 1
wais5_wv	Verbatim wearing	string	alphanumeric; -6, -9
wais5_act	Subtest to activate TEID	numeric	1,0, -6
wais5_gn	General Notes	string	alphanumeric; -6
subtest variables			
wais5_mr_st	mr Subtest Start Times	string	HH=01-12;MM=00-59 -9

Variable list

Digital Transition

- What Psychometrics needs to do to fit into this digital transition?
  - Get familiar with the test
  - Understand the administration rules
  - Learn the content in detail (subtests, items, design)
  - Know the Score flow
  - Learn how the test is being administered on Q-i
  - Learn how the output variables reflect the feature of the digital administration software (Q-i)

- But for the digital assessment (especially development phases), the traditional materials may not be available.
  - ~~Manual~~
  - ~~Stimulus Books~~
  - ~~Record form~~

Digital Transition (cont.)

- Paper vs. Digital

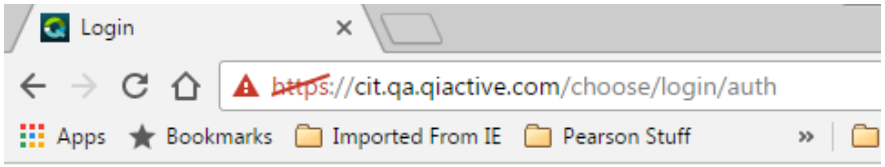
Things you need to know	Materials & Tools	
	Paper	**Digital**
Get familiar with the test	Manual/Stimulus book	CIT
Understand the administration rules	Manual/Record form	CIT/ Project Rex
Learn the content in detail	Manual/Stimulus book	CIT
Know the Score flow	Scoring specs/Record form	CIT/ Project Rex
Learn how the test is being administered on Q-i	----	Project Rex
Learn how the output variables reflect the feature of the digital administration software (Q-i)	Data from Rapid	JSON output

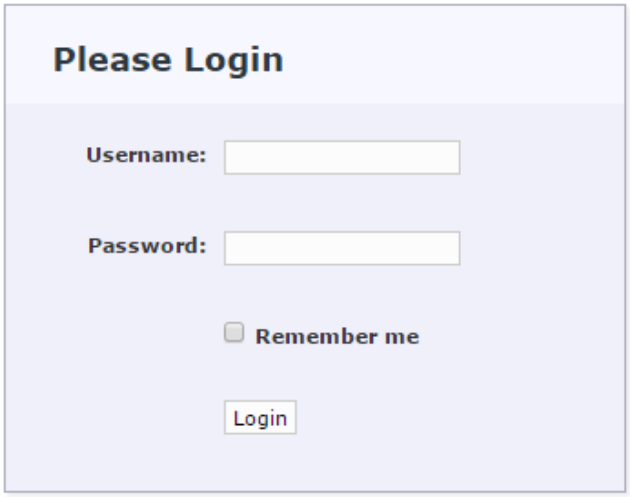
Content Ingestion Tool (CIT)

The CIT is a platform used by RDs and Engineers to communicate how the test is being designed/built on Q-i.

Link: https://cit.qa.qiactive.com/choose/login/auth

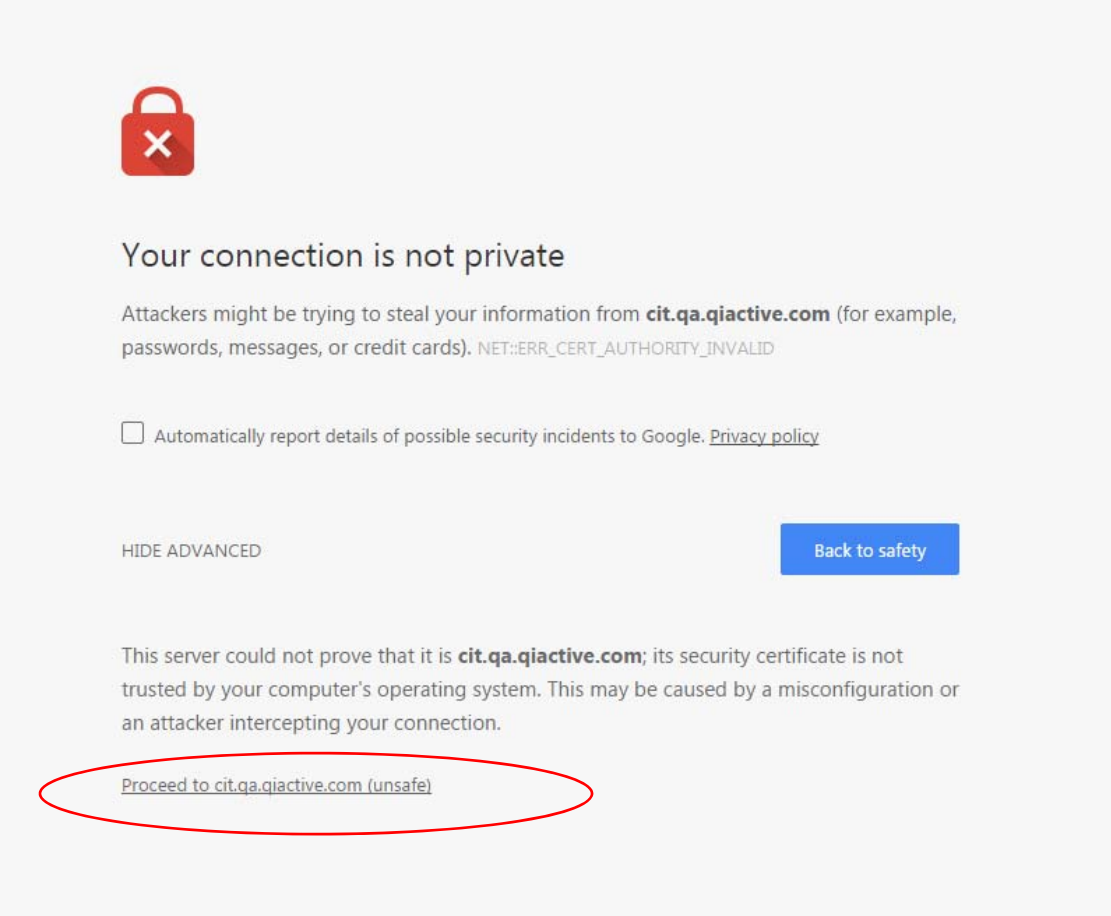
- Request access from Digital Content Dev Team (Kristen Getz, Erik Gallemore, or Regina Tovar).
- Include in your request which tests that you need access to (Read-only access).





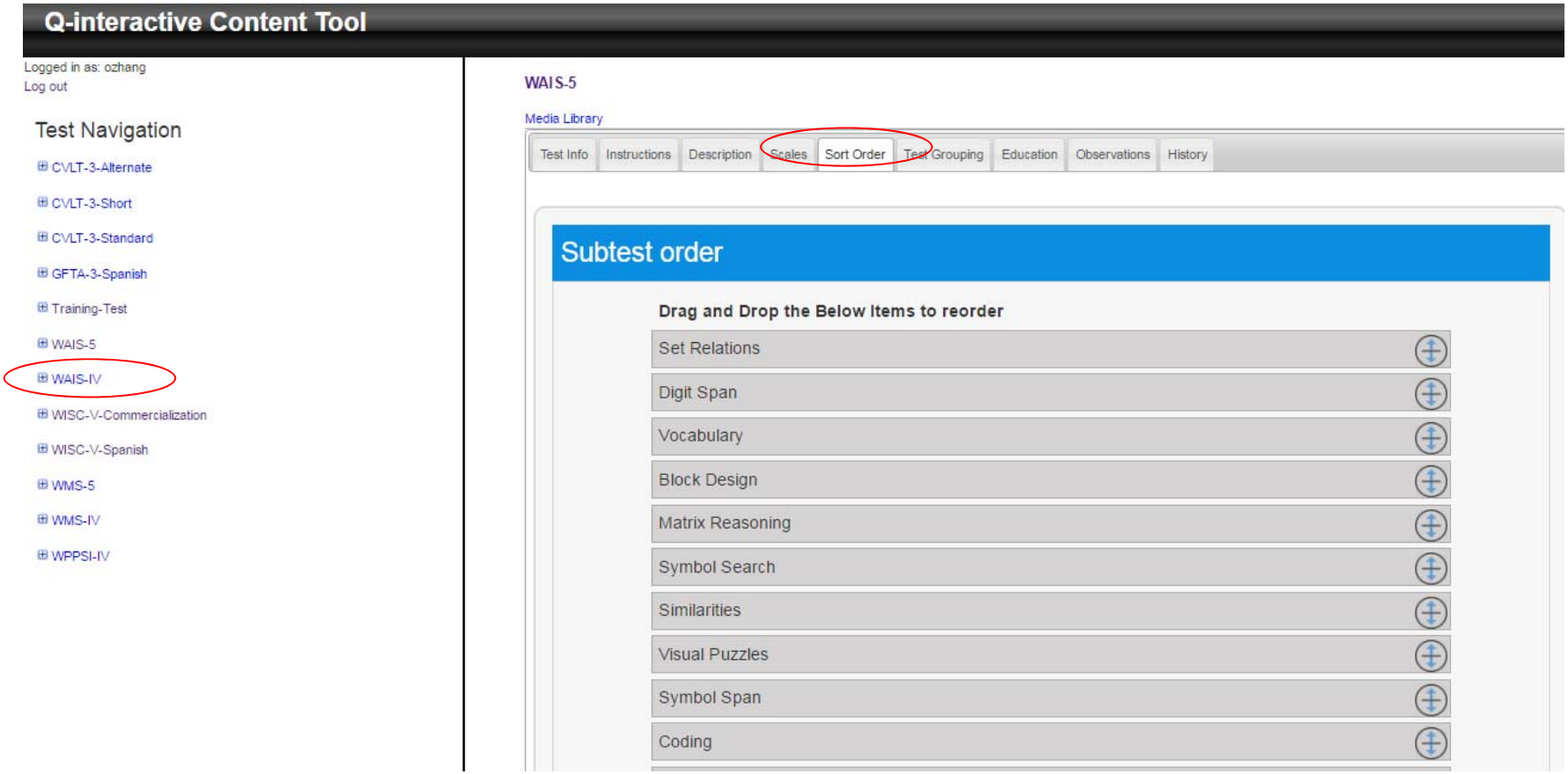
Content Ingestion Tool (cont.)

You may encounter some security issues after login, please proceed and trust the company website.



Content Ingestion Tool (cont.)

After logging in, click on the expansion button (+) next to the test name.



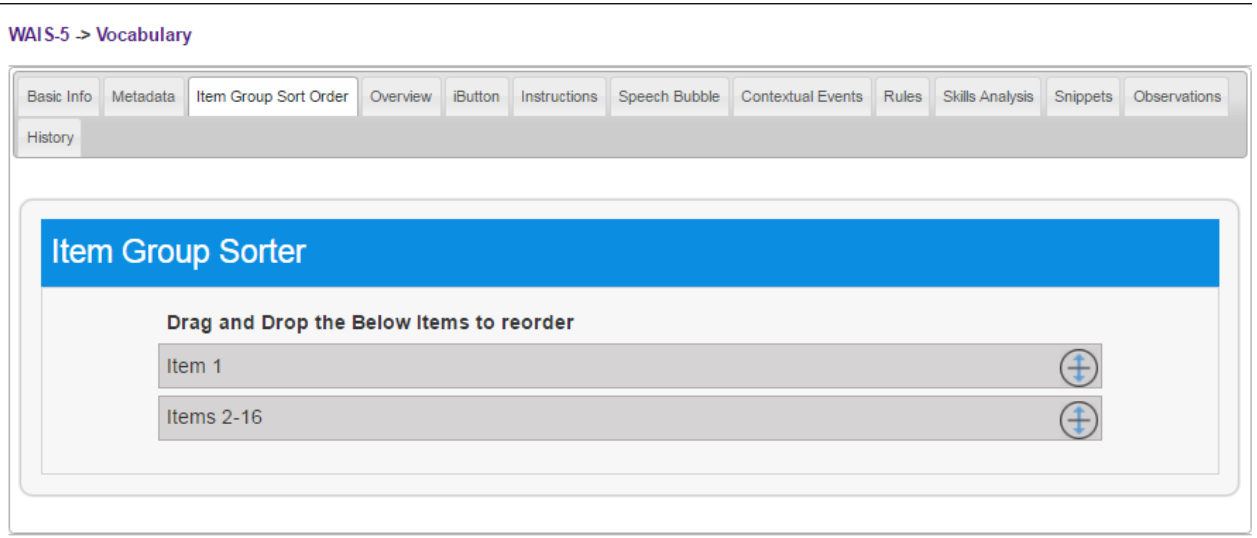
Content Ingestion Tool (cont.)

Then, Click the expansion button (+) to expand/compress the test to subtest level.

There are **13+** tabs of subtest information. Click through each tab to explore.

I've found "**iButton**" and "**Rules**" provide the most information related to test administration/scoring.

- WAIS-5
  - Set Relations
  - Digit Span
  - Vocabulary
  - Block Design
  - Matrix Reasoning
  - Symbol Search
  - Similarities
  - Visual Puzzles
  - Symbol Span
  - Coding
  - Information
  - Figure Weights
  - Response Booklet Symbol Search
  - Arithmetic
  - Response Booklet Coding
  - Auditory Working Memory

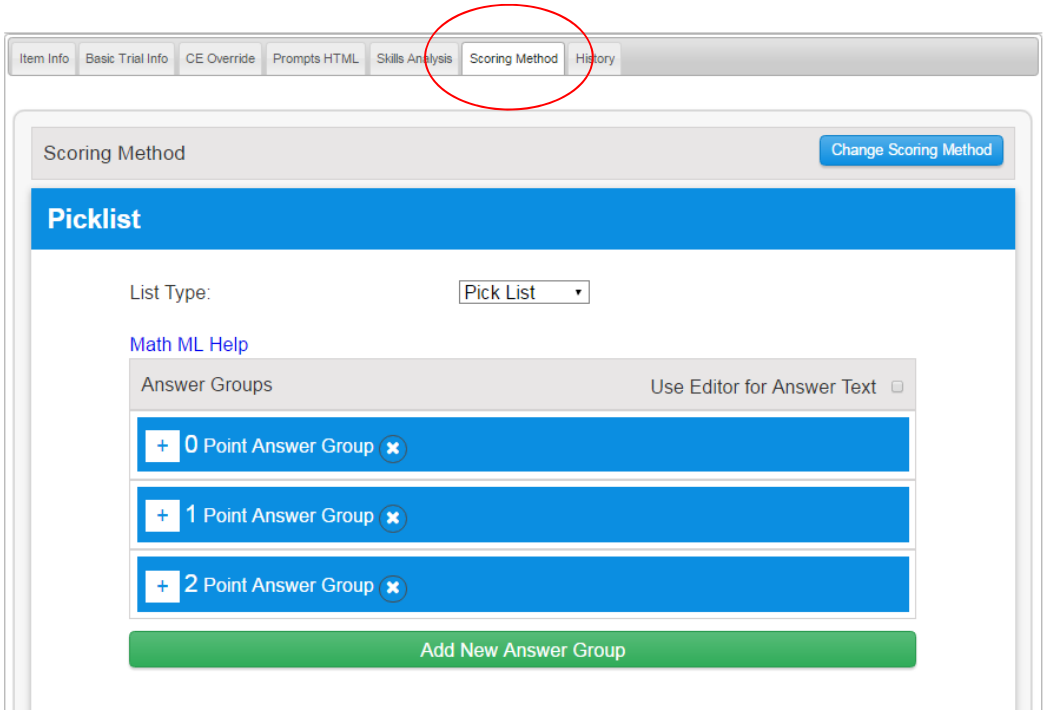


Content Ingestion Tool (cont.)

Then, Click the expansion button (+) to expand/compress the subtest to the item level.

There are **multiple** tabs of item information. Click through each tab to explore.

The "**Scoring Method**" provides the most information related to item scoring.



Content Ingestion Tool (cont.)

In order to understand the test, the most important thing is making sure to click on every possible field while exploring all subtests, item groups and items. Andrea has created a separate slide with more details concerning the available information (K:\flea market\q-i training).

Let's go to the website for a demo.

Link: https://cit.qa.qiactive.com/choose/login/auth

Project Rex simulator (Q-i)

- Project Rex a simulator you can run on your computer that mirrors Q-interactive. It helps Psychometrics understand the administration rules, score flow, and to know how the test is truly being administered on Q-i.

- It helps with all of the variable list/mapping.

- It also helps with creating system cases and edge cases.

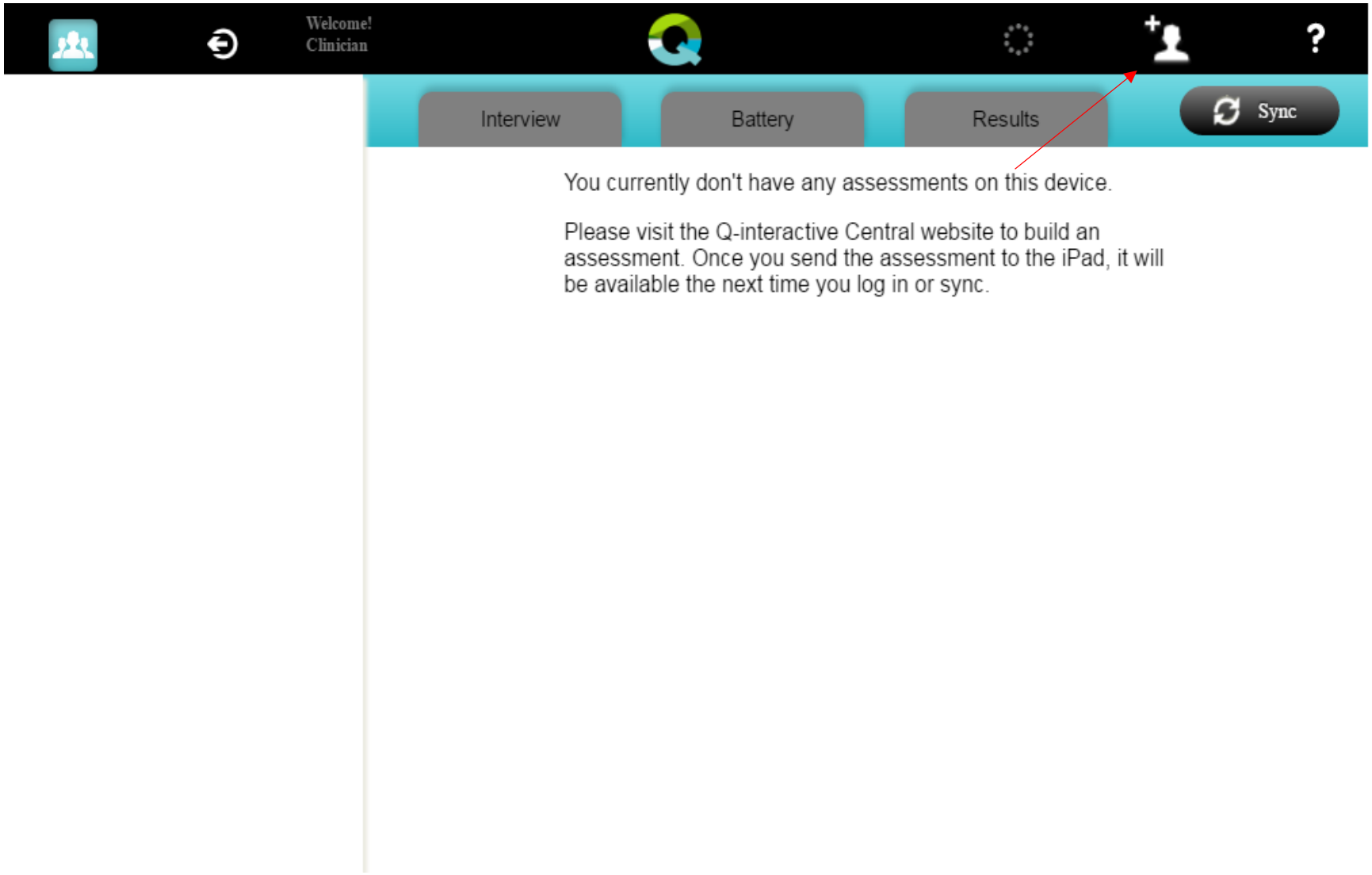
Link: http://10.72.2.226/homeUI_en.html

- What you need to learn today?

  - How to run Project Rex on PC
  - How to start/stop the stopwatch for timed subtests so that the entire subtest items are simulated
  - How to pull your simulated JSON results

Project Rex simulator (cont.)

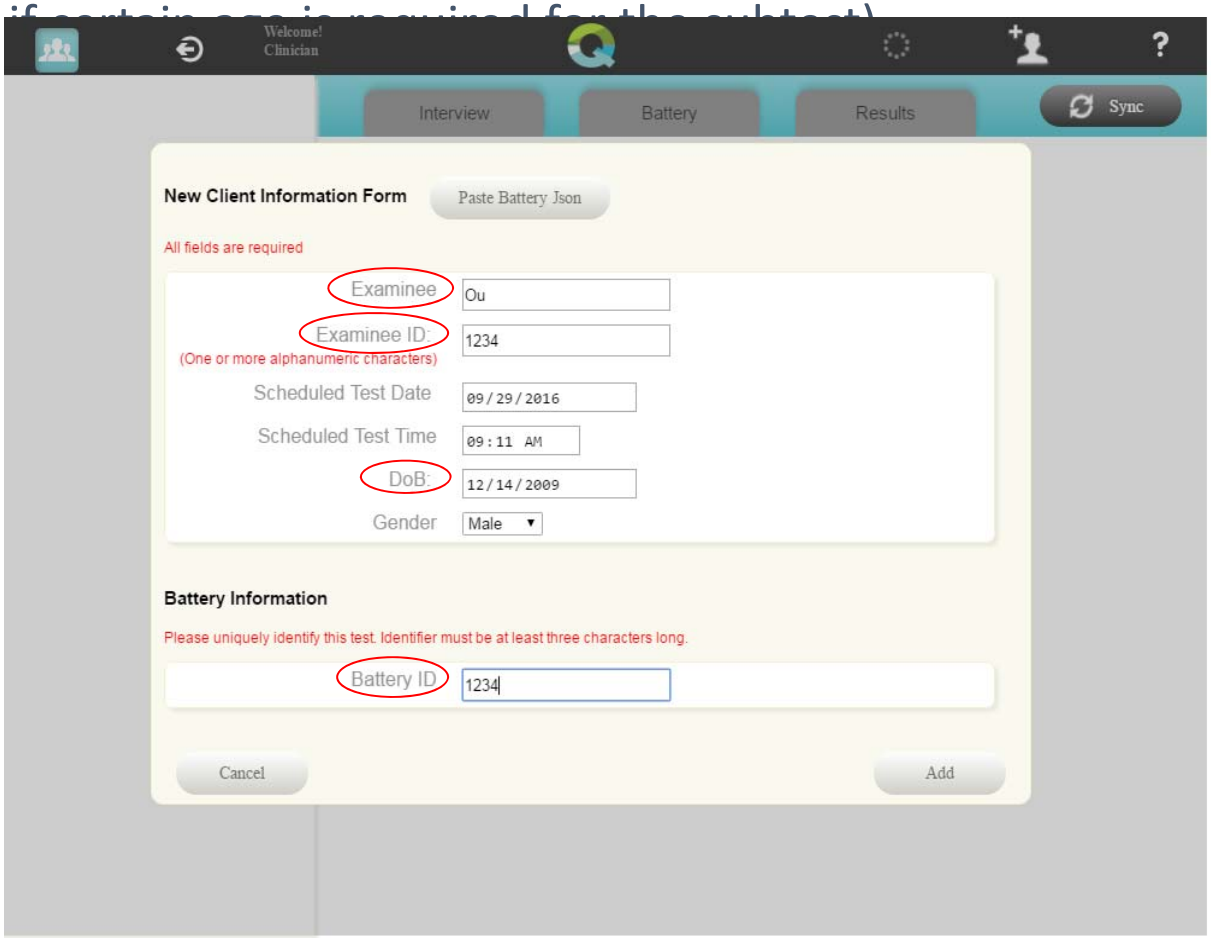
- **Step 1:** Open in the browser and click the little man on the top right



EBR35

Project Rex simulator (cont.)

- **Step 2:** Fill in the following blank cells with mock information:
- Examinee
- Examinee ID
- DoB (optional if certain age is required for the subtest)
- Battery ID



Slide 15

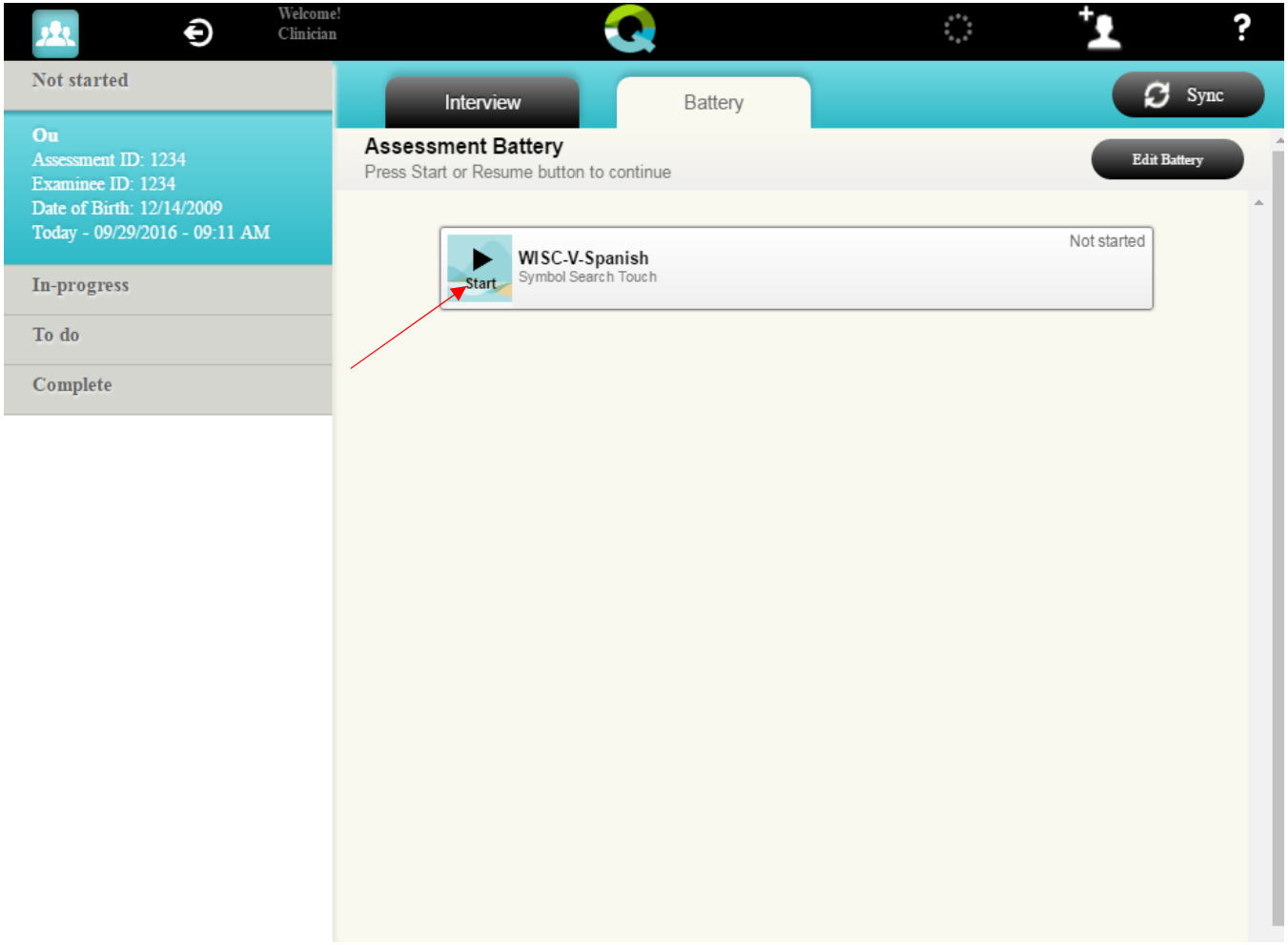
EBR35 Changed entire slide
Ehler, Benjamin R, 10/12/2016

Project Rex simulator (cont.)

- **Step 3:** Choose assessment product and subtests you want to administer/simulate

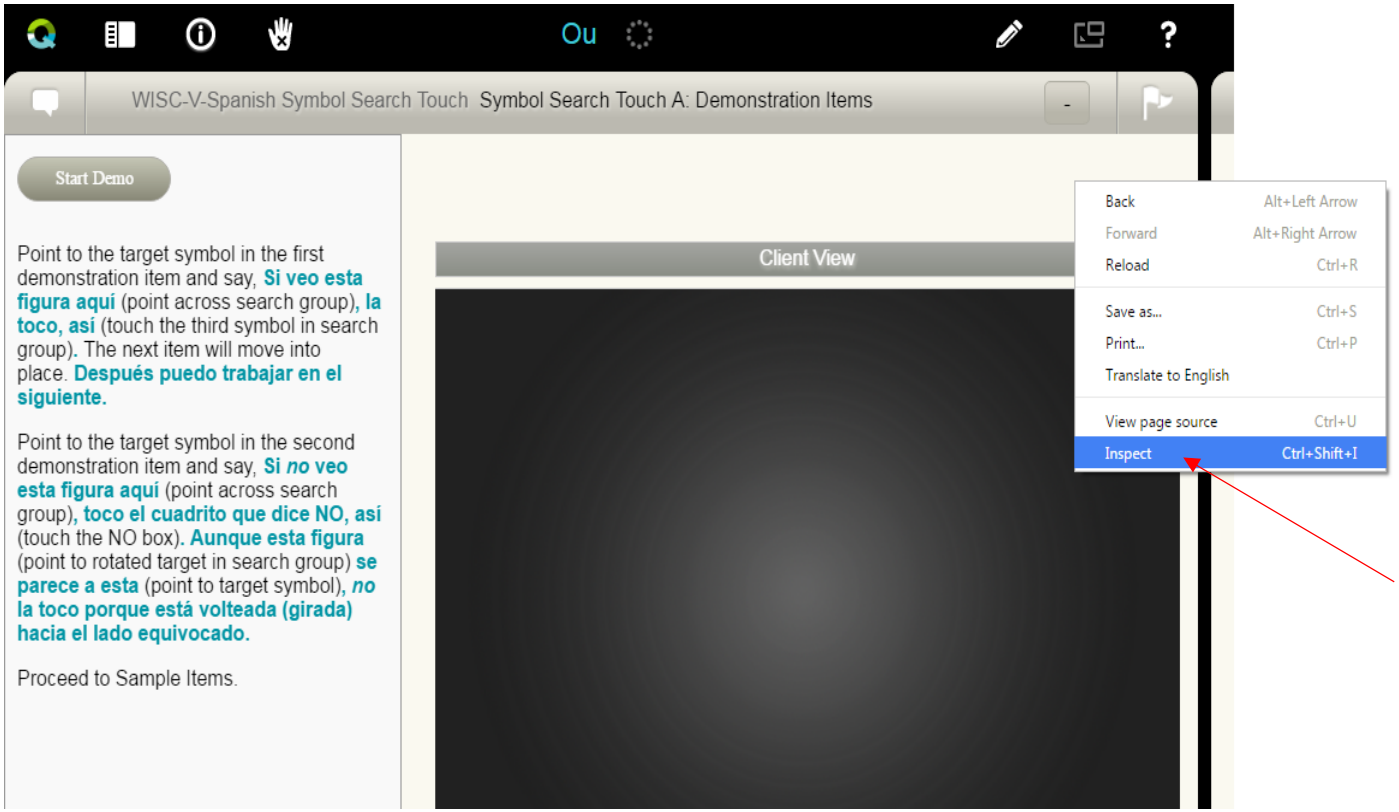
Project Rex simulator (cont.)

- **Step 4:** Launch the test through Project Rex



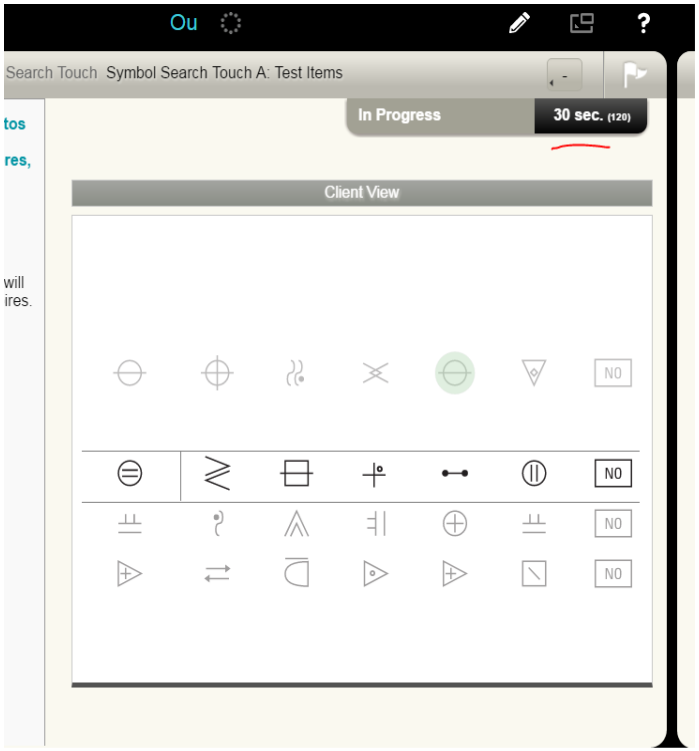
Project Rex simulator (cont.)

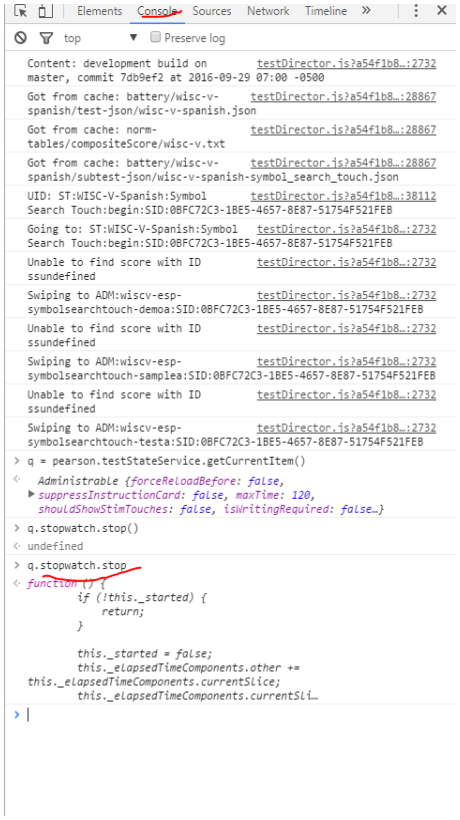
- **Step 5:** Administer test. Once the test is launching, right click your mouse and select “inspect”, which will open up a side-panel with tabs of javascript



Project Rex simulator (cont.)

- **Step 6:** In Console tab, you can either query the current item or Pause the stopwatch



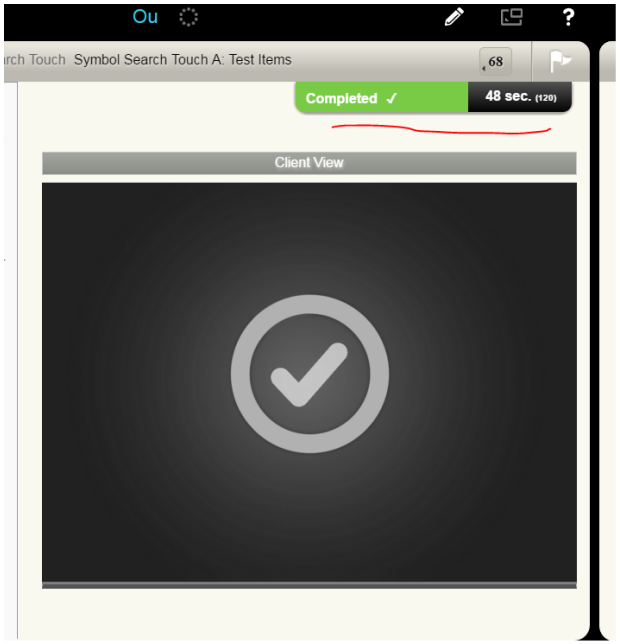


q = pearson.testStateService.getCurrentItem()

q.stopwatch.stop()

Project Rex simulator (cont.)

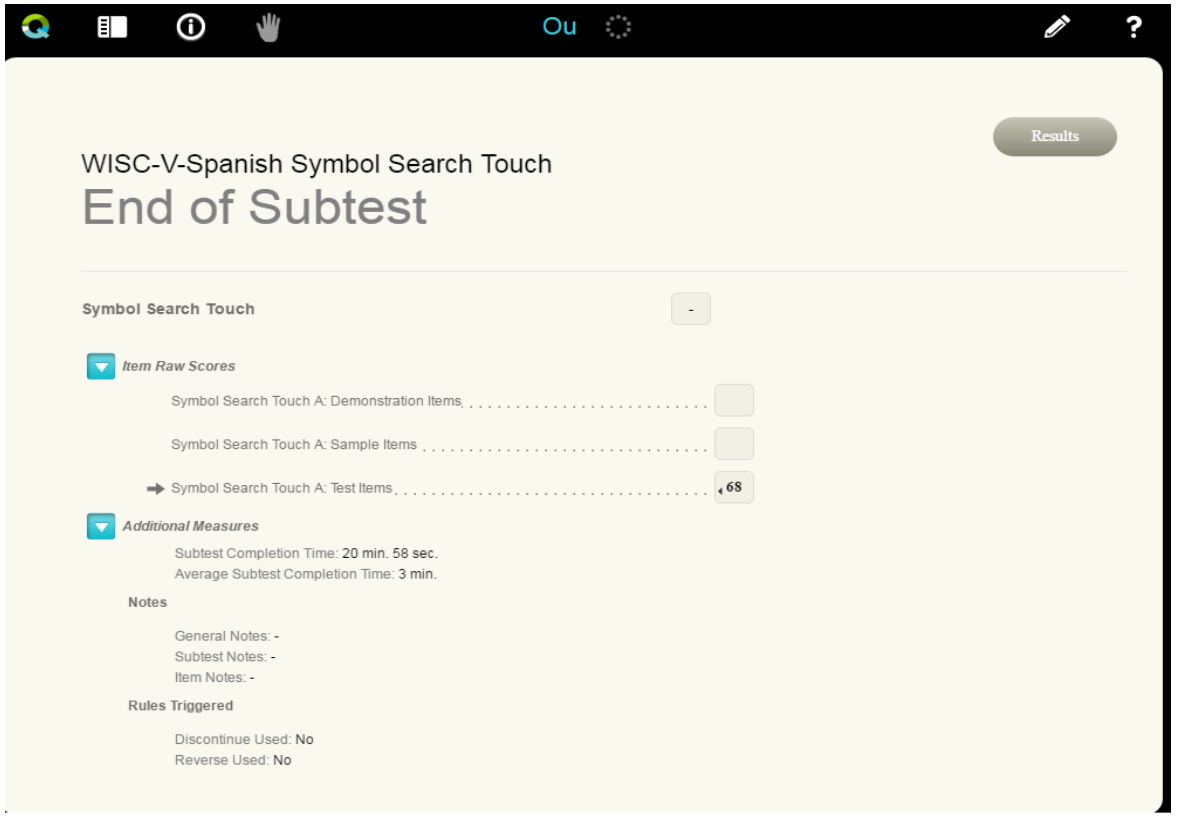
- **Step 7:** Once the subtest is fully administered, restart the stopwatch



q.stopwatch.start()

Project Rex simulator (cont.)

- **Step 9:** Make sure the subtest is fully administered (see the summary page)

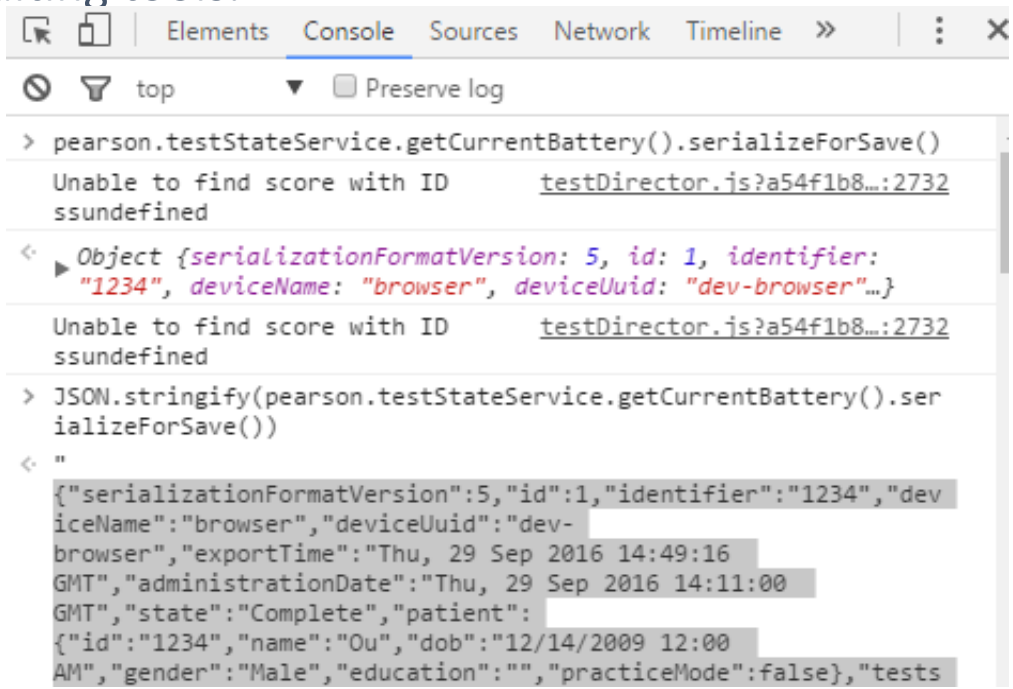


Project Rex simulator (cont.)

- **Step 10:** Multiple ways to save your results in a JSON file (option 1)

- JSON.stringify(pearson.testStateService.getCurrentBattery().serializeForSave())

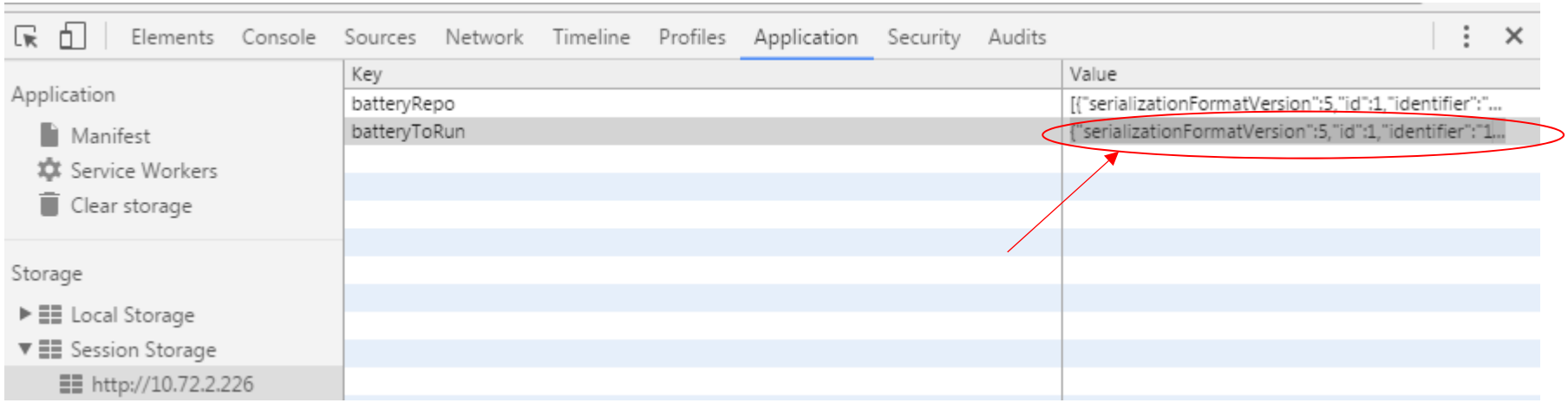
- The JSON format file is shown in the console. You can copy/paste to any of your JSON editing tools.



Project Rex simulator (cont.)

- **Step 10:** Multiple ways to save your results in JSON file (option 2)

  - Go to Application\Session Storage\ BatteryToRun\ Value, then Copy/Paste files to your JSON editing tool.



A separate document has been created with more details (K:\flea market\q-i training).

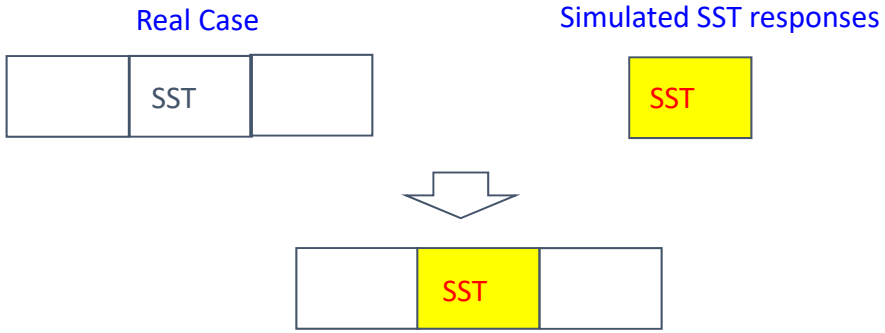
Project Rex simulator (cont.)

- **Summary**

- Project Rex can help you create cases with specific conditions.
- Pausing the stopwatch helps you complete the entire timed subtest.
- The simulated cases (JSON output files) can be easily used/modified as system cases.

- Example: WISC5 Spanish-Symbol Search Touch (SST) 1-75 items (form B)

  - The maximum item number in the real sample data is 53. So, the system variables from item 54 to item 75 are missing. The fully-administered simulated cases are used to replace all the responses for the SST.



Using JSON output to verify the variable list

- Original .json File in Notepad (WISC-V Spanish Example)

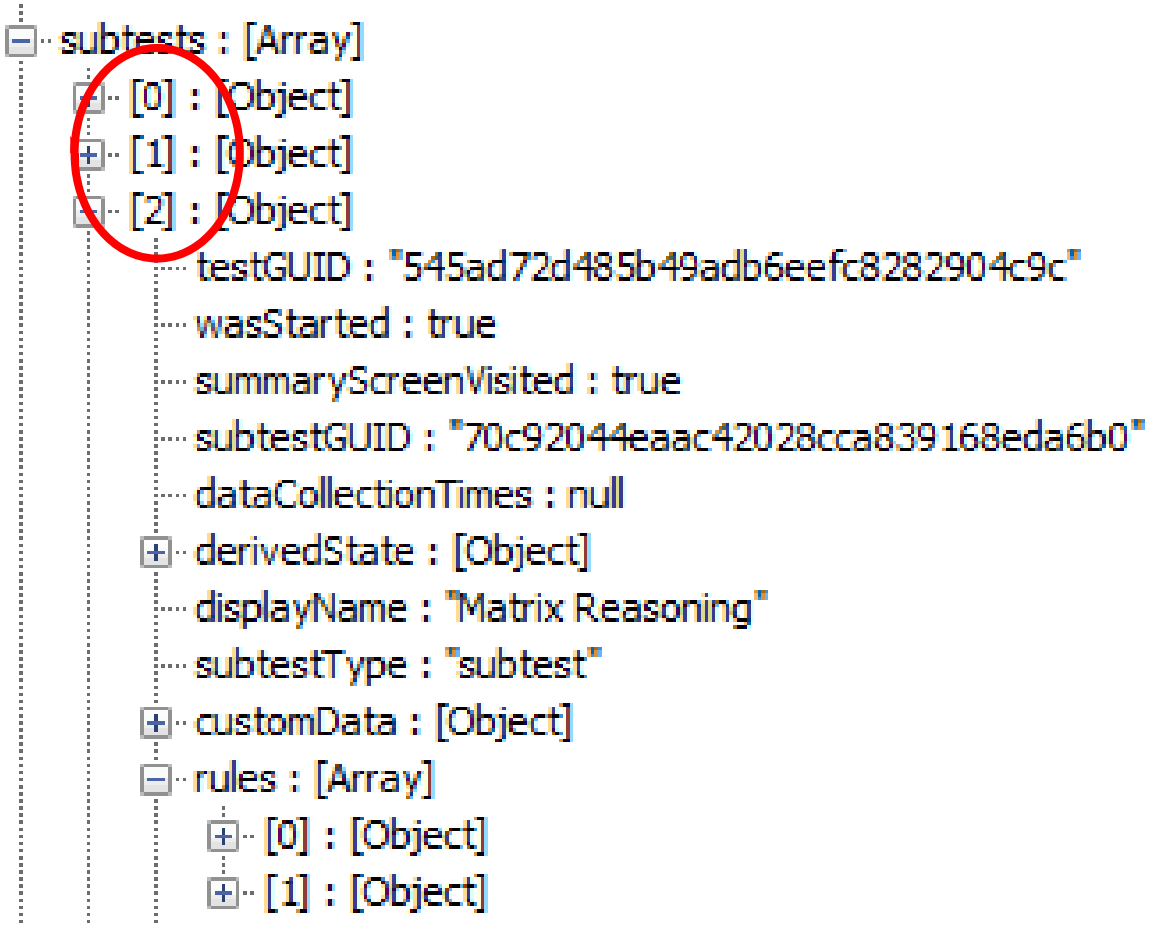

```
{    "identifier": "*****",    "deviceUuid": "*****",    "hasObservations": false,    "notes": [],    "examiners": ["2FBEA7F0518635C5E0532E0F190AF277"],
"administrationDate": "Fri, 15 Apr 2016 19:41:00 GMT",    "grades": [],    "serializationFormatVersion": 5,    "subtests": [        {            "testGUID":
"545ad72d485b49adb6eefc8282904c9c",            "wasStarted": true,            "summaryScreenVisited": true,            "subtestGUID":
"3ca6f1f8ec884525b36aefe1dd5b97ef",            "dataCollectionTimes": null,            "derivedState": {                "scaledScore": null,
"customScoringFields": [],            "graphConfig": {                "scoreLabel": "give.sharedSummarySheets.scaledScoreLabel",
"min": 1,                "max": 19,                "graphClass": "scaledScoreGraph",                "interval": 1            },
 "totalRawScore": 22,                "state": "Complete",                "customScoringType": "wiscv",                "totalTodos": 0,
"todoItemIds": []            },            "displayName": "Block Design",            "subtestType": "subtest",            "customData": {
"rawScoreData": {                "bd_pRank": "63",                "bd_scaled": 11,                "bd_raw": 22,                "bd_ae": "8:6'
            },                "rotationErrors": 0,                "dimensionErrors": 1            },            "rules": [                null,
    {                "mode": "start",                "skipToQuestionId": "wisc-v-esp-blockdesign-item3",                "positionToResumeTo":
null,                "itemTriggeredResumeId": null,                "gradeStartingPointItemId": null            },                null
],            "subtestId": 916,            "title": "Block Design",            "discontinueUsed": true,            "subtestInstanceID": "06BB069A-3096-48A2-
8C79-A1516950D755",            "completionTime": 399514,            "reverseUsed": false,            "questionsInOrderAdministered": [
"QWT:wisc-v-esp-blockdesign-item1",            "QWT:wisc-v-esp-blockdesign-item2",            "QWT:wisc-v-esp-blockdesign-item3",
"ADM:wisc-v-esp-blockdesign-item4:SID:06BB069A-3096-48A2-8C79-A1516950D755",            "ADM:wisc-v-esp-blockdesign-item5:SID:06BB069A-3096-48A2-8C79-
A1516950D755",            "ADM:wisc-v-esp-blockdesign-item6:SID:06BB069A-3096-48A2-8C79-A1516950D755",            "ADM:wisc-v-esp-blockdesign-
item7:SID:06BB069A-3096-48A2-8C79-A1516950D755",            "ADM:wisc-v-esp-blockdesign-item8:SID:06BB069A-3096-48A2-8C79-A1516950D755",
"ADM:wisc-v-esp-blockdesign-item9:SID:06BB069A-3096-48A2-8C79-A1516950D755"            ],            "normType": "wisc-v",            "startTime": "Tue, 10
May 2016 17:05:52 GMT",            "abbr": "bd",            "itemGroups": [            {                "shouldSuppressScore": true,
  "derivedState": {"totalRawScore": 10},            "questions": [                {                "trials": [
          {                "maxTime": 30,                "isItemSetReverseStopPoint": false,
          "derivedState": {                "shouldShowGrayedScore": false,
"shouldShowDiscontinue": false,                "scoreDisplay": {                "score": 2,
                "displayType": "showScore"            },
"ceButtonShouldNotCountTowardsSubtestTotal": false,            "isTodo": false,
"simpleSkip": false        },            "isStartingPoint": true,
"wasSkippedForStartingPoint": false,            "description": null,            "type": "blockDesign",
                "title": "Item 1 Trial 1",            "contextualEventButtons": {},
"completionTime": 1,            "predefinedBehavior": {            "data": {
          "is2x2WithTopRowOffset": false,            "width": 2,
"expectedState": {            "blocks": [            5,
                0            ],            "rotation": 0
              },            "isResponseCorrect": false,            "strategyState":
{}            },            "type": "blockDesign"            },
            "rawScore": 2,            "wasAdministered": true,            "reversedState": false,
                "startTime": "2016-05-10T17:05:52.927Z",            "id": "wisc-v-esp-blockdesign-item1-546",
```

Using JSON output to verify the variable list (cont.)

A Better W cally View .json Files

Notepad++

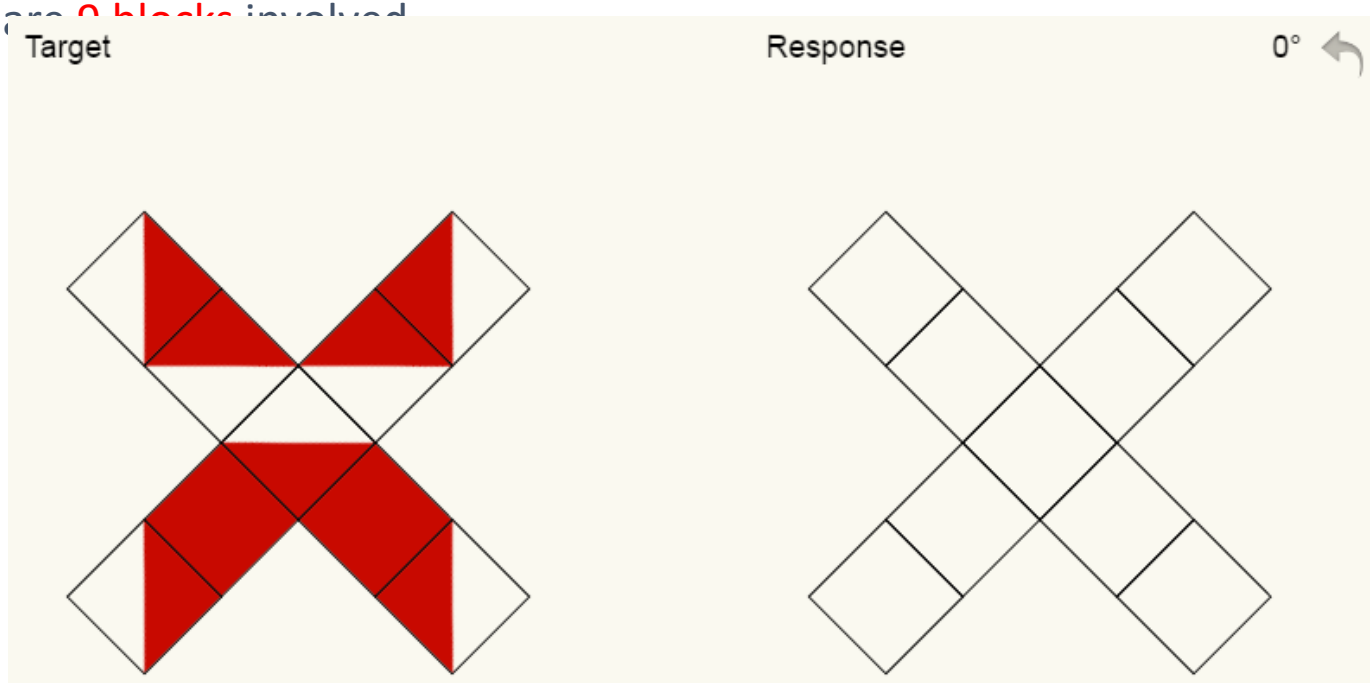
- Open Your .json File
- Plugins
  - Plugin Manager
    - Show Plugin Manager
      - Available Tab
      - Check JSTool
- Plugins
  - JSTool
    - Json Viewer



Using JSON output to verify the variable list (cont.)

Example: WISC-V Spanish Block Design Item 13

Psychometrically, we assume there might be 9 variables related to this item, because there are 9 blocks involved.



Using JSON output to verify the variable list (cont.)

The fact is, there are 25 JSON variables related to this item. So, we need to create 25 variables to meet psychometric-system variable one-on-one connection requirement.

Using JSON output to verify the variable list (cont.)

Rules To Follow

1. The Psychometric variable list must reflect the feature of the system variables from digital administration software so that the psychometric-system variable one-on-one connection requirement is met

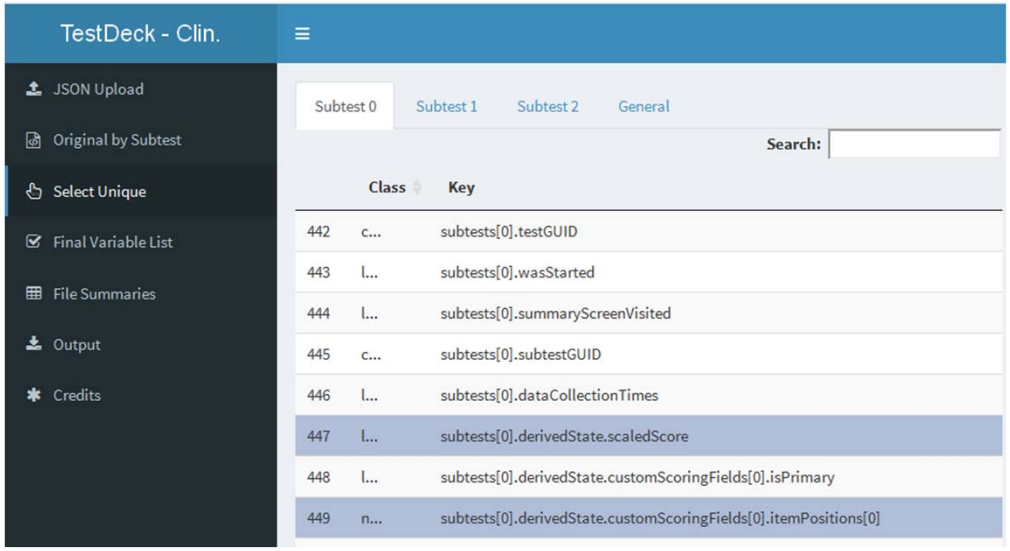
2. In the future, what value/variable type you see in the Q-i output file, will be directly mapped to the psychometric variable.

3. Psychometrics will no longer ask Field Research team to convert value/variable type/variable pattern for us.

Ben created a separate slide for this part with more details (K:\flea market\q-i training).

Functions under development

1. Variable list/map mapping tool (Ben & Ou)



2. CIT-Master specs procedure exploration (Ou)
3. Universal sas macro for parsed json/sas data conversion (Wendy & Ou)
4. JSON/Aurora data comparison code/tool (Ben & Ou)
5. System case simulation, converting input format file for digital platform application (Suping, Ou, Ben)

Acknowledgement

- Thanks to Kristen Getz for all her help, assistance, and information!