

# openSAP Evolved Web Apps with SAPUI5

## Week 3 Unit 1: Kick-Start Apps with Templates

### Exercises

PUBLIC



# TABLE OF CONTENTS

WEEK INTRODUCTION..... 3

KICK-START APPS WITH TEMPLATES ..... 4

  Create an App from the “Master-Detail” Template ..... 5

  Customize Application Texts ..... 9

★ CHALLENGE YOURSELF: BRAND THE MASTER PAGE .....11

RELATED MATERIAL.....12

## WEEK INTRODUCTION

### Summary

In this week's units we will start to develop a more complex application that allows us to explore some advanced topics around UI5.

We will get acquainted with

- An application project generated from SAP Web IDE
- OData and its internals
- SAP Web IDE's layout editor
- Advanced control topics
- Code reuse pattern in UI5

### Preview

Our scenario for this and the upcoming week is to create an order browser for a global player in the transport industry: **UI5 Air cargo**. The app will ease order processing and can finally replace an old legacy application that causes some major hick-ups in the delivery department. They requested to develop an SAP Fiori app with SAPUI5 which helps their employees getting the work done.

The screenshot displays the 'UI5 Air Cargo' application interface. The left sidebar shows a list of orders with details like company name, amount, date, and location. The main area shows the 'Order Preparations' section for 'Danish Fish Trading Company', including a table of 'Order Items (5)' with columns for Quantity, ProductId, Delivery date, Delivery urgency, and Price. The right sidebar shows the 'DETAILS' section for the selected order, including item description, language, and delivery date.

**UI5 Air Cargo**  
You name it, we ship it!

Orders (52)

Search

Siwusha 438.00 USD  
3/7/19  
0100000042 CN: Shanghai

Danish Fish Trading Comp any 4911.00 USD  
3/6/19  
0100000043 DK: Copenhagen

Sorali 95508.50 USD  
3/6/19  
0100000044 DE: Karlsruhe

South American IT Company 231.00 USD  
3/3/19  
0100000041 AR: Cordoba

Siwusha 454.88 USD  
3/3/19  
0100000042 CN: Shanghai

**Danish Fish Trading Company**

Delivery status: **NEW**

Order Preparations

1. Pack Items ☐ 2. Print Invoice ☐ 3. Add Sample ☐

Order Items (5)

Quantity	ProductId	Delivery date	Delivery urgency	Price
3	HT-1070 0000000010	Mar 13, 2019		212.70 \$
1	HT-1071 0000000020	Mar 13, 2019		81.70 \$
3	HT-1072 0000000030	Mar 13, 2019		303.60 \$
3	HT-1137 0000000040	Mar 13, 2019		4290.00 \$
1	HT-1138 0000000050	Mar 13, 2019		23.00 \$

0500000494 / 0000000040 / HT-1137  
Danish Fish Trading Company

Item description  
EPM DG: SO ID 0500000494 Deliver as fast as possible

Language  
EN

DETAILS

Gross Price: 5105.10 USD  
Net Price: 4290.00 USD  
Tax: 815.10 USD  
Quantity: 3 EA

DELIVERY

Delivery date: March 13, 2019

Accept Reject Cancel

*The final app at the end of the week*

## KICK-START APPS WITH TEMPLATES

### Summary

The SAP Fiori Master Detail template in SAP Web IDE allows us to display two SAP Fiori floorplans in one layout. On the master side, a list of sales orders can be displayed, while on the detail side an object page to show sales orders is possible. The `sap.f.FlexibleColumnLayout` gives the user the ability to resize each column.

### Preview

Orders (502)

Search

AVANTEL

85125.40

USD

AVANTEL

724.98

USD

AVANTEL

345.80

USD

AVANTEL

1431.97

USD

AVANTEL

454.88

USD

AVANTEL

438.00

USD

AVANTEL

85125.40

USD

AVANTEL

724.98

USD

AVANTEL

345.80

USD

AVANTEL

Price:

85125.40 USD

Products (7)

Name

Price

HT-1500

5000.00 USD

0000000010

HT-1501

30000.00 USD

0000000020

HT-1502

50000.00 USD

0000000030

HT-1110

26.70 USD

0000000040

HT-1111

13.80 USD

0000000050

HT-1112

78.00 USD

0000000060

HT-1113

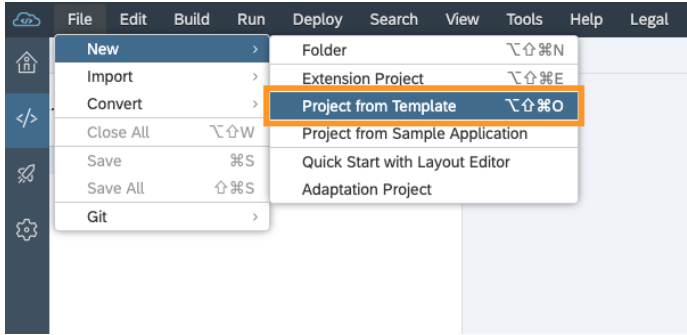
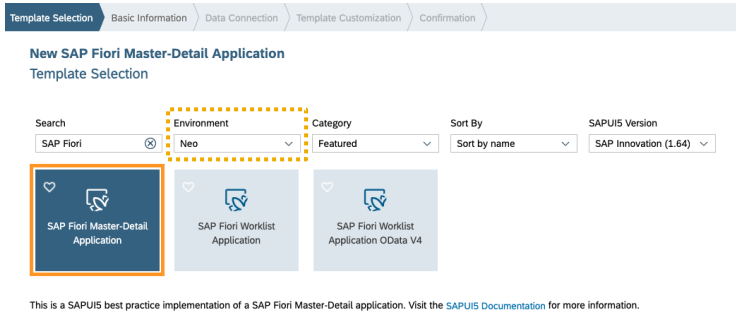


6.90 USD

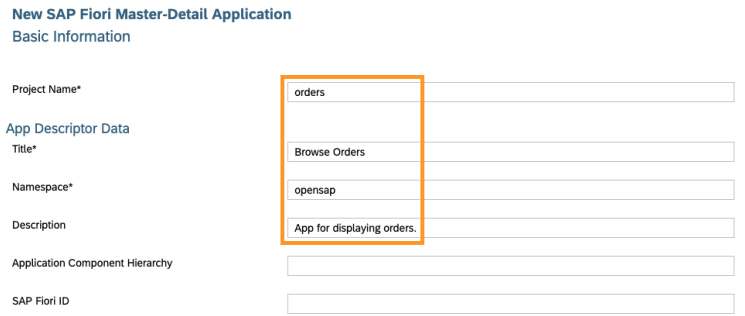

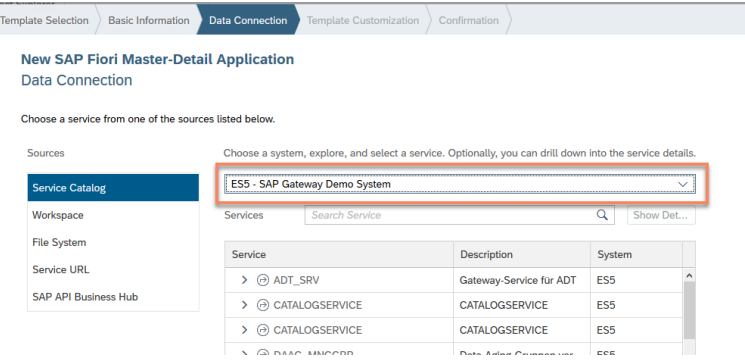
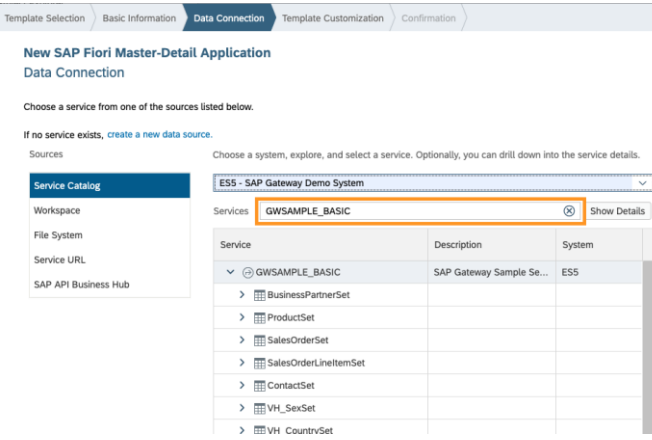

0000000070

*Master-Detail application showing sales orders and the corresponding line items*

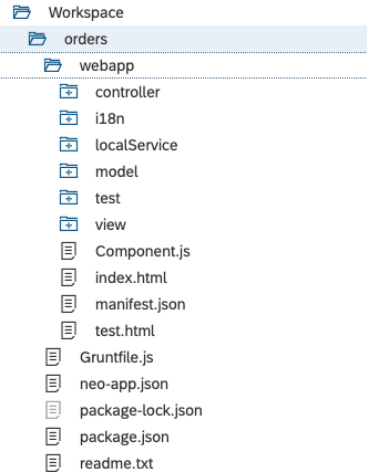
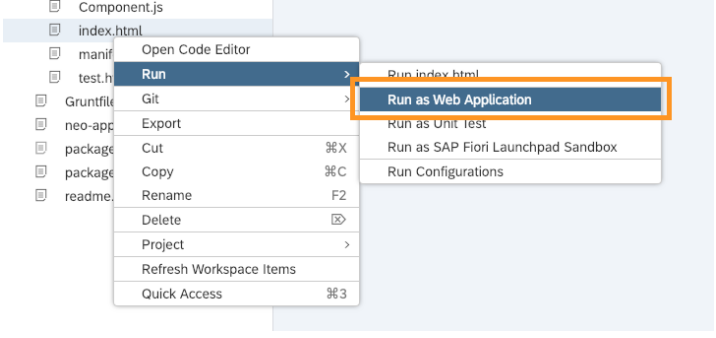
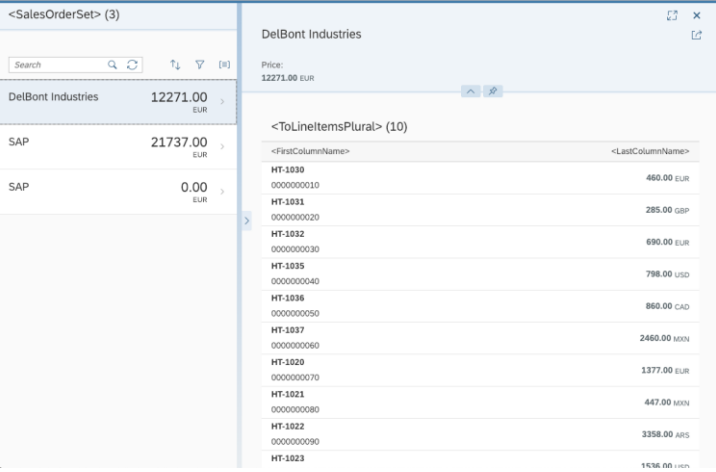
## Create an app from the “Master-Detail” Template

The first step we take in developing our new application is to generate the application project from the Master-Detail template in SAP Web IDE. During generation, we will also connect to an OData service. Based on the service, we will map the available data to our template, so that it will be displayed in the right places.

Explanation	Screenshot
<p>1. Choose <i>File</i> → <i>New</i> → <i>Project from Template</i> from the menu to open the wizard.</p>	
<p>2. In the <i>Template Selection</i> step, search for <b>SAP Fiori</b>, and select <i>SAP Fiori Master-Detail Application</i> to continue the wizard with this template.</p> <p><b>Important:</b> Ensure that value <i>Neo</i> is selected for environment otherwise your project folder departs from this exercise and you might have issues running the app as your ES5 destination is only configured in your NEOtrial account.</p> <p><b>Note:</b> If you cannot find this template, select <i>All</i> in the <i>Category</i> dropdown menu.</p>	
<p>3. Select the latest available SAPUI5 version, for example <i>SAP Innovation (1.64)</i>.</p>	
<p>4. Choose <i>Next</i> to go to the next wizard step.</p>	

Explanation	Screenshot
<p><i>Basic Information</i> step:</p> <ol style="list-style-type: none"> <li>Enter <b>orders</b> in the <i>Project Name</i> field.</li> <li>Enter <b>Browse Orders</b> as the title.</li> <li>Enter the namespace <b>opensap</b></li> <li>Enter the description <b>App for displaying orders.</b></li> </ol>	
<ol style="list-style-type: none"> <li>Choose <i>Next</i> to go to the next wizard step.</li> </ol>	
<p><i>Data Connection</i> step</p> <ol style="list-style-type: none"> <li>In the <i>Sources</i> list, select <i>Service Catalog</i>.</li> <li>Select <i>ES5 – SAP Gateway Demo System</i> in the corresponding field. A list below the input field shows all available services on the ES5 system.</li> </ol>	
<ol style="list-style-type: none"> <li>Enter the search term <b>GWSAMPLE_BASIC</b>, and select the service in the list below.</li> </ol>	
<ol style="list-style-type: none"> <li>Choose <i>Next</i> to go to the next wizard step.</li> </ol>	

Explanation	Screenshot
<p>Template Customization step</p> <p>14. Enter the following values:</p> <p><i>App Type:</i> <b>Standalone App</b></p> <p><i>Object Collection:</i> <b>SalesOrderSet</b></p> <p><i>Object Collection ID:</i> <b>SalesOrderID</b></p> <p><i>Object Title:</i> <b>CustomerName</b></p> <p><i>Object Numeric Attribute:</i> <b>NetAmount</b></p> <p><i>Object Unit of Measure:</i> <b>CurrencyCode</b></p> <p><i>Line Item Collection:</i> <b>ToLineItems</b></p> <p><i>Line Item Collection ID:</i> <b>ItemPosition</b></p> <p><i>Line Item Title:</i> <b>ProductID</b></p> <p><i>Line Item Numeric Attribute:</i> <b>NetAmount</b></p> <p><i>Line Item Unit of Measure:</i> <b>CurrencyCode</b></p>	<p><b>New SAP Fiori Master-Detail Application</b> Template Customization</p> <p><b>Info:</b> More details and background information about this template and its features can be found in our official <a href="#">SAPUI5 documentation</a></p>
<p>15. Choose <i>Finish</i> to end the wizard and generate the app.</p>	

Explanation	Screenshot
<p>16. Check that a new folder <code>orders</code> was added to your local workspace. It should contain the files and folders of the initial app as displayed in the screenshot on the right.</p>	
<p>17. Run the app: Right-click on the <code>webapp/index.html</code> file and choose <code>Run</code> → <code>Run as</code> → <code>Web Application</code> to run the app.</p>	
<p>18. A preview of the app opens in a separate window for testing purposes.</p> <p>This is the initial version provided by the template. You can click on an order to see the line items of the order on the detail page.</p>	



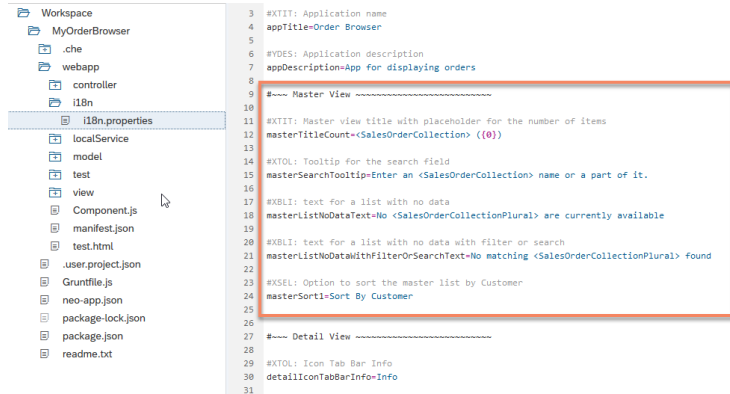
## Customize Application Texts

A typical activity after generating a template is adjusting it to your scenario. Before we add new features to our application, we adjust the generated texts based on the service entity that we have picked in the wizard.

1. Open the `webapp/i18n/i18n.properties` file and analyze the file.

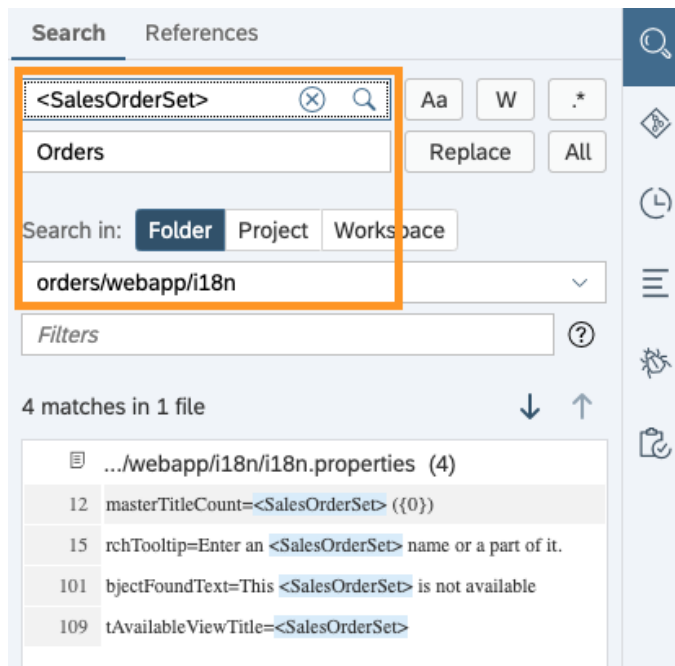
### Note:

The `<SalesOrderSet>` placeholder that you can find in several places is the technical name of the service entity and has been generated by the template. We will adjust it to our application scenario.



```
3 #XTIT: Application name
4 appTitle=Order Browser
5
6 #YDES: Application description
7 appDescription=App for displaying orders
8
9
10
11 #--- Master View ---
12 #XTIT: Master view title with placeholder for the number of items
13 masterTitleCount=<SalesOrderCollection> {{0}}
14
15 #XTOL: Tooltip for the search field
16 masterSearchTooltip=Enter an <SalesOrderCollection> name or a part of it.
17
18 #XBLL: text for a list with no data
19 masterListNoDataText=No <SalesOrderCollectionPlural> are currently available
20
21 #XBLL: text for a list with no data with filter or search
22 masterListNoDataWithFilterOrSearchText=No matching <SalesOrderCollectionPlural> found
23
24 #XSEL: Option to sort the master list by Customer
25 masterSort1=Sort By Customer
26
27
28 #--- Detail View ---
29 #XTOL: Icon Tab Bar Info
30 detailIconTabBarInfo=Info
31
```

2. Open the search panel in SAP Web IDE by clicking on the search icon in the right panel.
3. Enter `<SalesOrderSet>` in the *Find* field.
4. Enter **Orders** in the *Replace* field
5. In the *Search in...* section select the *Folder* option. Make sure that the `orders/webapp/i18n` folder is selected.
6. Click on the magnifying glass and make sure that only occurrences in the translation files are found.
7. Then choose *All* to update the translation file. In the confirm dialog select *Replace All*.



**Note:** If the search and replace feature does not find any occurrences of the search string, leave out the `<>` from the search string, and delete the occurrences from the file manually.

8. Repeat the text replacement for the following fields to improve readability of all captions.

**<SalesOrderSetPlural> with Orders**

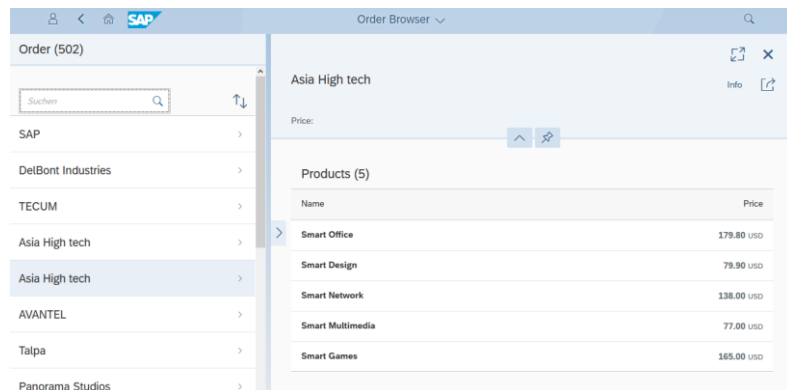
**<ToLineItemsPlural> with Products**

**<FirstColumnName> with Name**

**<LastColumnName> with Price**

**<CustomerName> with Customer**

**Note:** Other placeholder texts can be left as they are. We will replace them with other content in the next step.



## webapp/i18n/i18n.properties

```
#XTOL: Tooltip for the search field
worklistSearchTooltip=Enter an order name or a part of it.
...
#XBLI: text for a list with no data
masterListNoDataText=No orders are currently available
...
#XBLI: text for a table with no data with filter or search
masterListNoDataWithFilterOrSearchText=No matching orders found
...
#YMSG: The Products not found text is displayed when there is no product with
this id
noObjectFoundText=This order is not available
```

Fine tune some other generated texts to match our application scenario.

Now run the app again by right-clicking on the `webapp/index.html` file and select *Run* → *Run as* → *Web Application*. This time you should see the app without placeholders as well as the real service data as shown in the preview.

## ★ CHALLENGE YOURSELF: BRAND THE MASTER PAGE

**Note:** This task does not have a predefined solution and can be solved creatively – dive a bit deeper into the topics and exchange with other learners to make the most out of your learning experience. Good luck!

### Summary

Our client UI5 Air Cargo just saw your first explorations and asked you to include their company name and slogan on the master page of the app in order to raise brand awareness. Can you brand the header area according to the client's needs?

**Logo:** A phoenix

**Title:** UI5 Air Cargo

**Claim:** You name it, we ship it!

### Preview

UI5 Air Cargo

You name it, we ship it!

Orders (502)

Search

Bionic Research Lab

0.00

USD

Bionic Research Lab

0.00

USD

PicoBit

345.80

USD

Siwusha

438.00

USD

Bionic Research Lab

4917.00

USD

PicoBit

Price:

345.80 USD

Order Items (4)

ProductId	Price
HT-1104 0000000010	165.00 USD
HT-1105 0000000020	87.00 USD
HT-1106 0000000030	34.00 USD
HT-1107 0000000040	59.80 USD

Figure 1 - The master page should be branded for UI5 Air Cargo

### Hints:

- `sap.f.Avatar` seems like an interesting control for displaying small images or icons
- `sap.f.SemanticPage` offers a variety of aggregations to customize the page appearance
- Layout controls can be nested to achieve any kind of design pattern
- The [icon explorer](#) lets you browse the SAPUI5 icon font for any subject

## RELATED MATERIAL

- [Demo Kit: Get a Trial Account and Access SAP Web IDE](#)
- [SAP Fiori Design Guidelines](#)
- [Demo Kit: Flexible Column Layout](#)
- [Demo Kit: Template Demo Apps](#)

## Coding Samples

Any software coding or code lines/strings ("Code") provided in this documentation are only examples and are not intended for use in a production system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules for certain SAP coding. SAP does not warrant the correctness or completeness of the Code provided herein and SAP shall not be liable for errors or damages caused by use of the Code, except where such damages were caused by SAP with intent or with gross negligence.

**[www.sap.com/contactsap](http://www.sap.com/contactsap)**

© 2018 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company.

The information contained herein may be changed without prior notice. Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

In particular, SAP SE or its affiliated companies have no obligation to pursue any course of business outlined in this document or any related presentation, or to develop or release any functionality mentioned therein. This document, or any related presentation, and SAP SE's or its affiliated companies' strategy and possible future developments, products, and/or platform directions and functionality are all subject to change and may be changed by SAP SE or its affiliated companies at any time for any reason without notice. The information in this document is not a commitment, promise, or legal obligation to deliver any material, code, or functionality. All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, and they should not be relied upon in making purchasing decisions.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies. See <http://www.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.