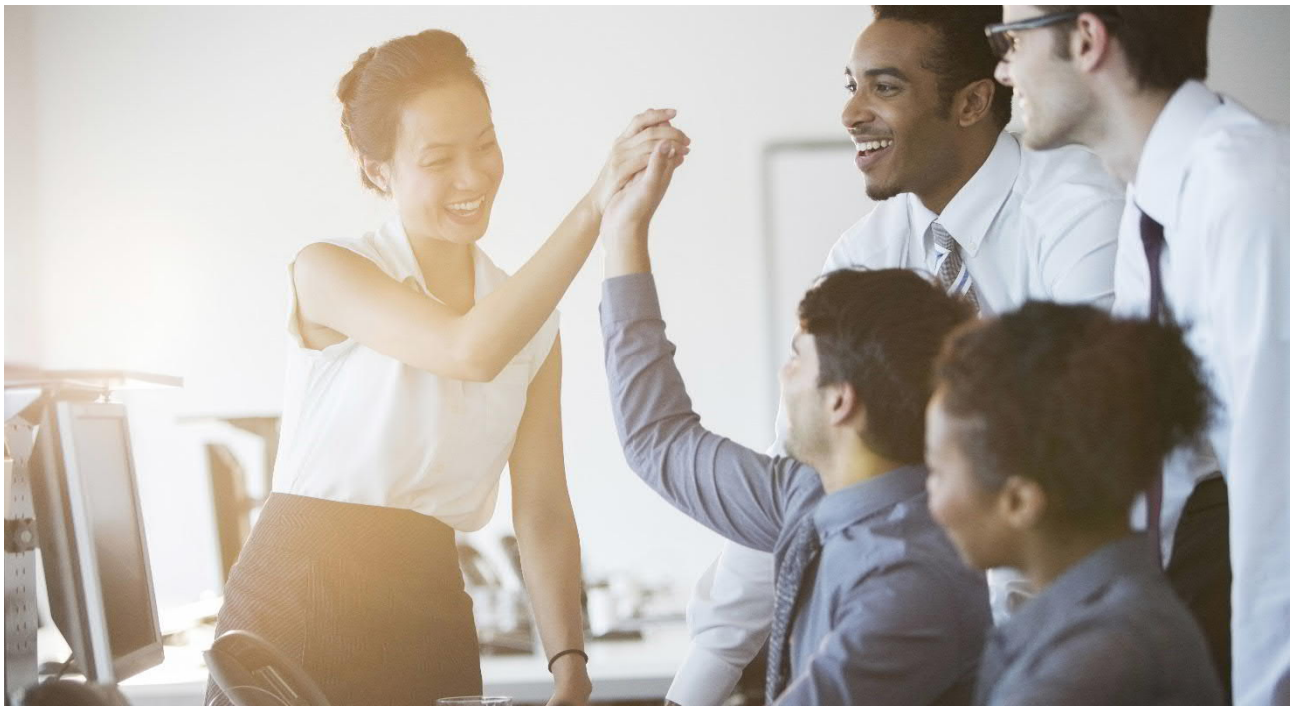

Exercise Solution: Create Menu Entries

User Interface API

PUBLIC

INTRODUCTION

In this exercise, you will perform the following tasks:

1. Add entry into the main menu (Modules)
2. Handle menu event
3. Add Context menu entry
4. Remove the menu entries

PREREQUISITE:

- This document is using the **C Sharp** (C#) language
- This document is using the Microsoft Visual Studio 2015
- Use the demo database for SAP Business One, version for SAP HANA or SAP Business One
- Credentials: User code: **manager**

GUIDELINES:

The screenshots provided here are for your reference only and may differ from the actual screenshots in your system.

© 2018 SAP SE. All rights reserved

1. TASK - ADD ENTRY INTO THE MAIN MENU (MODULES)

1.1. Add a menu entry to the "Modules" menu called **DVD Store**. Those are the menu entries which are also displayed in the Main Menu. Add a new sub-menu to that menu entry called **DVD Availability**.

```
        public static void CreateMenu()
        {
            SAPbouiCOM.Menus moduleMenus;
            SAPbouiCOM.MenuItem menuItem;
            SAPbouiCOM.MenuItem fatherMenuItem;
            SAPbouiCOM.MenuCreationParams oCreationPackage;

            try
            {
                // Find the id of the menu into which you want to add your menu item
                // ModuleMenuId = "43520"
                menuItem = SBO_Application.Menus.Item("43520");

                // Get the menu collection of SAP Business One
                moduleMenus = menuItem.SubMenus;


                // Add a pop-up menu item
                oCreationPackage =
(SAPbouiCOM.MenuCreationParams)SBO_Application.CreateObject(SAPbouiCOM.BoCreatableObjectType
.cot_MenuCreationParams);
                oCreationPackage.Checked = false;
                oCreationPackage.Enabled = true;
                oCreationPackage.Type = SAPbouiCOM.BoMenuType.mt_POPUP;
                oCreationPackage.UniqueID = "TB1_DVDStore";
                oCreationPackage.String = "DVD Store";
                oCreationPackage.Position = -1;

                fatherMenuItem = moduleMenus.AddEx(oCreationPackage);

                // add a submenu item to the new pop-up item
                oCreationPackage =
(SAPbouiCOM.MenuCreationParams)SBO_Application.CreateObject(SAPbouiCOM.BoCreatableObjectType
.cot_MenuCreationParams);
                oCreationPackage.Checked = false;
                oCreationPackage.Enabled = true;
                oCreationPackage.Type = SAPbouiCOM.BoMenuType.mt_STRING;
                oCreationPackage.UniqueID = "TB1_Avail";
                oCreationPackage.String = "DVD Availability";
                oCreationPackage.Position = -1;

                menuItem = fatherMenuItem.SubMenus.AddEx(oCreationPackage);
            }
            catch (Exception)
            {
                SBO_Application.StatusBar.SetText("Menu already exists",
SAPbouiCOM.BoMessageTime.bmt_Short, SAPbouiCOM.BoStatusBarMessageType.smt_Error);
            }

            }
```

© 2018 SAP SE. All rights reserved

2. TASK - HANDLE MENU EVENT

When you choose the menu entry, your form should be displayed.

2.1. Create a function which will handle the mandatory *MenuEvent*.

```
    public static void SBO_Application_MenuEvent(ref SAPbouiCOM.MenuEvent pVal, out bool
BubbleEvent)
     {
         BubbleEvent = true;
         if (pVal.BeforeAction && pVal.MenuUID == "TB1_Avail")
         {
             LoadFormFromXML();
         }

             }
```

2.2. Create event subscriber for the previously create function before the application run will be called in the **Main** function.

```
SBO_Application.MenuEvent += new
SAPbouiCOM._IApplicationEvents_MenuEventEventHandler(SBO_Application_MenuEvent);
```

2.3. Comment the functions call **LoadFormFromXML** and **ExtendCancelButtonWithEvent** in the Main function.

```
//LoadFormFromXML();
//ExtendCancelButtonWithEvent();
```

2.4. Extend the **SetEventFilters** function with the *Menu_Click* event type.

```
oFilter = oFilters.Add(SAPbouiCOM.BoEventTypes.et_MENU_CLICK);
```

3. TASK - ADD CONTEXT MENU ENTRY

3.1. Add another menu only visible when your form is open. This menu should appear under the GoTo menu.

```
    private static void CreateContextMenu(SAPbouiCOM.Form oForm)
     {
         SAPbouiCOM.MenuCreationParams oCreationPackage;
         oCreationPackage =
(SAPbouiCOM.MenuCreationParams)SBO_Application.CreateObject(SAPbouiCOM.BoCreatableObjectType
.cot_MenuCreationParams);
         oCreationPackage.Type = SAPbouiCOM.BoMenuType.mt_STRING;
         oCreationPackage.UniqueID = "TB1_Remove";
         oCreationPackage.String = "Remove Menu";
         oForm.Menu.AddEx(oCreationPackage);

        }
```

3.2. Call the **CreateContextMenu** function in the **LoadFormFromXML** function.

```
CreateContextMenu(oForm);
```

© 2018 SAP SE. All rights reserved

4. TASK - REMOVE THE MENU ENTRIES

4.1. Create an event handler for the Context menu entry “Remove Menu” to remove all menu entries. The **SBO_Application_MenuEvent** function needs to be extended.

```
if(pVal.BeforeAction && pVal.MenuUID == "TB1_Remove")
    {
        SBO_Application.Menus.RemoveEx("TB1_Avail");
        SBO_Application.Menus.RemoveEx("TB1_DVDStore");
        SBO_Application.Menus.RemoveEx("TB1_Remove");

     }
```

A similar solution can be found in the SDK UI samples (in the SDK Folder – see Appendix “SDK Installations” for more information), COM UI/ 05.AddingMenuItems

© 2018 SAP SE. All rights reserved

www.sap.com

© 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.
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. Please see http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark for additional trademark information and notices. 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 SE or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP SE 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.
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, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.