



Бесплатная электронная книга

УЧУСЬ

sapui5

Free unaffiliated eBook created from
Stack Overflow contributors.

#sapui5

| | |
|------------------------|-----------|
| | 1 |
| 1: sapui5 | 2 |
| | 2 |
| Examples..... | 3 |
| , !..... | 3 |
| , | 3 |
| , | 4 |
| 2: | 5 |
| Examples..... | 5 |
| OData Viz-Chart..... | 5 |
| 3: | 6 |
| | 6 |
| Examples..... | 6 |
| xmlview..... | 6 |
| | 7 |
| | 8 |
| 4: sapui5 | 10 |
| | 10 |
| | 10 |
| Examples..... | 10 |
| sapui5 | 10 |
| SAPUI5..... | 13 |
| | 15 |

Около

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [sapui5](#)

It is an unofficial and free sapui5 ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official sapui5.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

глава 1: Начало работы с sapui5

замечания

Что такое SAPUI5?

Основываясь на вышеприведенной теории, SAP представила инструментарий для разработки на базе HTML5, SAPUI5, который поощряет один постоянный пользовательский интерфейс. Используя вышеприведенную теорию, приложения, созданные с использованием SAPUI5, реагируют на браузеры и устройства - они работают на смартфонах, планшетах и настольных компьютерах. Элементы пользовательского интерфейса автоматически адаптируются к возможностям каждого устройства. Для этого SAPUI5 обеспечивает надежные концепции разработки для создания приложений с бизнес-приложениями на основе бизнес-приложений. Вкратце, UI5 - это клиентский интерфейс, основанный на JavaScript, CSS и HTML5. Серверы приходят в игру для развертывания ваших приложений, хранения библиотек SAPUI5 и подключения к базе данных. В зависимости от среды, в которой вы используете SAPUI5, библиотеки и ваши приложения хранятся, например, в SAP HANA Cloud Platform или другом сервере приложений. Предпочтительным средством доступа к бизнес-данным для вашего приложения является использование модели oData.

Инструментарий разработки SAP UI для HTML5 - это технология пользовательского интерфейса, которая используется для создания и адаптации клиентских приложений. Время выполнения - это клиентская библиотека рендеринга HTML5 с богатым набором стандартных и расширенных элементов управления и легкая модель программирования.

Существует два варианта: [OpenUI5](#) (версия Open Sourced) и SAPUI5 (исходная лицензионная версия). Оба имеют одинаковые технические основные механизмы и в совокупности называются UI5.

UI5 предоставляет множество функций, позволяющих вам легко создавать и расширять современные пользовательские интерфейсы:

- Он поддерживает RIA как клиентские функции на основе JavaScript
- Он поддерживает CSS3, который позволяет эффективно адаптировать темы к брендингу вашей компании
- Он основан на встроенных концепциях расширяемости на уровне кода и приложения
- Он использует библиотеку jQuery с открытым исходным кодом в качестве основы
- Он полностью поддерживает стандарты продуктов SAP
- Он соответствует OpenAjax и может использоваться вместе со стандартными библиотеками JavaScript
- Он предлагает множество гибких средств управления для создания независимых пользовательских интерфейсов платформы с меньшими усилиями

- Он предлагает полную поддержку Translation / i18n
- В нем есть язык дизайна Fiori , основанный на обширных исследованиях UX

Вы можете получить свою первую страницу UI5, начатую [здесь](#) . Также для получения дополнительной информации см. Руководство разработчика и ссылку на API, доступную в соответствующих ссылках SDK: [OpenUI5 SDK](#) , [SAPUI5 SDK](#) . Демо-приложения можно найти [здесь](#)

Examples

Привет, мир!

Начнем с создания HTML-страницы для приложения. Там мы определяем метатеги, тег сценария для загрузки библиотек SAPUI5 и место заполнителя для содержимого приложения.

```
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta charset="utf-8">
  <title>Hello World App</title>
  <script src="http://<<server>>:<<port>>/resources/sap-ui-core.js"
    id="sap-ui-bootstrap"
    data-sap-ui-theme="sap_bluecrystal"
    data-sap-ui-libs="sap.m">
  </script>
</head>
<body class="sapUiBody" id="content">
</body>
</html>
```

Адаптируйте путь, в котором расположены ресурсы (<>: <>) в соответствии с вашей установкой. Для OpenUI5 вы можете использовать src = "<https://openui5.hana.ondemand.com/resources/sap-ui-core.js>". Например, для доступа к SAPUI5 на облачной платформе SAP HANA используйте src = "<https://sapui5.hana.ondemand.com/resources/sap-ui-core.js>".

Привет, мир

```
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
  <title>SAPUI5 Hello World</title>
  <!-- Load SAPUI5 , theme and control library -->
  <script id="sap-ui-bootstrap"
    src="https://sapui5.hana.ondemand.com/resources/sap-ui-core.js"
    data-sap-ui-theme="sap_bluecrystal"
    data-sap-ui-libs="sap.m"></script>
```

```

<!-- Create a UI5 button and place it onto the page -->
<script>
    new sap.m.Button({
        text:"Hello world",
        press: function(){
            alert("hello SapUI5!");
        }
    }).placeAt("content");
</script>
</head>
<body class="sapUiBody" id="content">
</body>
</html>

```

Привет, мир

```

<!DOCTYPE HTML>
<html>
<head>
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta http-equiv='Content-Type' content='text/html; charset=UTF-8' />
    <script type="text/javascript" charset="utf-8" src="cordova.js"></script>
    <script src="resources/sap-ui-core.js"
        id="sap-ui-bootstrap"
        data-sap-ui-libs="sap.m"
        data-sap-ui-theme="sap_bluecrystal"
        data-sap-ui-xx-bindingSyntax="complex"
        data-sap-ui-compatVersion="1.24"
        data-sap-ui-resourceroots='{ "<projectname>": "./" }'>
    </script>
    <!-- only load the mobile lib "sap.m" and the "sap_bluecrystal" theme -->

    <script>
    sap.ui.getCore().attachInit( function () {
        new sap.ui.core.ComponentContainer ("<ComponentId(can be anyname you wish)>", {
            height : "100%",
            name : "<name of component>"
        }).placeAt ('content');
    });
    </script>

</head>
<body class="sapUiBody" role="application">
    <div id="content"></div>
</body>

```

Поместите код начальной загрузки в attachInit, потому что он будет запущен после загрузки основной библиотеки

Прочитайте Начало работы с sapui5 онлайн: <https://riptutorial.com/ru/sapui5/topic/970/начало-работы-с-sapui5>

глава 2: Графики

Examples

Применение фильтра в OData и Viz-Chart

Это пример Viz-Charts с линейной диаграммой с фильтрами. Существует много методов, которые позволяют решить проблему фильтрации.

Следует отметить, что вам необходимо привязать набор данных VizFrame по его идентификатору, а затем применить фильтрацию на FlattenedDataset

В контроллере:

```
// defining the Filter
var oFilter = new sap.ui.model.Filter("Data1", sap.ui.model.FilterOperator.GT, 10);

//Setting oModel
var oModel = new sap.ui.model.odata.ODataModel("/destinations/v4/abc/http/app.svc", oConfig);
this.getView().setModel(oModel);

//Binding the filtered data to the chart by calling it from its ID and binding the data there
this.getView().byId("idVizFrame").getDataset().getBinding("data").filter([oFilter]);
```

В представлении XML:

```
<viz:VizFrame id="idVizFrame" uiConfig="{applicationSet:'fiori'}" height='100%' width="100%"
vizType='line' >
  <viz:dataset>
    <viz.data:FlattenedDataset data="{YOUR_ENTITY_SET}">
      <viz.data:dimensions>
        <viz.data:DimensionDefinition name="TimeStamp" value="{TimeStamp}"/>
      </viz.data:dimensions>
      <viz.data:measures>
        <viz.data:MeasureDefinition name="SENSOR1" value="{SENSOR1}"/>
      </viz.data:measures>
    </viz.data:FlattenedDataset>
  </viz:dataset>
  <viz:feeds>
    <viz.feeds:FeedItem id='valueAxisFeed' uid="valueAxis" type="Measure"
values="Data_SENSOR1"/>
    <viz.feeds:FeedItem id='categoryAxisFeed' uid="categoryAxis" type="Dimension"
values="TimeStamp"/>
  </viz:feeds>
</viz:VizFrame>
```

Прочитайте Графики онлайн: <https://riptutorial.com/ru/sapui5/topic/7620/графики>

глава 3: Связывание агрегации

параметры

| параметр | подробность |
|-------------|---|
| дорожка | Путь объекта или списка объектов, которые будут включены в агрегацию. |
| завод | Функция, которая создаст элемент представления агрегации. |
| сортировщик | Объект, который представляет способ сортировки объектов агрегации. |

Examples

Связывание агрегирования с использованием шаблонов в xmlview

XmlView:

```
<mvc:View
  controllerName="sap.m.sample.ListCounter.List"
  xmlns:mvc="sap.ui.core.mvc"
  xmlns="sap.m">
  <List
    headerText="Products"
    items="{products>/Products}">
    <!-- Template of the list item -->
    <StandardListItem
      title="{Name}"
    />
  </List>
</mvc:View>
```

контроллер:

```
sap.ui.define([
  'jquery.sap.global',
  'sap/ui/core/mvc/Controller',
  'sap/ui/model/json/JSONModel'
], function(jQuery, Controller, JSONModel) {
  "use strict";

  var ListController = Controller.extend("sap.m.sample.ListCounter.List", {

    onInit : function (evt) {
      // Model
      var oModel = new JSONModel("/products.json");
      this.getView().setModel(oModel, "products");
    }
  });
});
```



```
});

return ListController;

});
```

products.json:

```
{
  Products : [
    {"Name": "Product 1"},
    {"Name": "Product 2"},
    {"Name": "Product 3"},
  ]
}
```

Связывание агрегирования с сортировочными и статическими фильтрами

```
<mvc:View
  controllerName="sap.m.sample.ListCounter.List"
  xmlns:mvc="sap.ui.core.mvc"
  xmlns="sap.m">
  <List
    headerText="Fruits"
    items="{path:'products>/Products', sorter:{path:'Name'}, filter:{path:'Type',
operator:'EQ',value1:'Fruit'}}">
    <!-- Template of the list item -->
    <StandardListItem
      title="{Name}"
    />
  </List>
  <List
    headerText="Food"
    items="{path:'products>/Products', sorter:{path:'Name'}, filter:{path:'Type',
operator:'EQ',value1:'Food'}}">
    <!-- Template of the list item -->
    <StandardListItem
      title="{Name}"
    />
  </List>
</mvc:View>
```

контроллер:

```
sap.ui.define([
  'jquery.sap.global',
  'sap/ui/core/mvc/Controller',
  'sap/ui/model/json/JSONModel'
], function(jQuery, Controller, JSONModel) {
  "use strict";

  var ListController = Controller.extend("sap.m.sample.ListCounter.List", {

    onInit : function (evt) {
```

```

        // Model
        var oModel = new JSONModel("/products.json");
        this.getView().setModel(oModel, "products");
    }
});

return ListController;
});

```

products.json:

```

{
  Products : [
    { "Name": "Banana", "Type": "Fruit" },
    { "Name": "Meat", "Type": "Food" },
    { "Name": "Apple", "Type": "Fruit" },
    { "Name": "Rice", "Type": "Food" },
  ]
}

```

Связывание агрегирования с заводской функцией

XmlView:

```

<mvc:View
  controllerName="sap.ui.demo.wt.controller.App"
  xmlns="sap.m"
  xmlns:mvc="sap.ui.core.mvc"
  displayBlock="true">
  <App>
    <pages>
      <Page content="{path:'Tiles'/>Tiles',factory:'.tileFactory'}">

      </Page>
    </pages>
  </App>
</mvc:View>

```

контроллер:

```

sap.ui.define([
  "sap/ui/core/mvc/Controller",
  "sap/ui/model/json/JSONModel"
], function (Controller, JSONModel) {
  "use strict";

  return Controller.extend("sap.ui.demo.wt.controller.App", {

    onInit: function() {
      var oModel = new JSONModel("./model/data.json");

      this.getView().setModel(oModel, "Tiles");
    },
    tileFactory: function(sId, oContext) {

```

```

var oUIControl = null;

var type = oContext.getProperty("type");

switch(type){
    case "STD":
        var title = oContext.getProperty("Title");
        oUIControl = new sap.m.StandardTile();

        oUIControl.setTitle(title);
        break;
    case "NEWS":
        var title = oContext.getProperty("Title");
        var newsContent = new sap.m.NewsContent({contentText:title});
        oUIControl = new sap.m.GenericTile();
        oUIControl.addTileContent(new sap.m.TileContent({content:newsContent}));
        break;
    case "IMG":
        var src = oContext.getProperty("src");
        var imgContent = new sap.m.ImageContent({src});
        oUIControl = new sap.m.GenericTile();
        oUIControl.addTileContent(new sap.m.TileContent({content:imgContent}));
        break;
}

return oUIControl;

}

});

});

```

data.json:

```

{
  "Tiles":[
    {
      "type": "STD",
      "Title": "Standard Tile"
    },
    {
      "type": "NEWS",
      "Title": "NEWS Tile"
    },
    {
      "type": "IMG",
      "src": "https://1.bp.blogspot.com/-2YLGmdxqXMk/V58ki-
s5DLI/AAAAAAAAANhs/jcSRMEeJN_89vXNdrieljDGFhF5X-yh4ACLcB/s1600/ui5.png"
    }
  ]
}

```

Прочитайте **Связывание агрегации онлайн**: <https://riptutorial.com/ru/sapui5/topic/7930/связывание-агрегации>

глава 4: Таблица sapui5

Синтаксис

1. var oTable = new selectAllVisibleRowsTable ({... // alternatively может использовать новую sap.ui.table.Table
2. sap.ui.table.Table.extend ('selectAllVisibleRowsTable', {..... // здесь имя таблицы находится в кавычках

замечания

Официальная документация для таблицы sapui5, которая предоставляет только спецификацию API. <https://sapui5.hana.ondemand.com/docs/api/symbols/sap.ui.table.Table.html>

Examples

Пример таблицы для sapui5 с примерами управления и обработки

```
//Create a layout
var tableLayout = new sap.ui.commons.layout.MatrixLayout({
    layoutFixed : false,
    columns : 2,
    width : "100%",
    height : "100%",
    widths : [ "20%", "80%" ]
}).addStyleClass('dsAvailLayout');

sap.ui.table.Table.extend('selectAllVisibleRowsTable', {
    renderer : function(oRm, oControl) {
        sap.ui.table.TableRenderer.render(oRm, oControl);
    },

    selectAllVisibleRowIndex: function(checkKey) {
        var model = this.getModel();
        var rowPath = this.getBindingInfo('rows').path;
        var rows = model.getProperty(rowPath);
        var start = this.getFirstVisibleRow();
        var end = Math.min(start + this.getVisibleRowCount(), rows.length);

        for (var i = 0; i < rows.length; i++) {
            var row = rows[i];
            row[checkKey] = (i >= start && i < end);
        }
        this.invalidate();
    },

    selectAll: function(checkKey) {
        var model = this.getModel();
        var rowPath = this.getBindingInfo('rows').path;
        var rows = model.getProperty(rowPath);
        var start = this.getFirstVisibleRow();
```

```

    var end = rows.length;

    for (var i = 0; i < rows.length; i++) {
        var row = rows[i];
        row[checkKey] = (i >= start && i < end);
    }
    this.invalidate();
},
handle: function(){
    try{
        var model = this.getModel();
        var rowPath = this.getBindingInfo('rows').path;
        var rows = model.getProperty(rowPath);
        var selectedIndices = [];
        for (var i = 0; i < rows.length; i++) {
            var row = rows[i];
            if(row['checked'] == true){
                selectedIndices.push(i);
            }
        }

        objStr = "";
        var suffix = "";
        for (var i = 0; i < selectedIndices.length; i++) {
            var idx = selectedIndices[i];
            var cxt = this.getContextByIndex(idx);
            var path = cxt.sPath;
            var obj = this.getModel().getProperty(path);
            objStr = objStr+suffix+JSON.stringify(obj);
            suffix = ",";
        }
    }catch(err){

    }
}
});

var oTable = new selectAllVisibleRowsTable({
    width: '100%',
    selectionMode : sap.ui.table.SelectionMode.None,
    rowSelectionChange: function(e) {
        var indices = e.getParameter('rowIndices');
        for (var i = 0; i < indices.length; i++) {
            var idx = indices[i];
            if (oTable.isIndexSelected(idx)) {
                var cxt = oTable.getContextByIndex(idx);
                var path = cxt.sPath;
                var obj = oTable.getModel().getProperty(path);
                //console.log(JSON.stringify(obj));
                alert(JSON.stringify(obj));
            }
        }
    },
    columns:[new sap.ui.table.Column({
        label: '',
        width: '5%',
        template: new sap.ui.commons.CheckBox({
            checked: '{checked}'
        })
    })],
    new sap.ui.table.Column({

```

```

        label: new sap.ui.commons.TextView({
            text: "Property"
        }),
        width: '60%',
        disabled:true,
        template: new sap.ui.commons.TextView({
            text: '{property}'
        })
    }),
    new sap.ui.table.Column({
        label: new sap.ui.commons.TextView({
            text: "Type"
        }),
        width: '35%',
        template: new sap.ui.commons.TextView({
            text: '{type}'
        })
    })
]

});

var oTableLbl = new sap.ui.commons.Label({
    text : "Select Property:",
    labelFor : oTable
});

tableLayout.createRow({
    height : "70px"
}, oTableLbl,oTable);

tableLayout.createRow({
    height : "30px"
}, "" , (new sap.ui.commons.Button({
    text: 'Select visible',
    press: function(e) {
        oTable.selectAllVisibleRowsIndex('checked');
    }
})));

tableLayout.createRow({
    height : "30px"
}, "" , (new sap.ui.commons.Button({
    text: 'Select All',
    press: function(e) {
        oTable.selectAll('checked');
    }
})));

tableLayout.createRow({
    height : "30px"
}, "" , (new sap.ui.commons.Button({
    text: 'OK',
    press: function(e) {
        oTable.bindRows('/');
        var model = new sap.ui.model.json.JSONModel();
        entityResults = JSON.parse(response.replace("meta", ""));
        isErrorExists = false;
    }
})));

```



```
        <Text
            text="{Country}" />
        <ObjectNumber
            number="{ReportsTo}"
            unit="employees"
        />
    </cells>
</ColumnListItem>
</items>
</Table>
</content>
</Page>
</mvc:View>
```

Контроллер JS

```
var oModel = new
sap.ui.model.odata.ODataModel("http://services.odata.org/V2/Northwind/Northwind.svc");
this.getView().setModel(oModel);
```

Прочитайте Таблица sapui5 онлайн: <https://riptutorial.com/ru/sapui5/topic/6378/таблица-sapui5>

кредиты

| S. No | Главы | Contributors |
|-------|------------------------|---|
| 1 | Начало работы с sарuі5 | bharath muppa , Community , kuljit k , maillard , Sunil B N , Tuhin |
| 2 | Графики | Gopal Anand , inetphantom |
| 3 | Связывание агрегации | Guto |
| 4 | Таблица sарuі5 | mattymanme , Stephen S |