



Jama Connect® Intermediate Training Exercises

Table of Contents

Overview	2
Required Homework Materials:	2
Set Up Test Data in Jama Connect Instructions	2
Mockup Template Instructions	2
Intermediate Training Homework Prompts	3
Title Page Exercises	3
Exercise 1 – Title Page Structure.....	3
Exercise 2 – Title Project Content	3
Exercise 3 – Title Content	3
Exercise 4 – Subtitle Content	3
Exercise 5 – Report Run Date	3
Exercise 6 – Report Run Author	3
Page Layout Switch.....	4
Exercise 1 – Apply Page Layout Switch.....	4
Exercise 2 – Generate Headers and Footers	4
Exercise 3 – Format Headers and Footers	4
Content Page(s) Exercises.....	4
Exercise 1 – Print Item Fields	4
Exercise 2 – Call Item Macros	4
Exercise 3 – Define Item Macros	Error! Bookmark not defined.
Exercise 4 – Item Relationship Summary Macro	5
Exercise 5 – Item Version Comments Summary Macro	5
Exercise 6 – Item Version Baseline / Review Summary Macro	5
General Tips.....	5

Overview

The purpose of these exercises is to gain broader knowledge of utilizing the Velocity API and skills for developing custom reports. This document will provide you with a collection of exercise prompts that correspond to topics covered in the Intermediate Training material. Below you will find a list of the required homework materials that will be utilized to work through the exercises.

Required Homework Materials:

1. Sublime Text Editor & provided Velocity library package or a text editor of your choice.
2. Intermediate_Training_Template.vm
3. Intermediate_Training_Mockup_Example.docx
4. Intermediate_Training_Homework_Output_Example.docx
5. Jama Connect® instance
 - a. Note that to install your custom report under Admin > Organization > Reports to your Jama Connect® instance you will need to either user Admin permissions or a user with said permissions.

Exercises Test Data Instructions

For these exercises you are welcome to utilize instance data you already have, or you can create your own data in a sandbox project.

1. In a Jama Connect project tree ensure you have an item(s) with:
 - a. At least 2 levels of downstream related items.
 - i. [Creating Items in Jama Connect](#)
 - b. Multiple item versions.
 - i. [Item Versions in Jama Connect](#)
 - c. Item associated to Baseline/s and Review/s
 - i. [Create a Baseline Manually in Jama Connect](#) / [Reviews in Jama Connect](#)

Mockup Template Instructions

To begin that process please follow the defined instructions below.

1. Open the **Intermediate_Training_Mockup_Example.docx** file in Microsoft Word and perform a 'Save As' function setting the file format to .mht (Single File Webpage) to open on your preferred text editor.
 - a. If using Sublime text ensure your language option is set to 'Velocity (HTML)'
2. To adjust the .mht file for readability perform a 'Find & Replace' function action with your editor's Regex option active using the patterns learned in the Basic Training:
 - a. '=3D' is replaced with '='
 - b. '=\\n' is replace with nothing

Intermediate Training Homework Prompts

You will modify the provided ***Intermediate_Training_Velocity_Template.vm*** file for these code exercises. We recommend installing and running the report after each exercise to verify you changes step by step. Additionally, you can use the ***Intermediate_Training_Exercise_Output_Example.docx*** for output comparison but note that it does not need to be an exact match.

If you get stuck on any exercise, feel free to move to the following exercise as we will be reviewing the code solutions in following training video.

Title Page Exercises

For these exercises you will be working in the title page section denoted in your homework file with the label “**## <ENTER TITLE PAGE INFORMATION HERE>**”.

Exercise 1 – Title Page Structure

Extract title page definition from .mht mockup file and place it into the proper section in your Velocity homework file denoted by the “**## <ENTER TITLE PAGE INFORMATION HERE>**” placeholder.

Hint: it is recommended to clean away any excess or non-essential HTML / CSS content generated by Word to reduce code bulk and maintain code cleanliness for your Velocity report.

Exercise 2 – Current Project

Extract the current project name value and apply it to the `<project name>` placeholder.

Exercise 3 – Title Content

Populate the `<Export Location Name>` placeholder with the name of the report's export location.

Hint: review the `#getContainer` macro which is automatically called on report run.

Exercise 4 – Export Context

Populate the `<contextType>` placeholder with the export context this report was generated from. Ex: Project Tree (parent), Baseline, Test Plan, etc...

Hint: review the globally supplied variables for any report on the [Velocity API Overview page](#).

Exercise 5 – Report Run Date

Populate the `<report run date>` placeholder showing when the report was generated.

Hint: The `#getDate` macro can be utilized here alongside the arguments `$dateTool.date` and `$timeZone`

Exercise 6 – Report Run Author

Populate the `<report runner full name>` placeholder showing the report runner's full name.

Page Layout Switch

For these exercises review the **@page** definition sections that define your page containers before navigating to the “<ENTER PAGE LAYOUT SWITCH HERE>” placeholder.

Exercise 1 – Apply Page Layout Switch

Provide a page orientation layout switch to Landscape orientation following your Title Page content.

Hint: Ensure you follow the steps outlined in the Intermediate slides Part 1 Switching Page Layout

Exercise 2 – Re-Apply Headers and Footers

Following your page orientation change call the headers / footers macro `#responsiveHeadersFooter` to ensure those are applied to your new page orientation definition.

Exercise 3 – Define Headers and Footers

Populate your page header and footer definitions in the `#responsiveHeadersFooter` macro utilizing the provided mockup template.

Hint: convert your mockup template to a .mht to view the source HTML definitions to utilize in your homework exercise.

Content Page(s) Exercises

For these exercises first navigate to “<ENTER ITEM FORMATTING HERE>” placeholder for exercise 1 then the “<CUSTOM MACROS: INSERT/BUILD CUSTOM MACROS HERE>” placeholder for exercises 2 - 6.

Exercise 1 – Print Item Fields

Within your `ITEM TYPE MAP` section for any exported item that is not of type CMP, SET, FLD, or TXT output the item documentKey, name, and description fields as shown in the mockup example.

Hint: be aware that the description field is rich text and will need to be processed using the `#formatRichText()` macro.

Exercise 2 – Call Item Macros & Extract Item Versions

Following the description field output, implement logic to extract all item versions sorted by their created date. Then create three macro calls passing in the item `$doc` variable and your sorted item `$versions` variable in the following order.

- `itemVersionSummary`
- `itemVersionBaselineReviewSummary`
- `itemRelationshipSummary`

Using your mockup .mht file extract the general structure of each table to define the initial model of each macro implementation.

Exercise 3 – Item Version Summary Macro

Create a table definition with dynamic row creation for each item version gathered. Each gathered version will build a dynamic row showing the corresponding version number, active status, created date, and version comments.

Hint: review the #getDate() macro for date formatting utilization.

Exercise 4 – Item Version Baseline / Review Summary Macro

Utilizing the item's version list, gather all Baselines and Reviews associated to each item version. Each Baseline and Review pair will be presented as a dynamic table row showing the corresponding version number, baseline name, review name, review revision version, and review revision status.

If no baselines are found for a given version provide just the version number and a column that spans across the remaining columns populated with the following static text of "No Baselines Found".

Hint: utilize the colspan attribute to span cells across multiple columns.

Hint: recall that when using the Velocity API you want to identify the proper paths to reach the data you need. Begin first at the Version interface and then inspect the BaselineSource interface to find a method to get an associated Revision and from that the associated Review.

Exercise 5 – Item Relationship Summary Macro

Implement logic to extract both level one and level two downstream related items which will be presented as a merged cell format table row as shown in the mockup example.

If there are no trace items at either first or second downstream level, populate the cell with static text "N/A".

Hint: utilize the rowspan attribute to span cells across multiple rows which is how you will create the merged cell formatting for a trace table.

General Tips

- Remember to utilize the mockup .mht file for the HTML and CSS styles. Run the report first without updating the styles, and in the output see what elements you need to target for style updates. Then update those style classes or direct HTML tag elements with inline styles in the .vm file.
- If you want the tables you generate in each of the custom macros to have their header row repeat across pages, wrap the header row in <thead></thead> tags.
- This mockup does not have a Table of Contents, so for the <h1></h1> and <h2></h2> tags in the report you can give them an inline style of mso-list:none; to prevent automatic heading index numbers.