



MENU



Tous les Tutoriels

Java

Maven

Gradle

Servlet/Jsp

Thymeleaf

Spring

Struts2

Hibernate

Java Web Service

JavaFX

SWT

Oracle ADF

Android

iOS

Python

Swift

C#

C/C++

Ruby

Dart



Batch	
Database	
Oracle APEX	
Report	
Client	
ECMAScript / Javascript	
TypeScript	
NodeJS	
ReactJS	
Flutter	
AngularJS	
HTML	
CSS	
Bootstrap	
OS	
Navigateur	
Git	
SAP	
Amazon AWS	
Others	



Le Tutoriel Eclipse RCP 4 pour débutant - Application e4 Workbench

View more Tutorials:

Tutoriels Eclipse Technology
Eclipse RCP

1. Les installations requises
2. Créer un nouveau Java Workspace
3. Créer Eclipse 4 RCP Project vide
4. Structure de l'application Eclipse 4 RCP vide
5. Handler et Command
6. Créer des Menus
7. Create Toolbar
8. Part



Suivez-nous sur notre fanpage pour recevoir des notifications chaque fois qu'il y a de nouveaux articles.  Facebook

1- Les installations requises

Installez Plugin **WindowBuilder**, il s'agit d'un *Plugin* qui vous permet de concevoir l'interface d'application SWT par glisser-déplacer facilement.
Voir les instructions d'installation à :

- Installer WindowBuilder pour Eclipse

Installing e4 Tools Developer Sources:

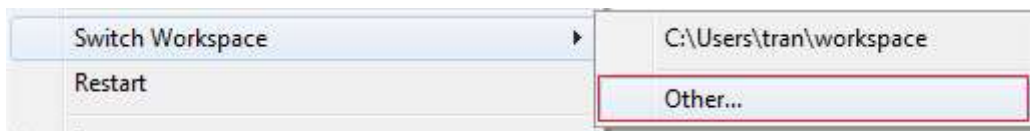


- [TODO Link?](#)

2- Créer un nouveau Java Workspace

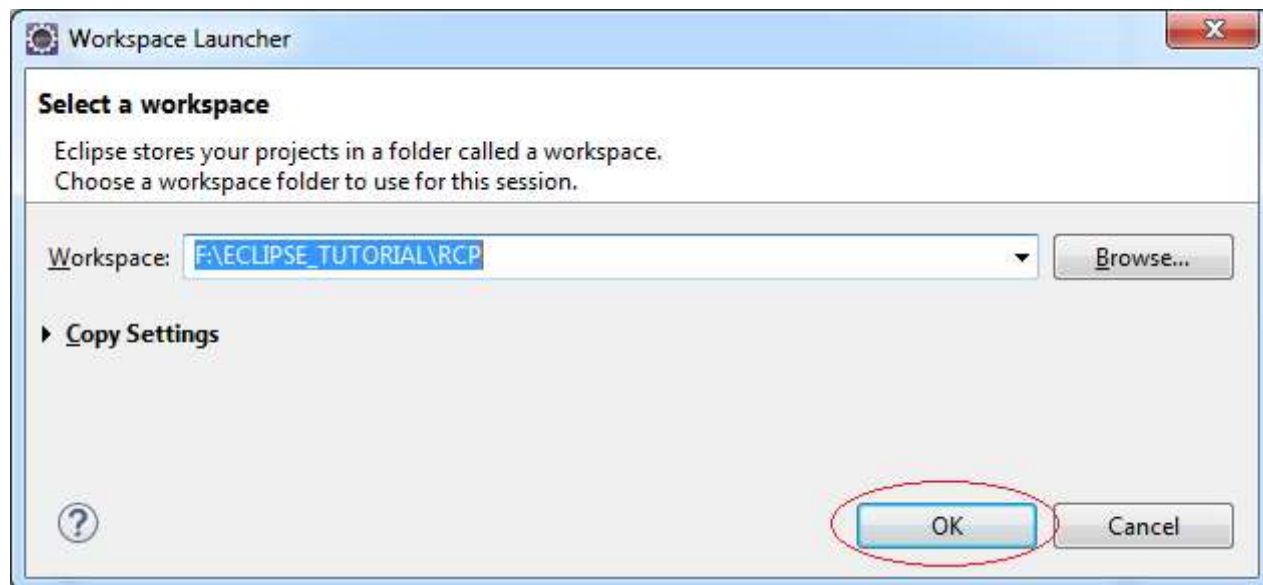
Pour programmer l'applications de RCP, vous devez créer un nouvel **Java Workspace** :

- **File/Switch Workspace/Other..**



Saisissez :

- **F:\ECLIPSE_TUTORIAL\RCP**

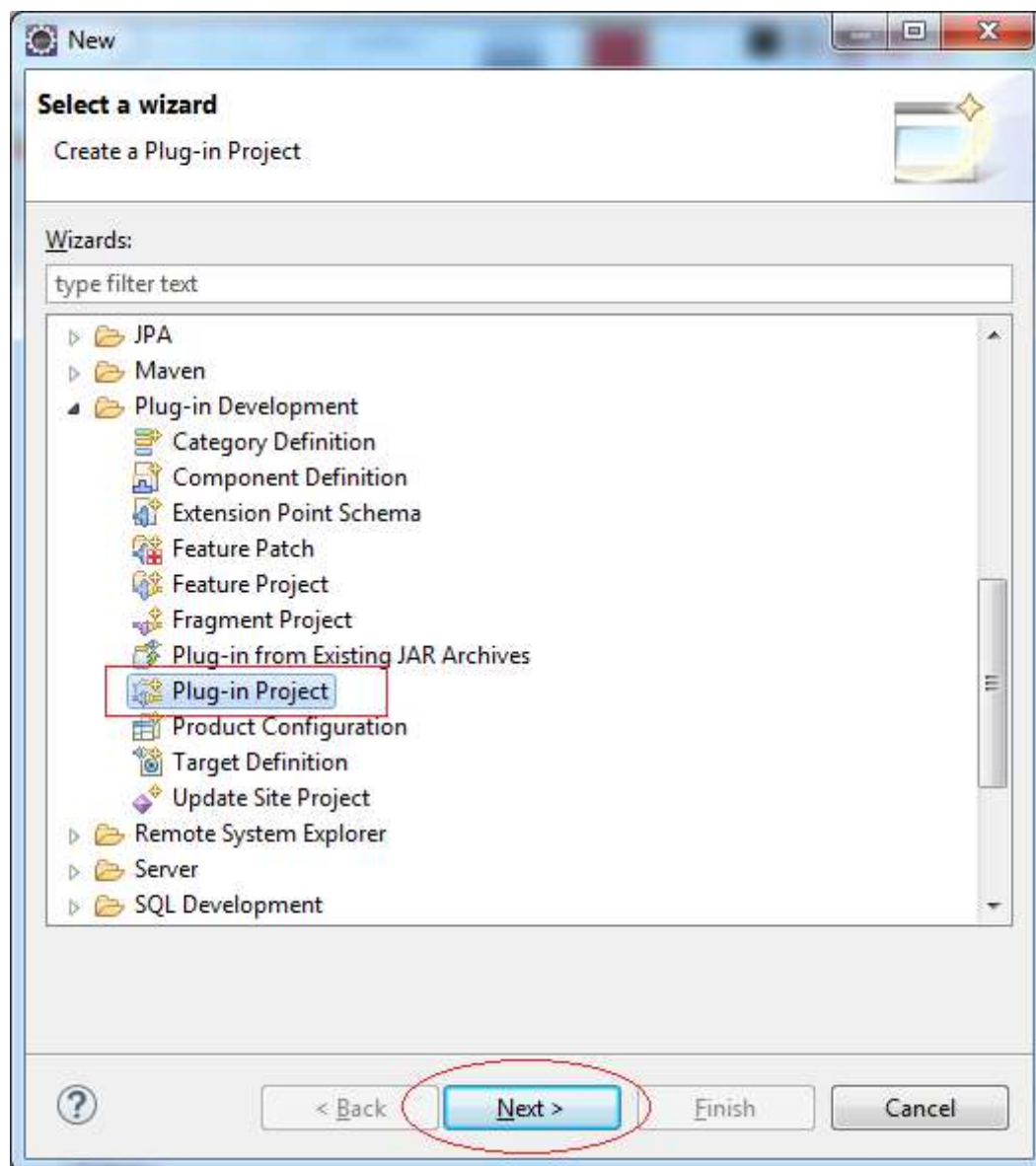


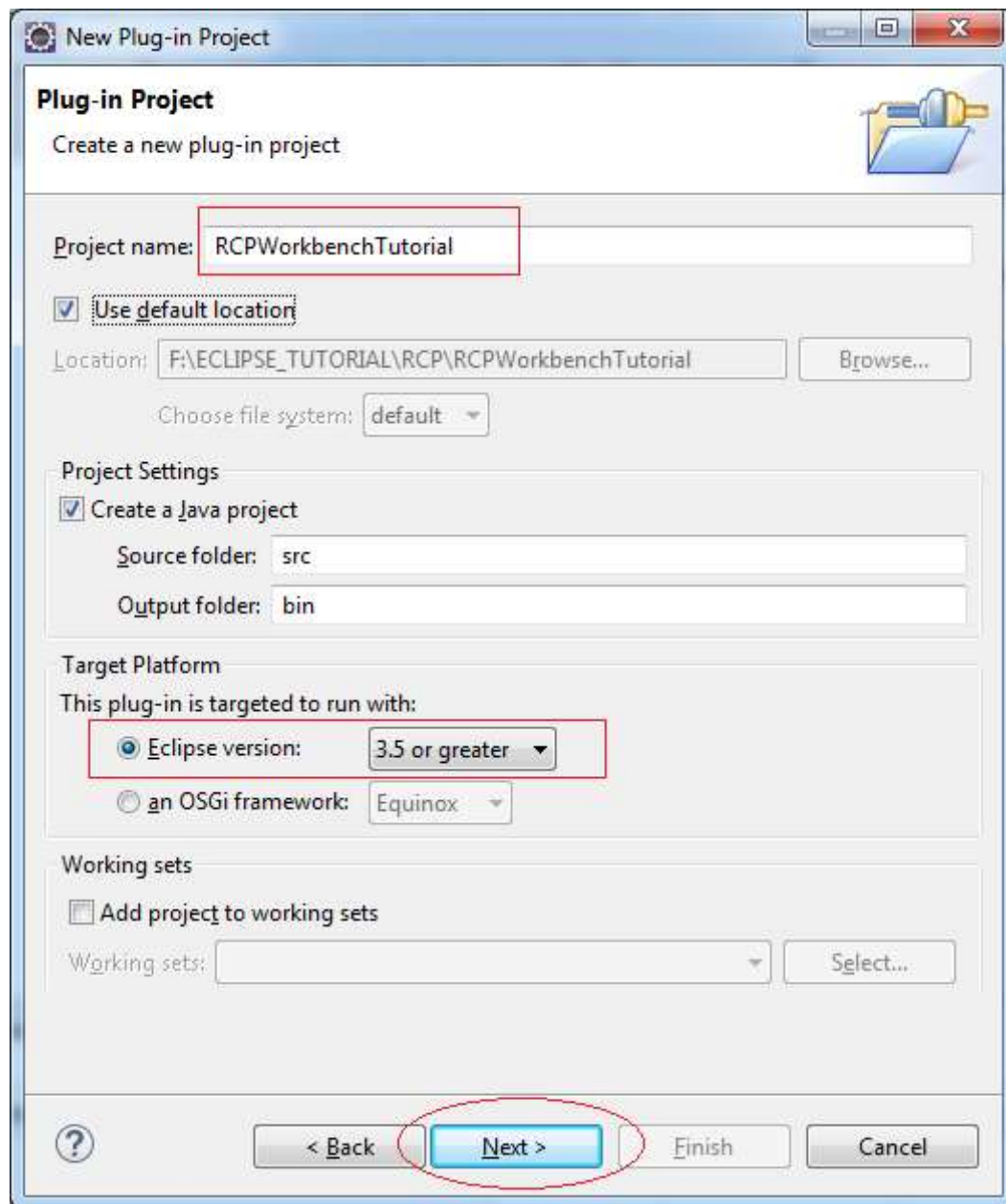
3- Créer Eclipse 4 RCP Project vide

Dans ce document, je vais vous donner des instructions de la création d'une application RCP dès le début (Ne pas suivre le modèle disponible), donc je vais créer une application d'Eclipse RCP vide

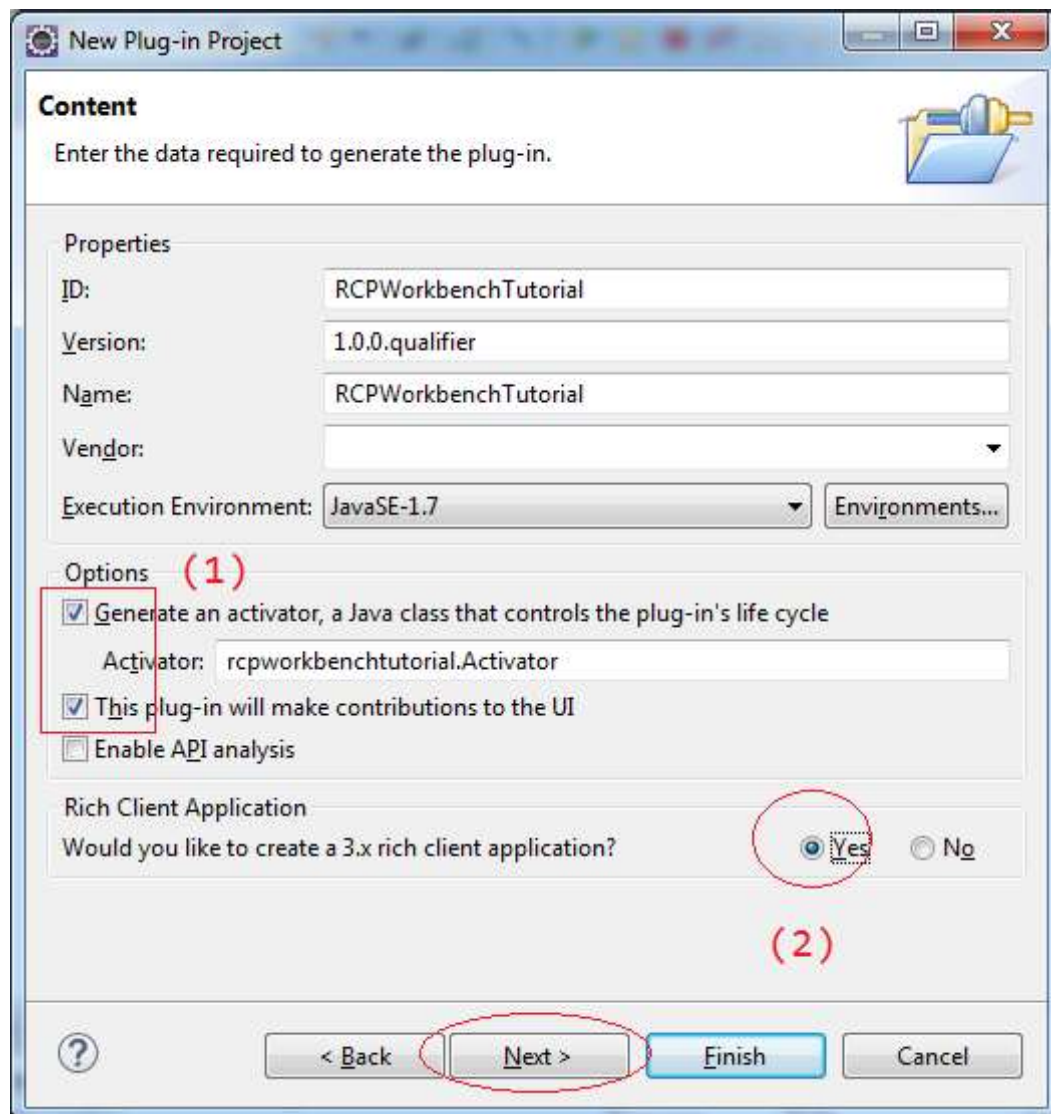
Sur Eclipse sélectionnez :

- **File/New/Other...**

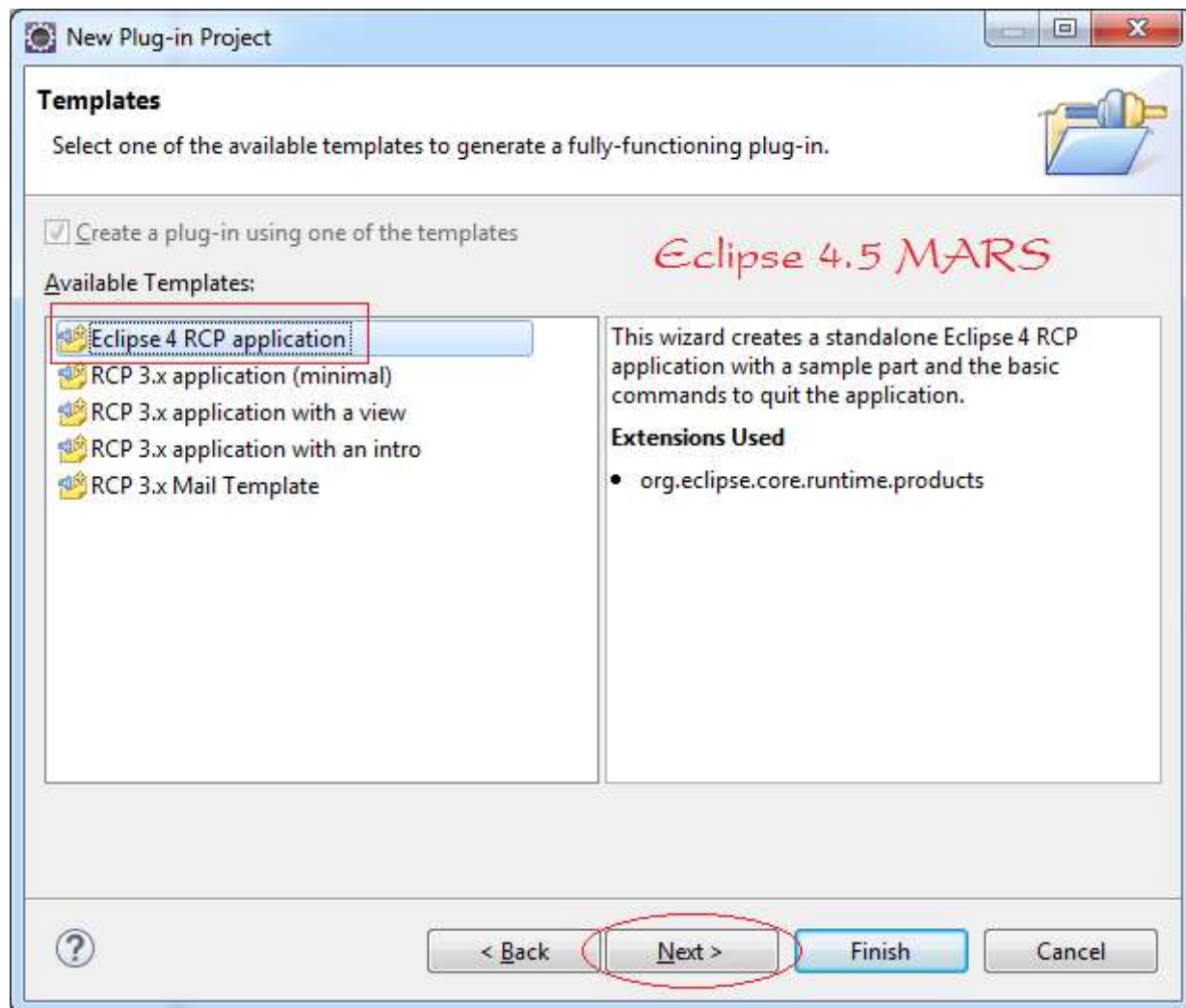




1. Cochez la case sélectionnée sur le (1)
2. Dans la zone (2) sélectionnez "Yes" pour créer RCP Project (Exécutant sur le Desktop), alors, il va créer RAP Project (Exécutant sur le Web)

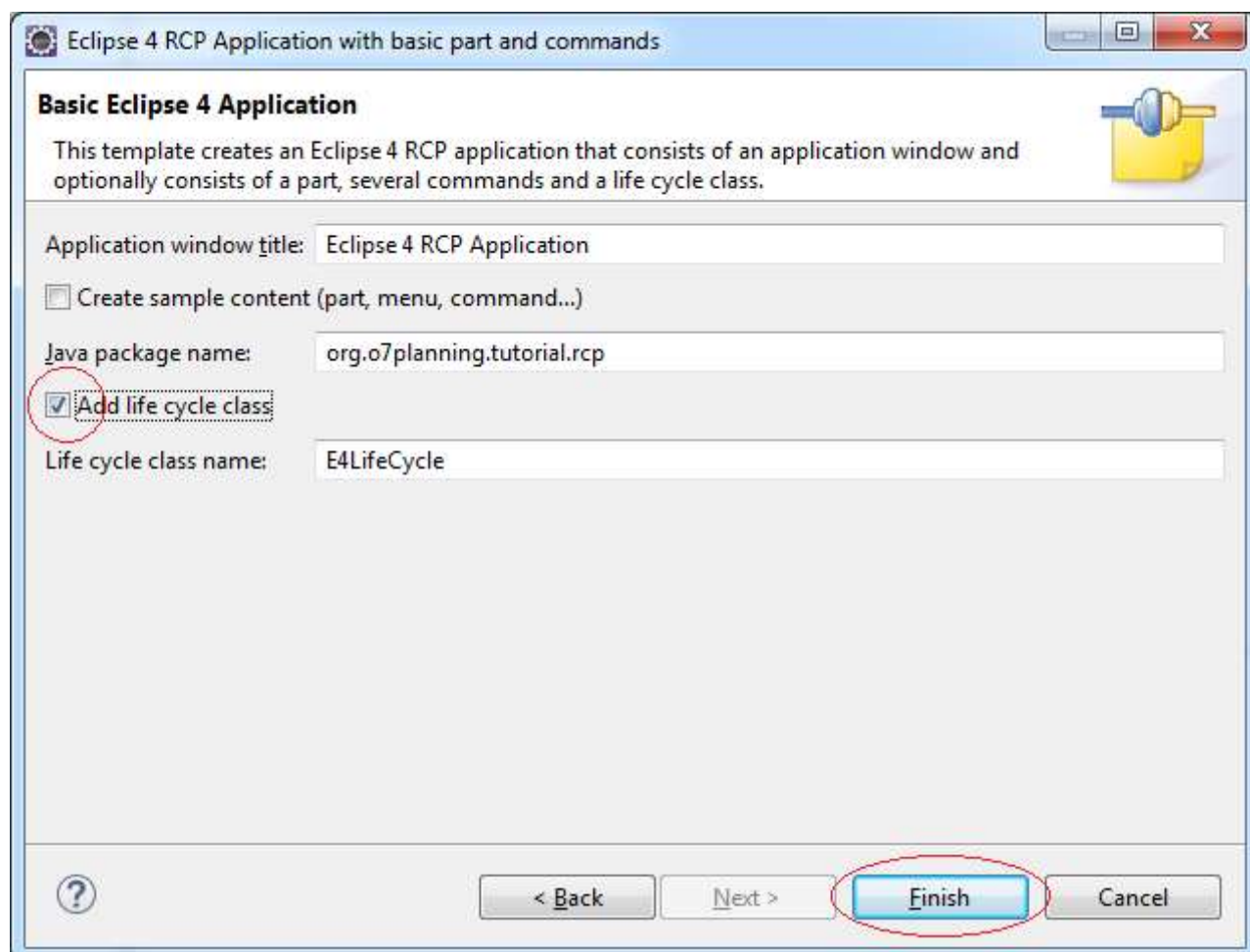


Sélectionnez "Eclipse 4 RCP Application"

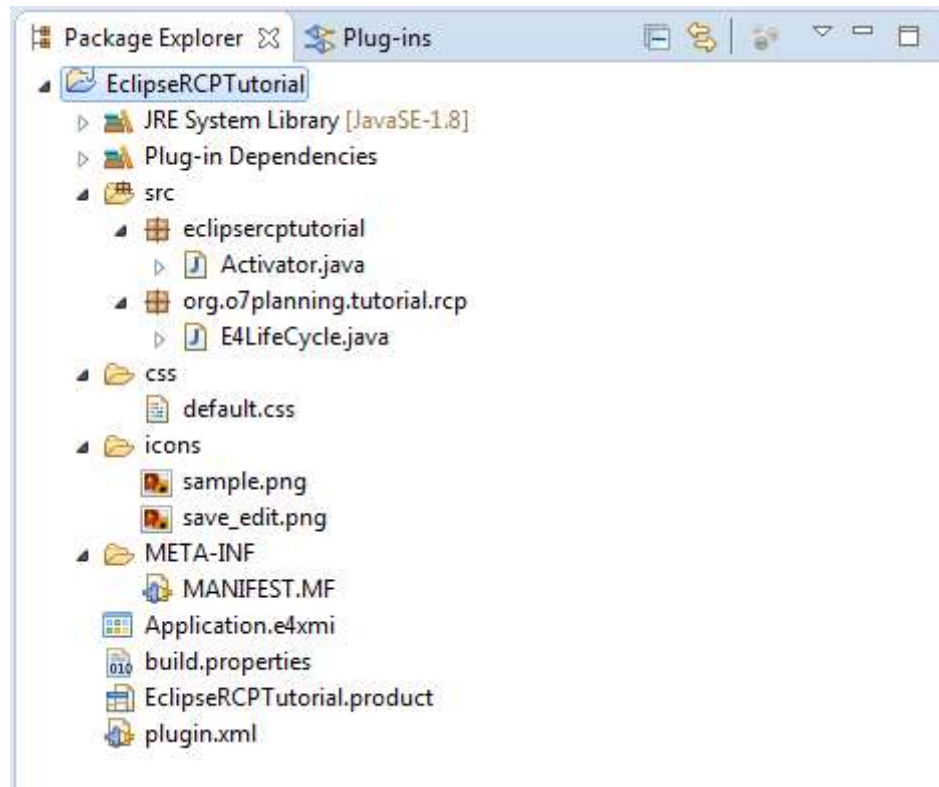


Saisissez :

- **Java package name:** org.o7planning.tutorial.rcp

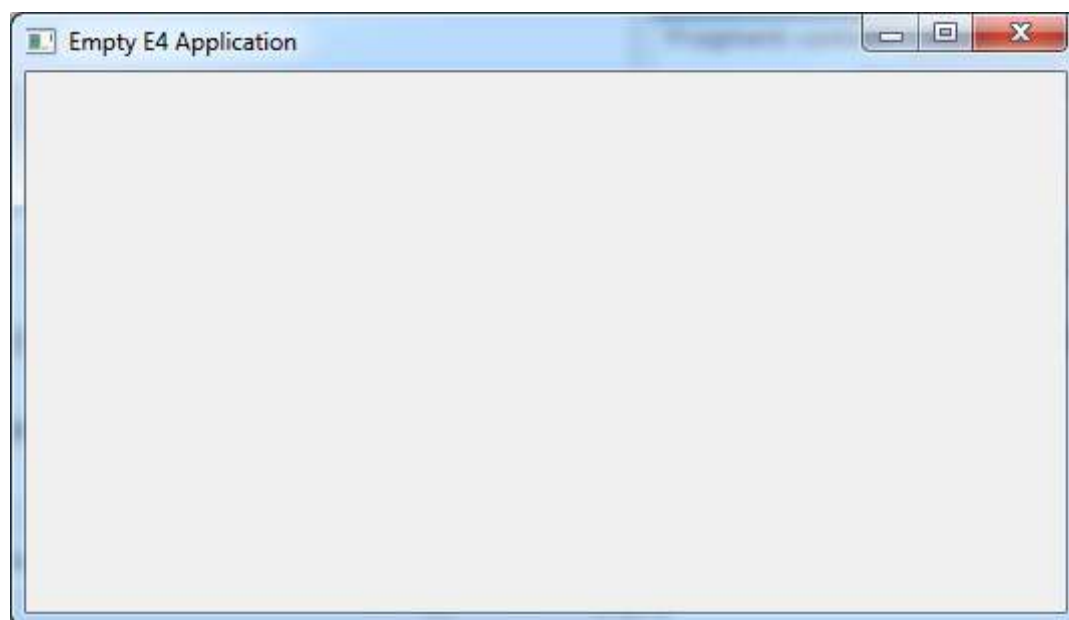
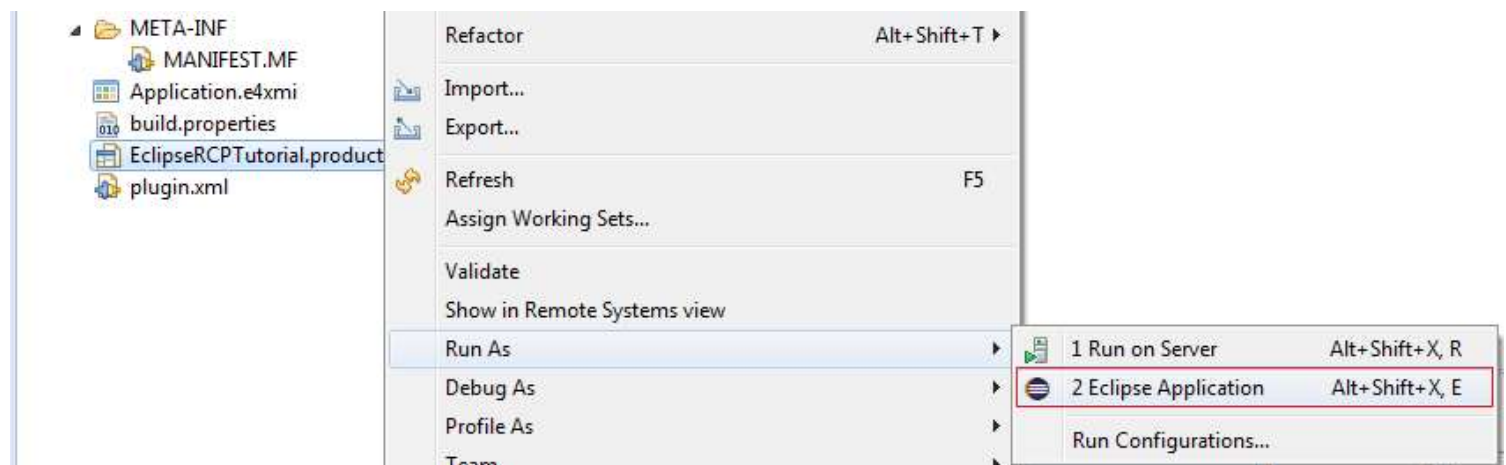


Le projet a été créé :



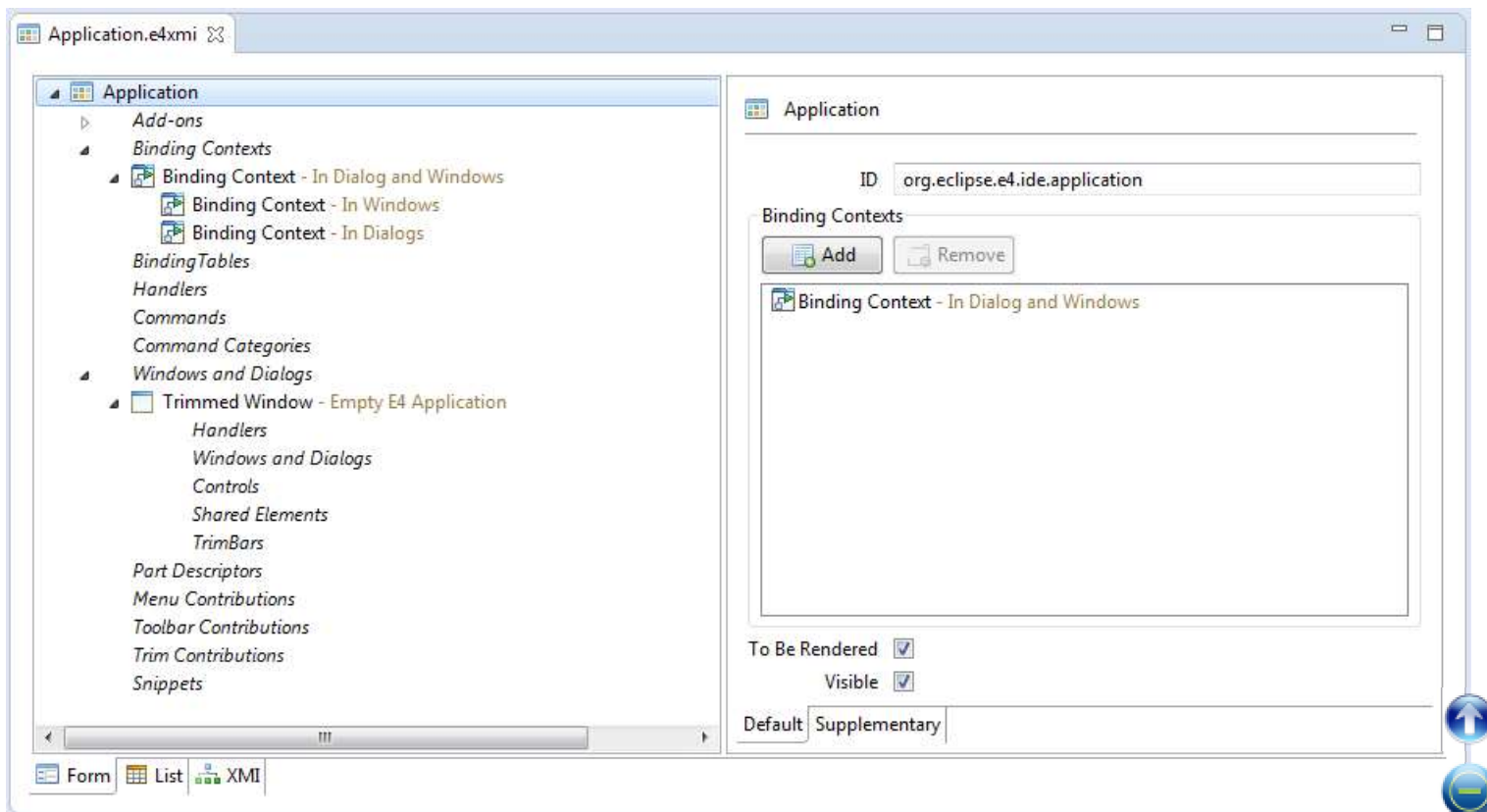
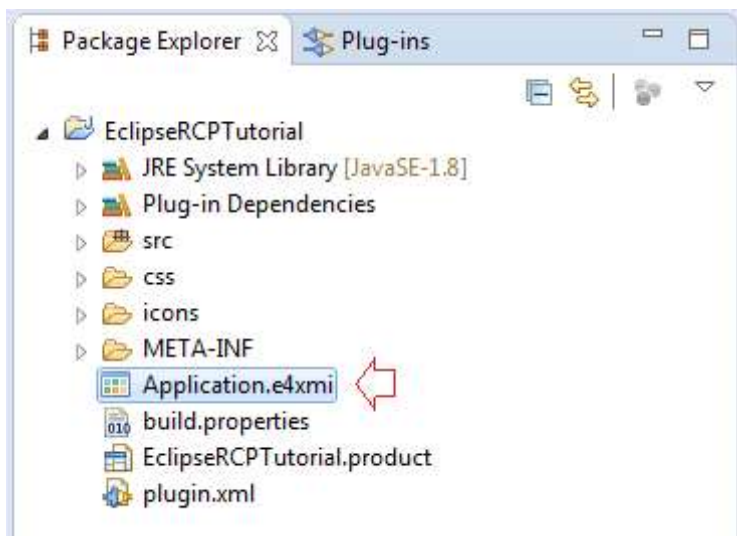
Exécutez l'application :

Cliquez sur le bouton droit sur **EclipseRCPTutorial.product** et sélectionnez **Run As/Eclipse Application**

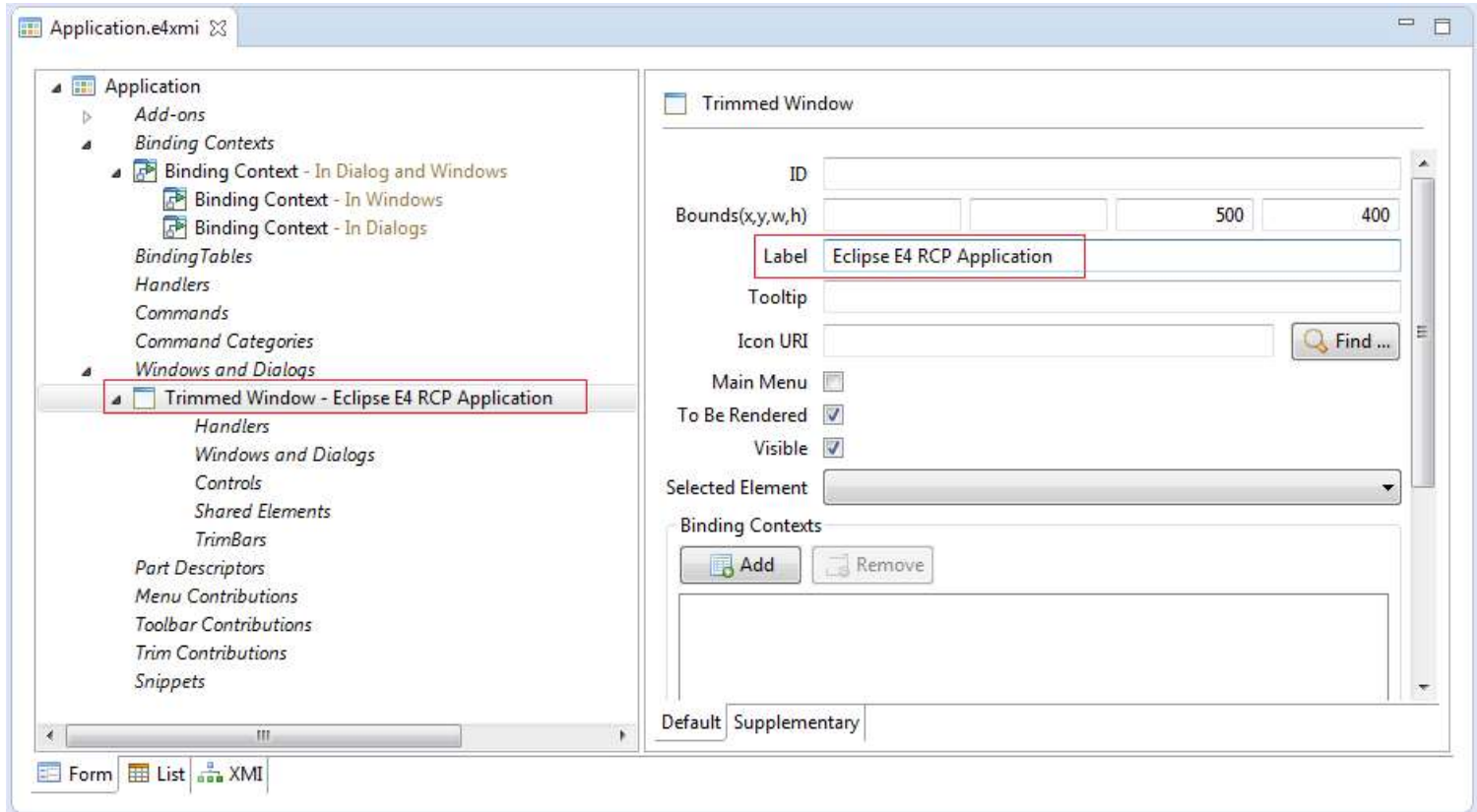


4- Structure de l'application Eclipse 4 RCP vide

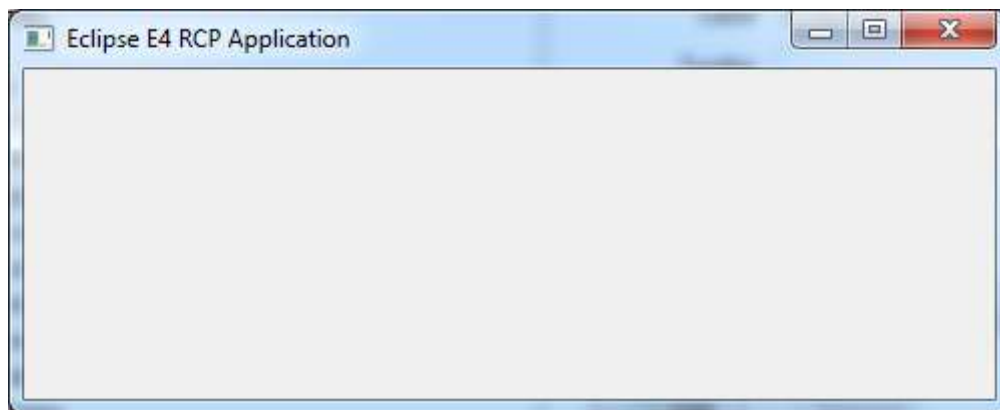
L'application d'**Eclipse 4 RCP** vide a été créée par **Eclipse** , voyez sa structure. Ouvrez le fichier **Application.e4xmi** :



Modifiez le titre de l'application à "**Eclipse E4 RCP Application**" comme l'exemple ci-dessous.



Et redémarrez l'application :



5- Handler et Command

RCP Framework construit un grand nombre de Commands, par exemple quatre Commands avec l'ID suivante :



1. org.eclipse.ui.file.exit

- Commande pour quitter l'application

org.eclipse.ui.file.open

- La commande pour ouvrir un fichier.

org.eclipse.ui.file.save

- La commande save editor

2. org.eclipse.ui.help.aboutAction

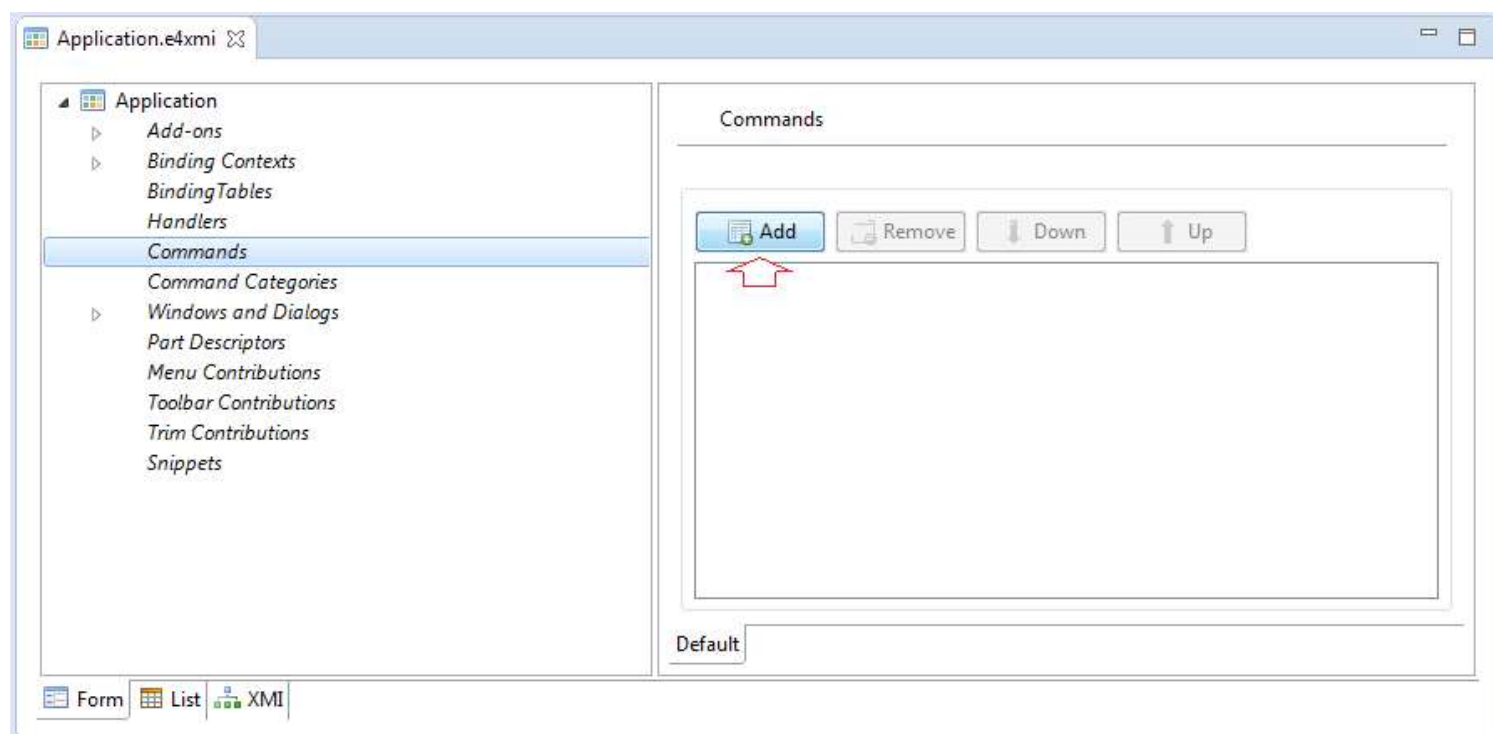
- La commande qui s'ouvre la fenêtre About.

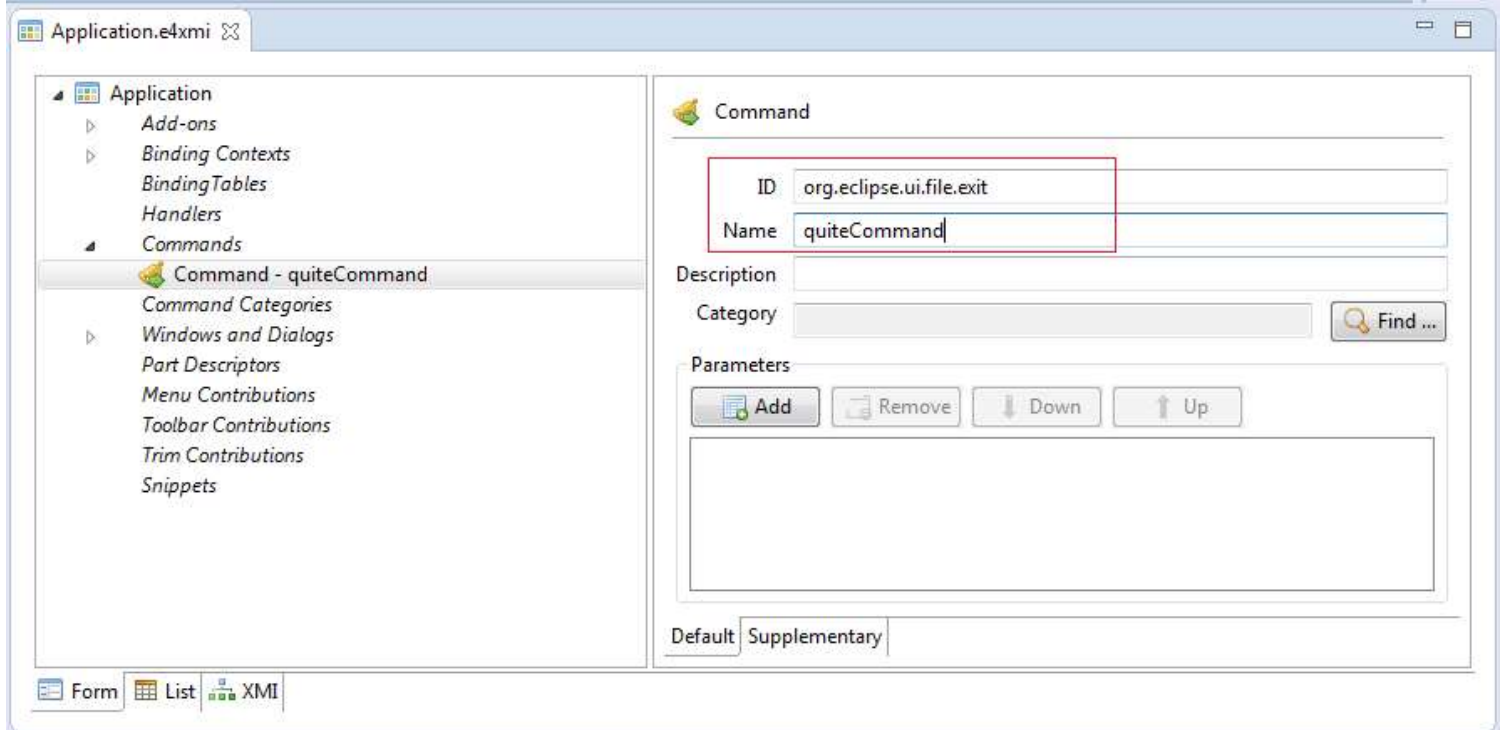
//

Vous pouvez voir une liste de la **command** qui a été construite de **RCP Framework** à :

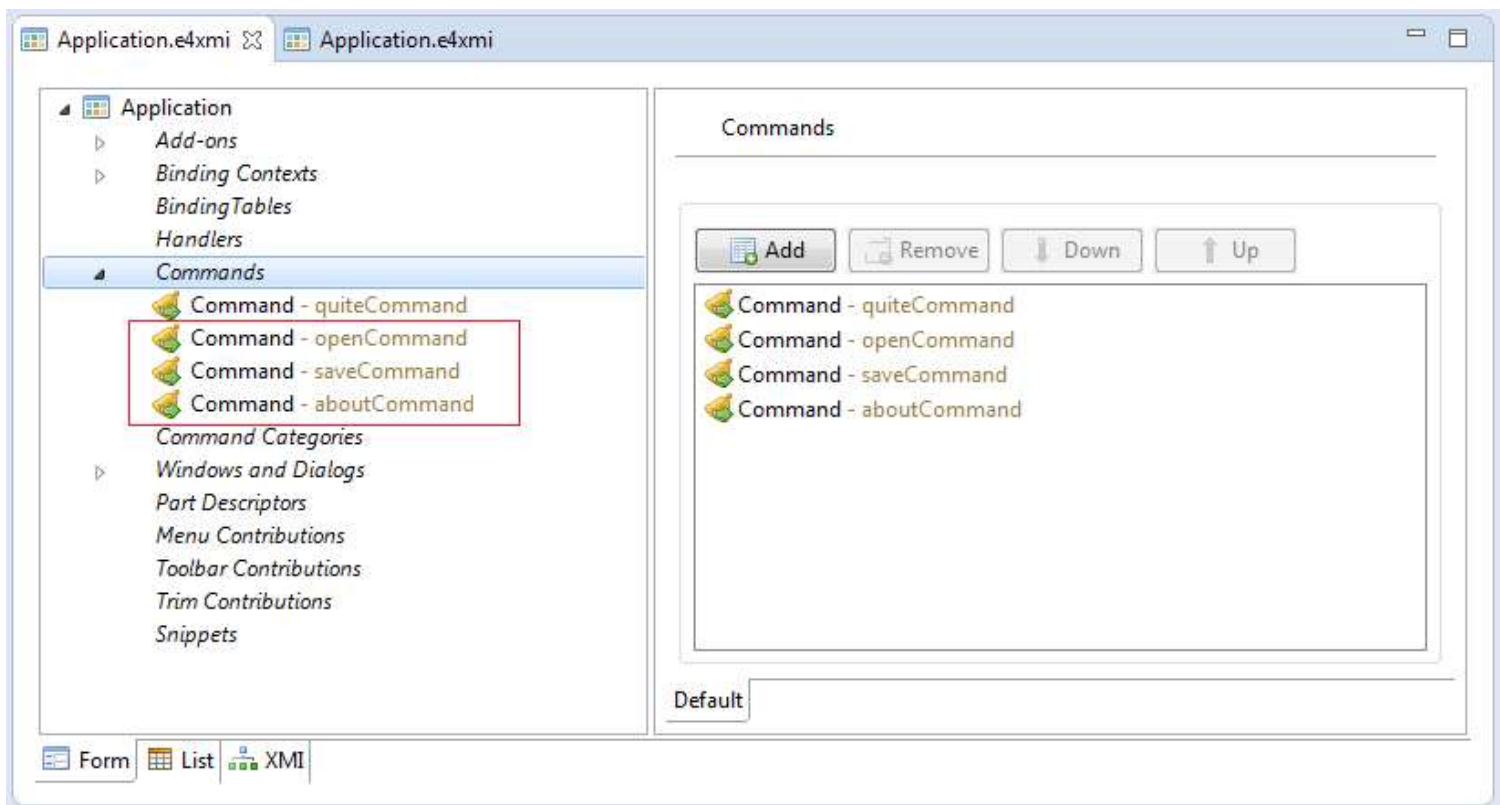
- <http://help.eclipse.org/mars/topic/org.eclipse.platform.doc.isv/reference/api/org/eclipse/ui/actions/ActionFactory.html>

Créez une nouvelle Command nommé **quiteCommand**, appelée de sortir l'application qui a été construite par **RCP Framework**.





De même manière, nous créons trois autres commandes :



Ouvrez le fichier sélectionné.

- **ID:** org.eclipse.ui.file.open
- **Name:** openCommand

Exécutez la commande d'enregistrement (save) le contenu sur l'Éditeur (Editor).

- **ID:** org.eclipse.ui.file.save
- **Name:** saveCommand

AboutCommand:



- **ID:** org.eclipse.ui.help.aboutAction
- **Name:** aboutCommand

Handler sont des classes de traitement des commandes de Menu ou Toolbar. Lorsque vous cliquez sur **MenuItem** ou **ToolItem**, c'est-à-dire que vous appelez une commande, avant l'exécution de la Commande, Handler sera exécutée avant, vous pouvez annuler la commande qui est exécutée dans le Handler.

Handler sont des classes qui exécuteront la tâche écrite dans la méthode annotée par **@Execute**.

Je vais créer quatre classes Handler :

AboutHandler.java

```
package org.o7planning.tutorial.rcp.handler;

import org.eclipse.e4.core.di.annotations.Execute;
import org.eclipse.jface.dialogs.MessageDialog;
import org.eclipse.swt.widgets.Shell;

public class AboutHandler {

    @Execute
    public void execute(Shell shell) {
        MessageDialog.openInformation(shell, "About", "Eclipse 4 RCP Application");
    }
}
```

OpenHandler.java

```
package org.o7planning.tutorial.rcp.handler;

import org.eclipse.e4.core.di.annotations.Execute;
import org.eclipse.swt.widgets.FileDialog;
import org.eclipse.swt.widgets.Shell;

public class OpenHandler {

    @Execute
    public void execute(Shell shell){
        FileDialog dialog = new FileDialog(shell);
        dialog.open();
    }
}
```

QuitHandler.java



```
package org.o7planning.tutorial.rcp.handler;
```

```
import org.eclipse.e4.core.di.annotations.Execute;
```

```
import org.eclipse.e4.ui.workbench.IWorkbench;
```

```
import org.eclipse.jface.dialogs.MessageDialog;
```

```
import org.eclipse.swt.widgets.Shell;
```

```
public class QuitHandler {  
    @Execute  
    public void execute(IWorkbench workbench, Shell shell){  
        if (MessageDialog.openConfirm(shell, "Confirmation",  
            "Do you want to exit?")) {  
            workbench.close();  
        }  
    }  
}
```

SaveHandler.java

```
package org.o7planning.tutorial.rcp.handler;
```

```
import org.eclipse.e4.core.di.annotations.CanExecute;
```

```
import org.eclipse.e4.core.di.annotations.Execute;
```

```
import org.eclipse.e4.ui.workbench.modeling.EPartService;
```

```
public class SaveHandler {  
  
    @CanExecute  
    public boolean canExecute(EPartService partService) {  
        if (partService != null) {  
            return !partService.getDirtyParts().isEmpty();  
        }  
        return false;  
    }  
  
    @Execute  
    public void execute(EPartService partService) {  
        partService.saveAll(false);  
    }  
}
```

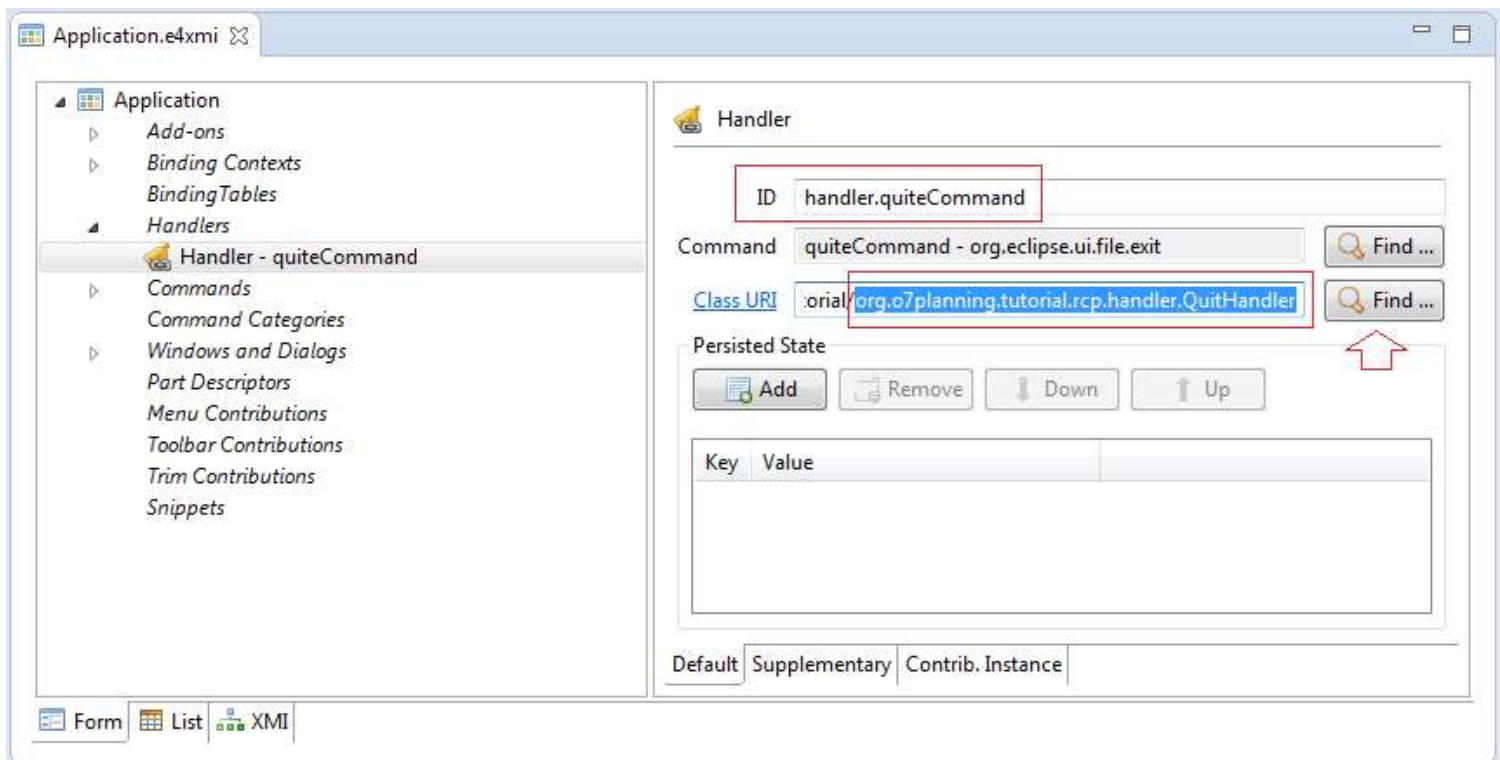
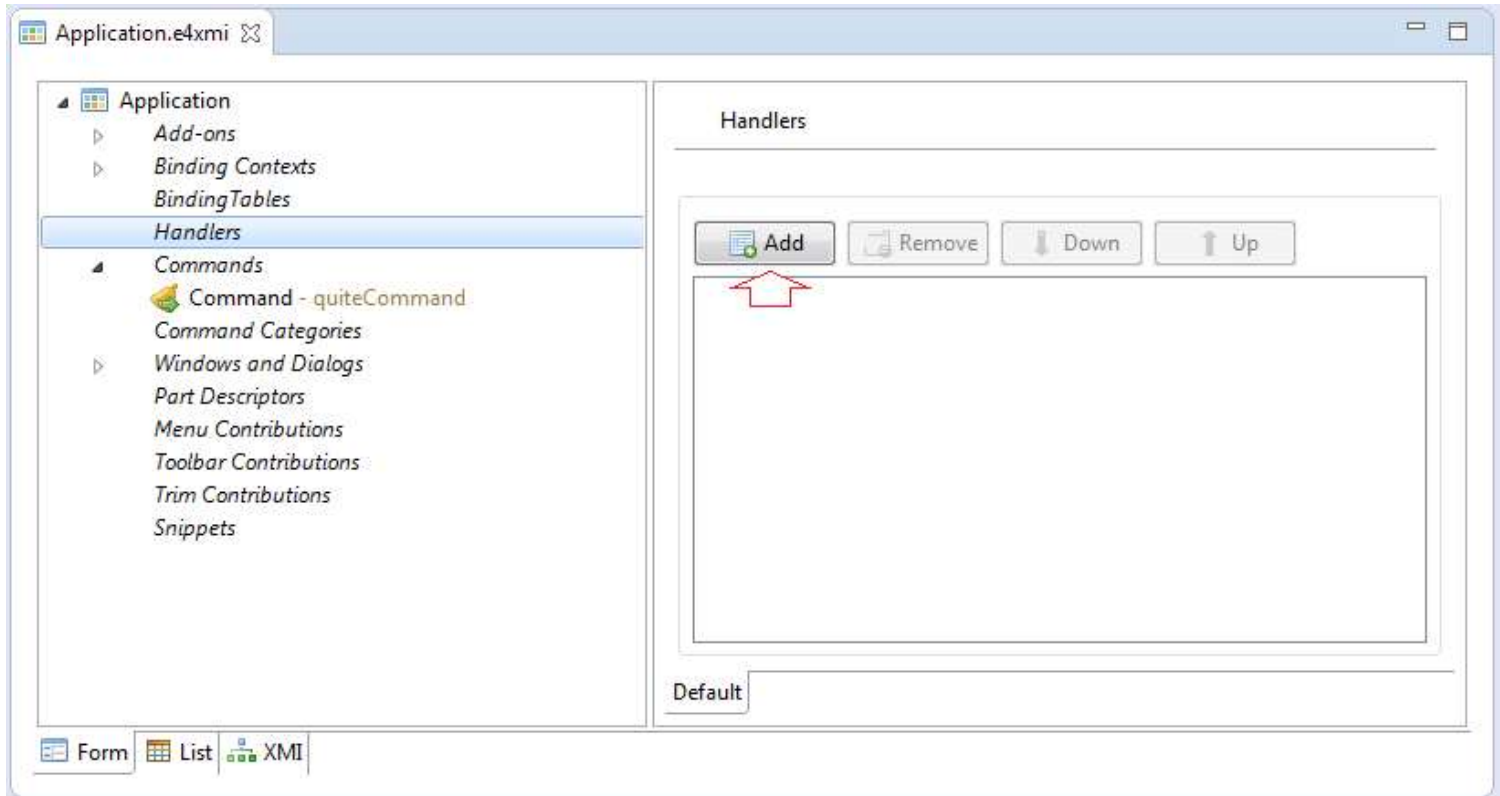
Déclaration des classes Handler avec des applications.

Pour déclarer un Handler avec Application, vous devez déclarer l'ID, Handler class et Command.

Par exemple, l'utilisateur ferme l'application en cliquant sur menu Item Exit, ce menu Item associé

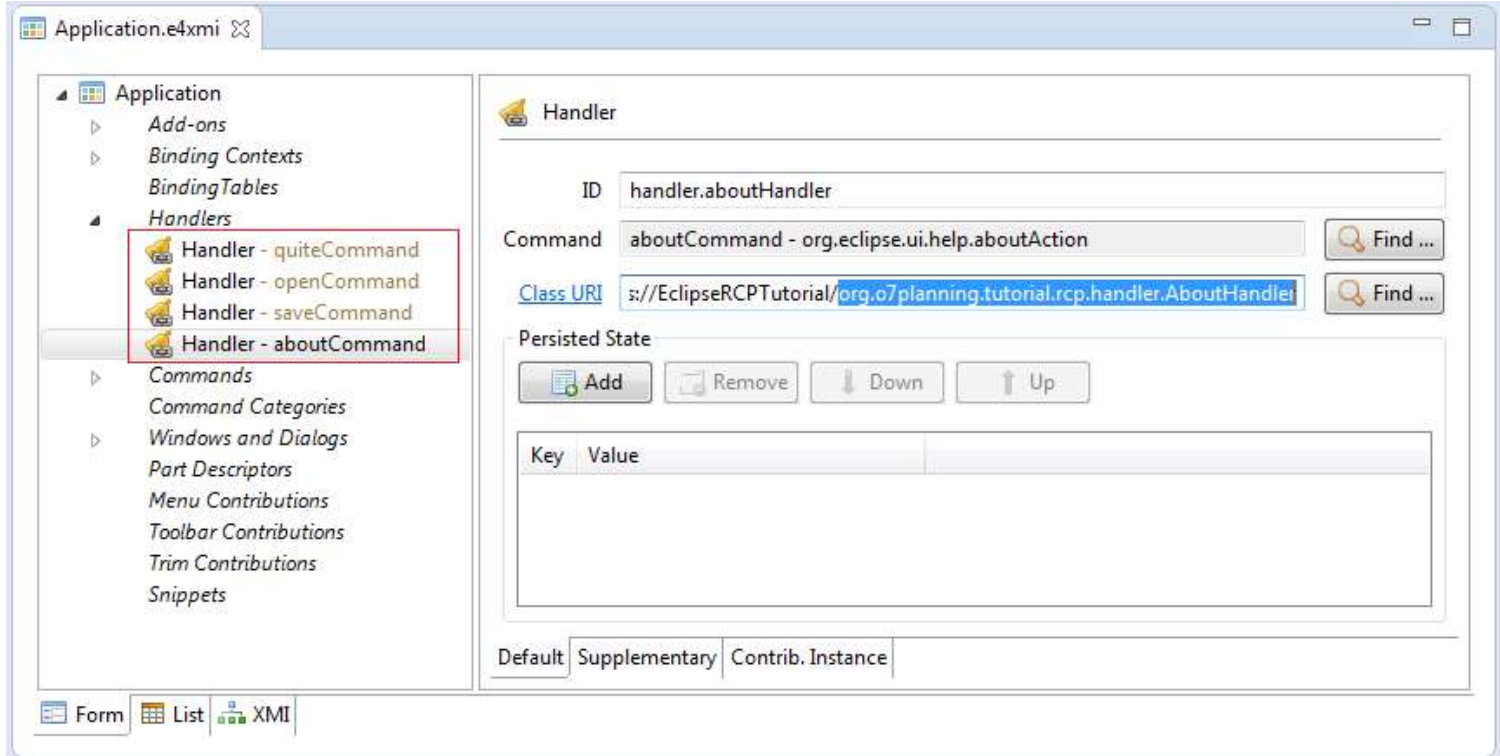


à **quiteCommand**, QuiteHandler est la classe de traitement cette commande (comme déclaré ci-dessus), la classe **QuiteHandler** demande aux utilisateurs de vouloir fermer l'application ou non, si ou, la commande **quiteCommand** sera exécutée.



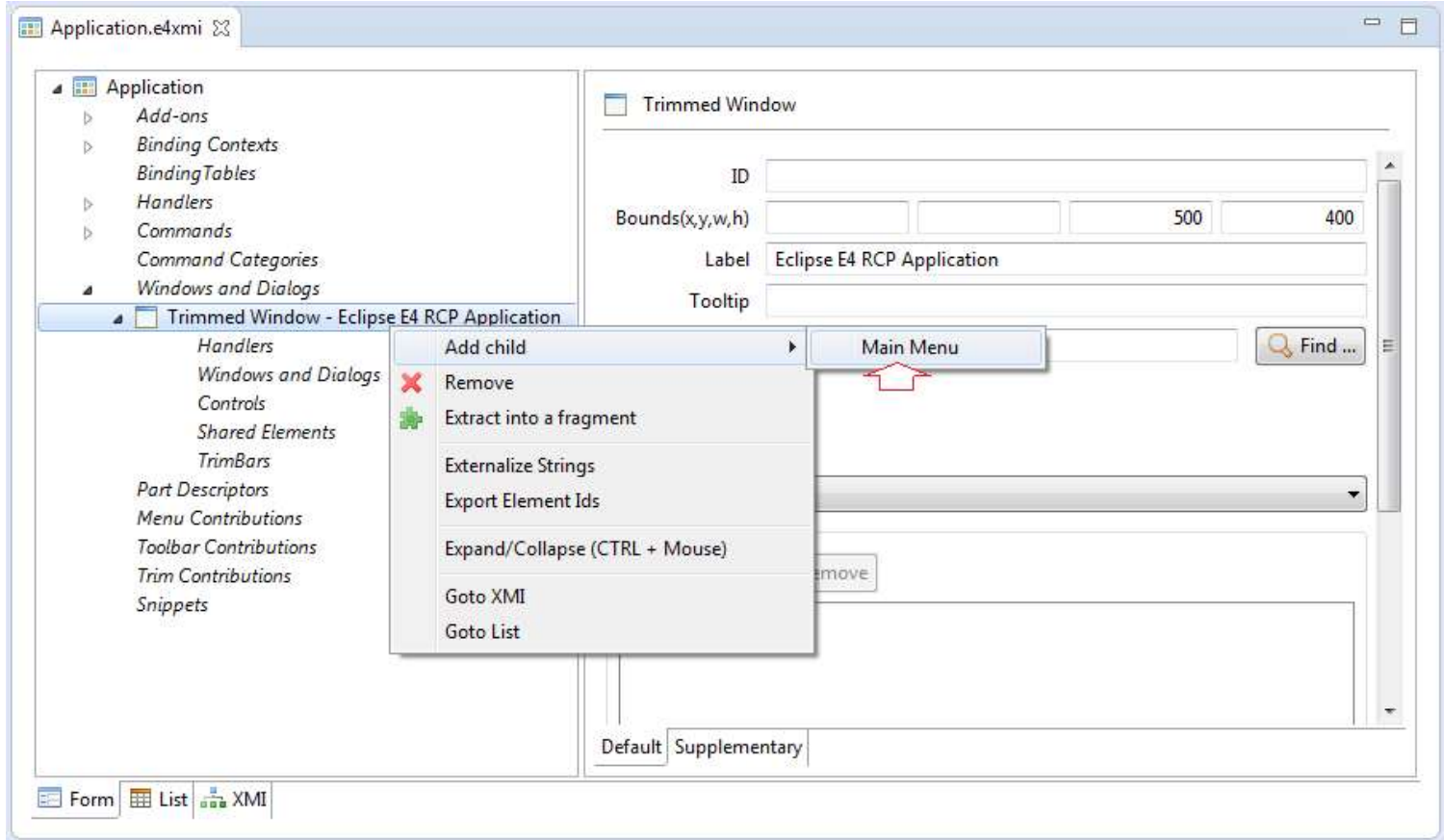
De même façon, déclarez les autres Handler.



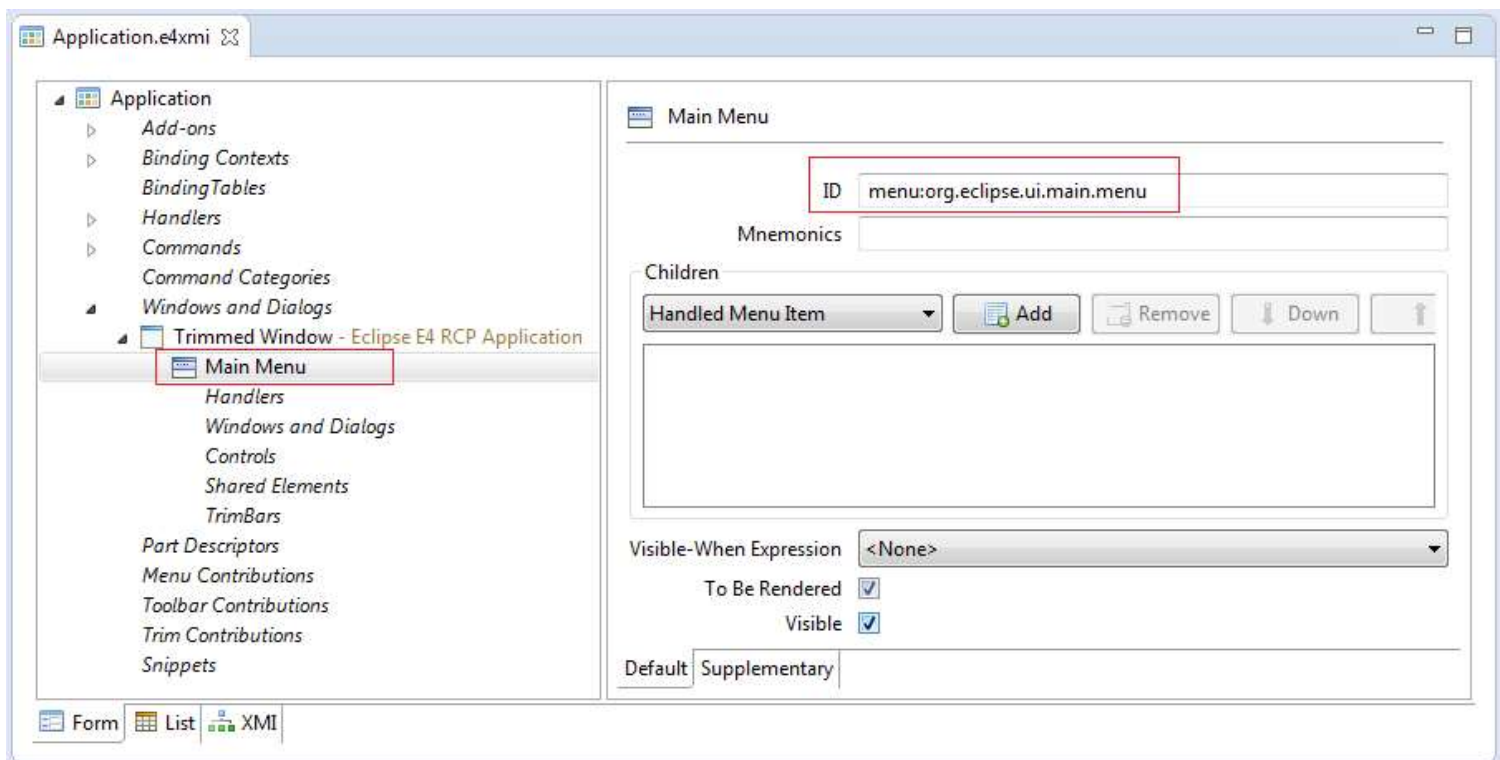


6- Créer des Menus

Créez le menu principal (menu principal) :



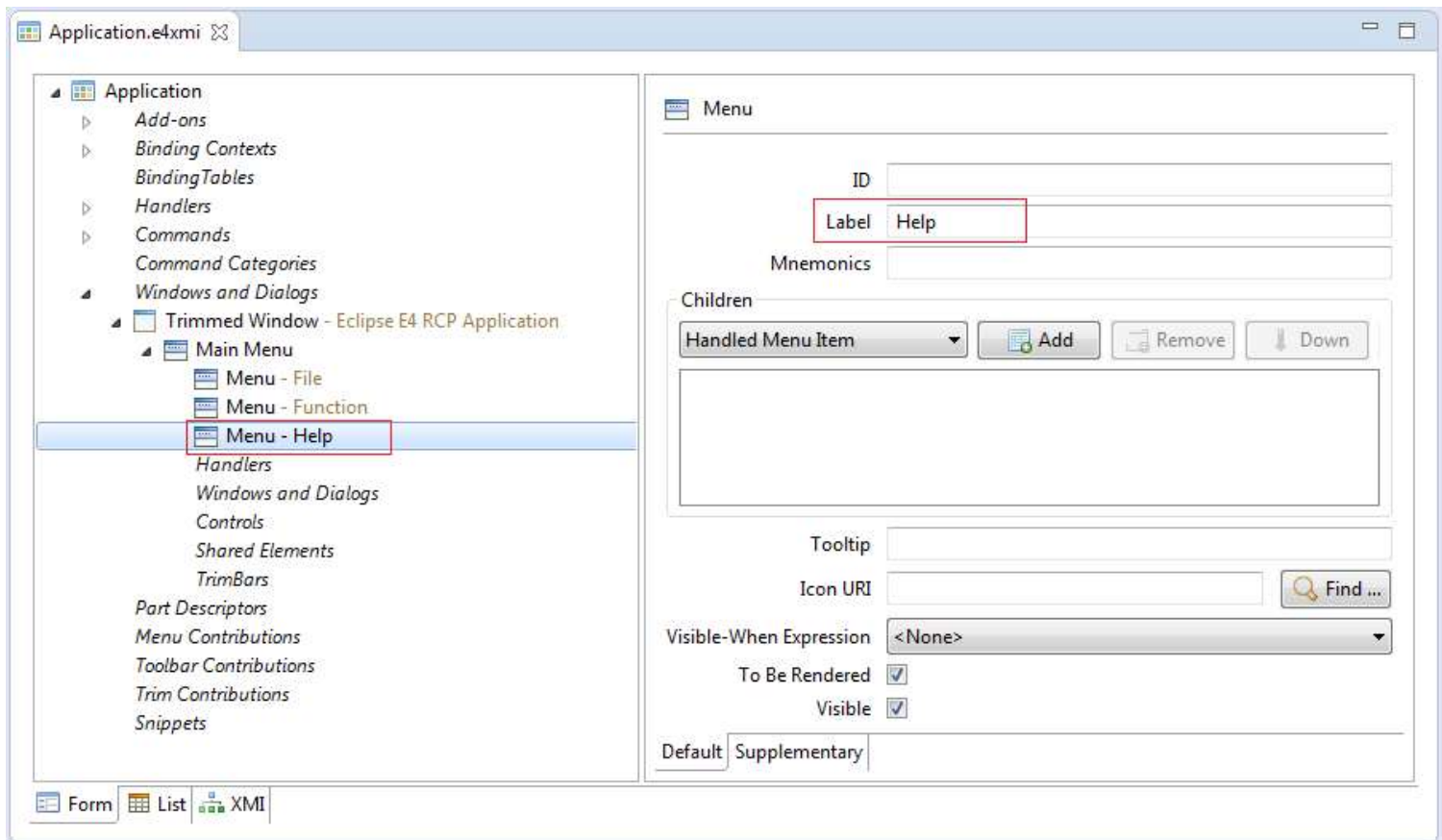
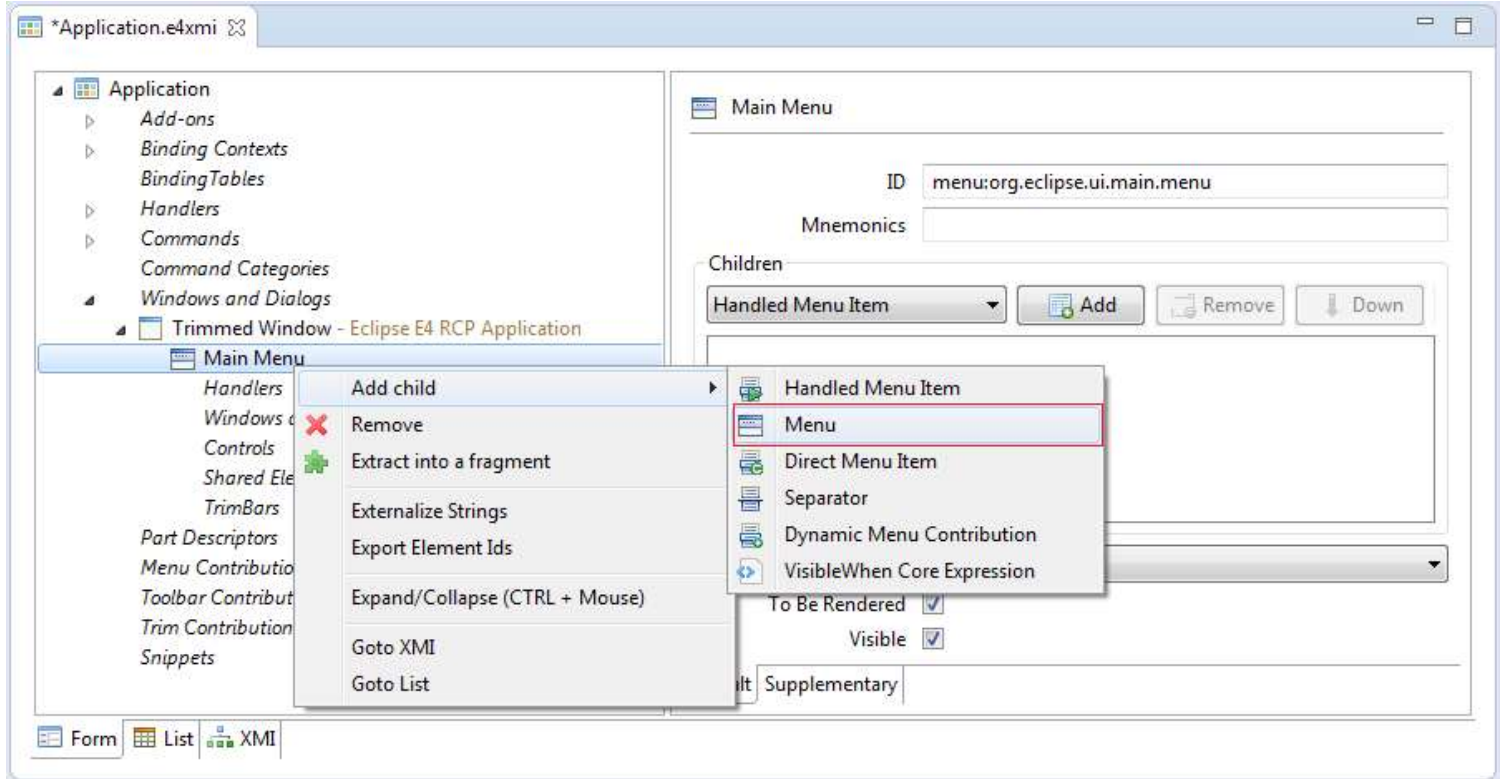
Saisissez l'ID du menu principal, il doit être nommé : ***menu:org.eclipse.ui.main.menu***



Créez trois sous-menus :

- ***File, Function, Help***

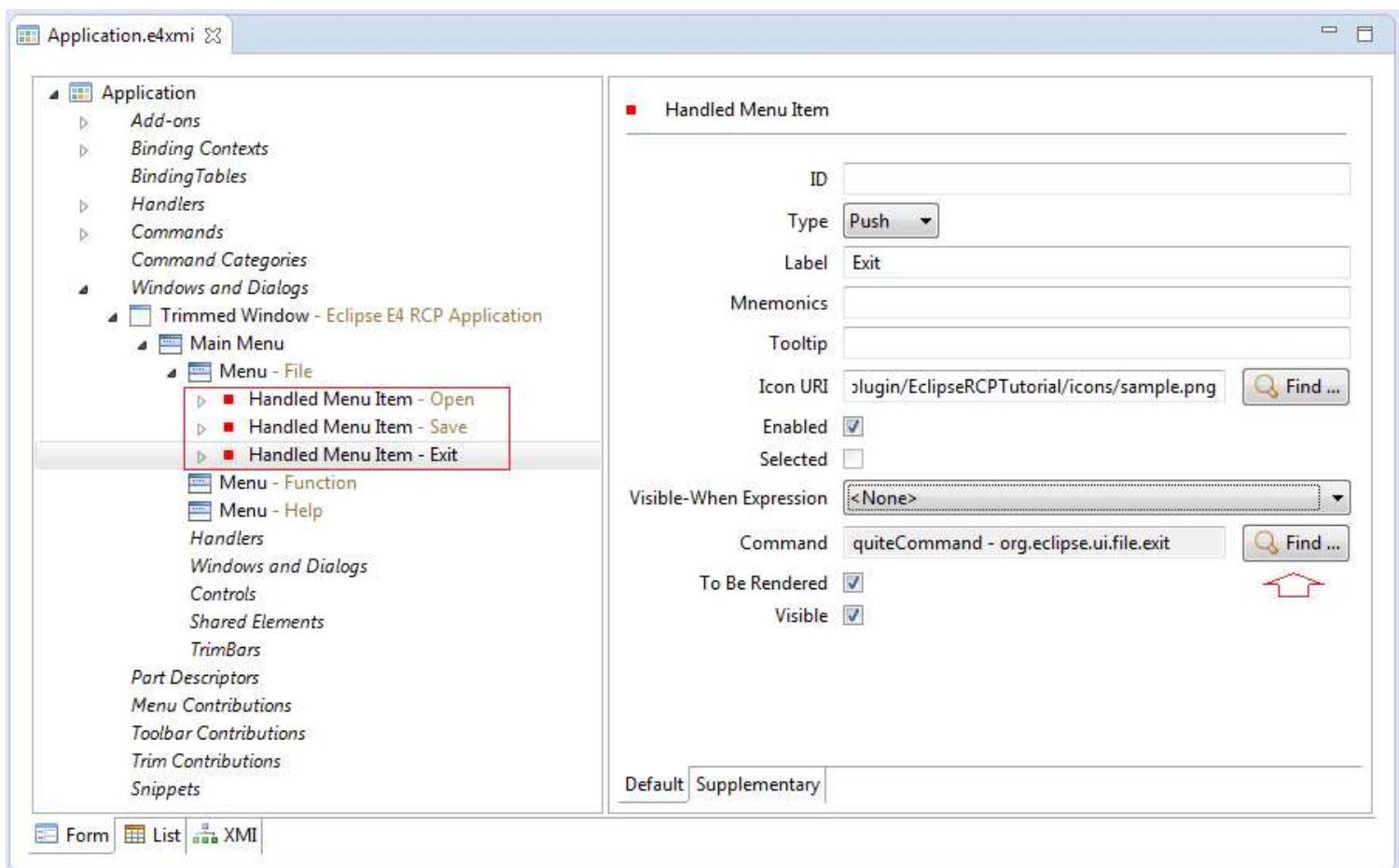
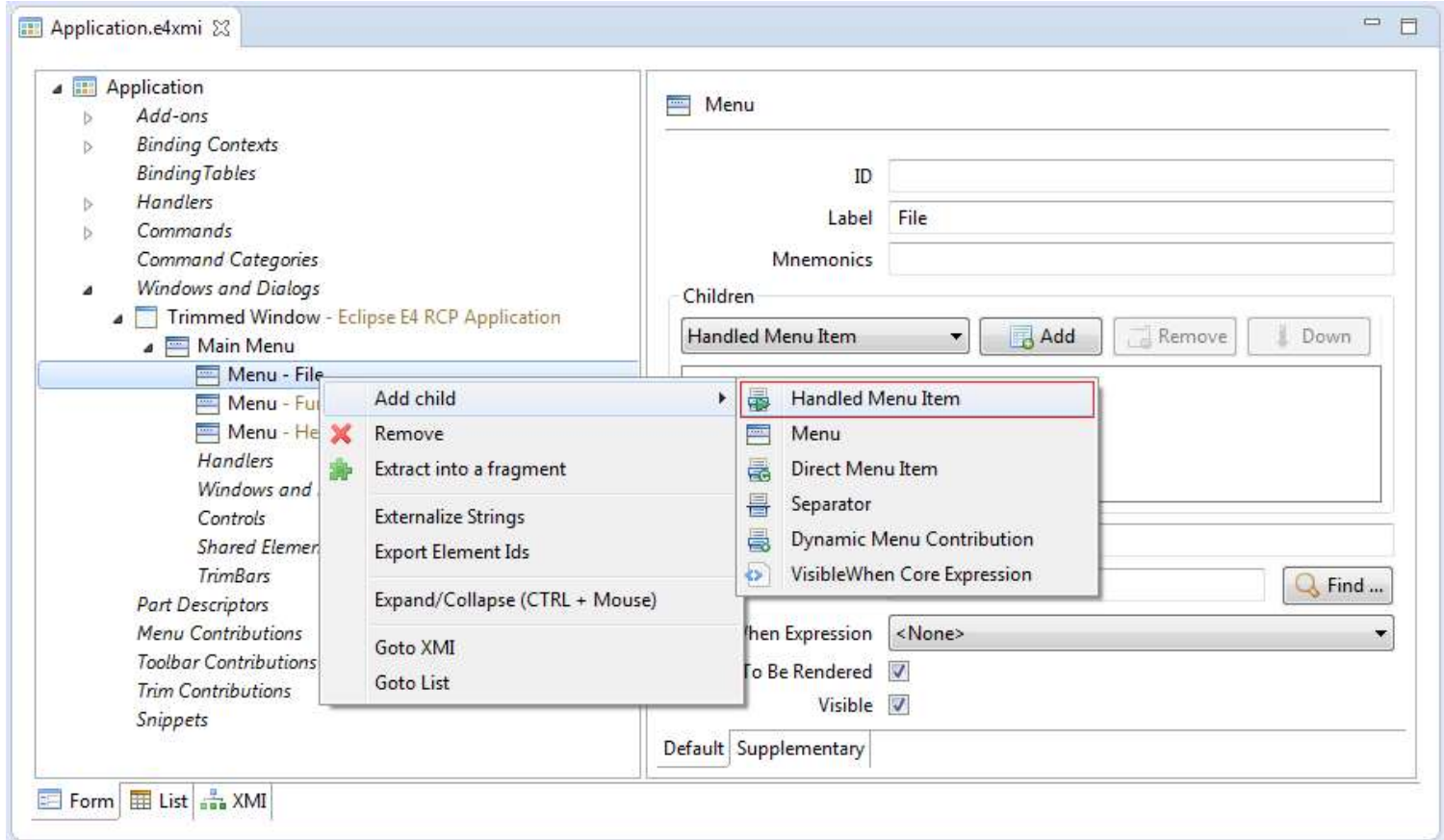




Créez des sous-menus du menu **File**

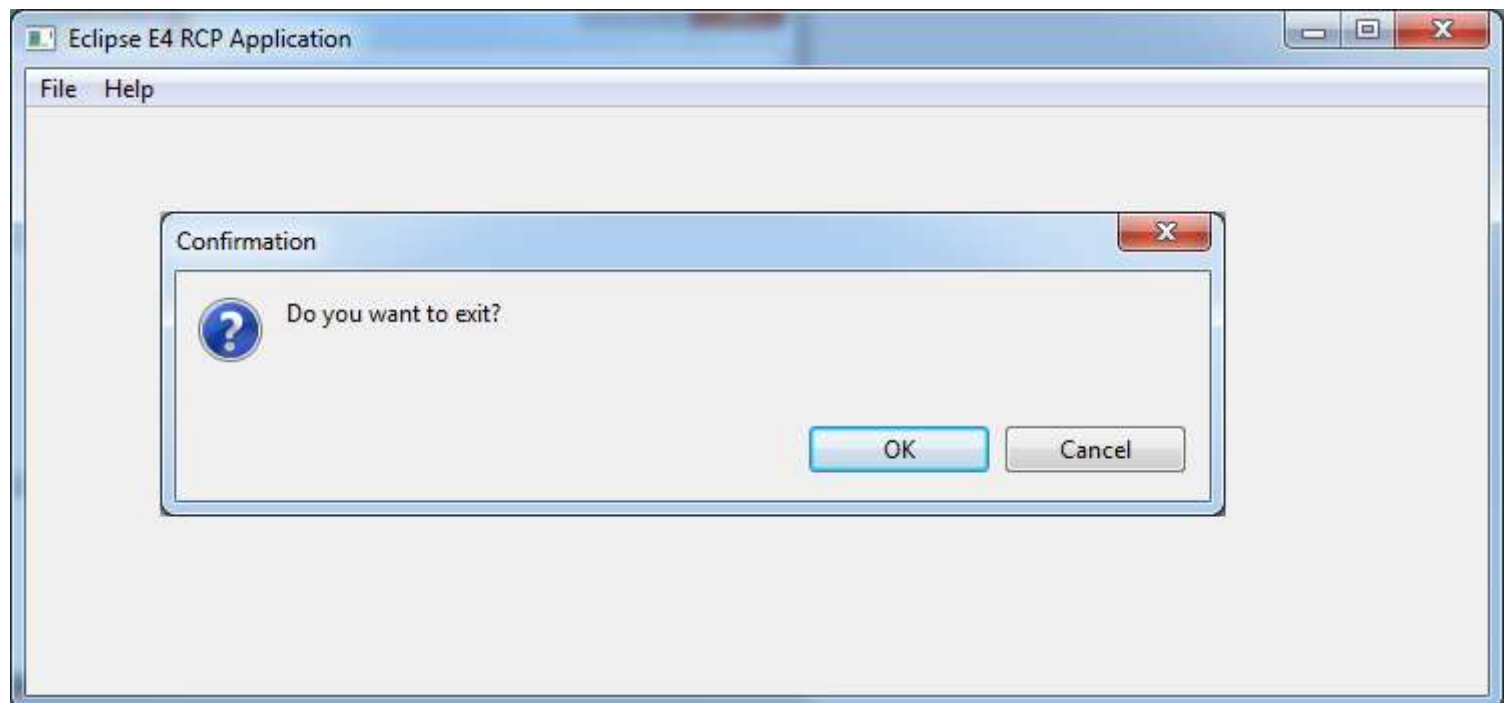
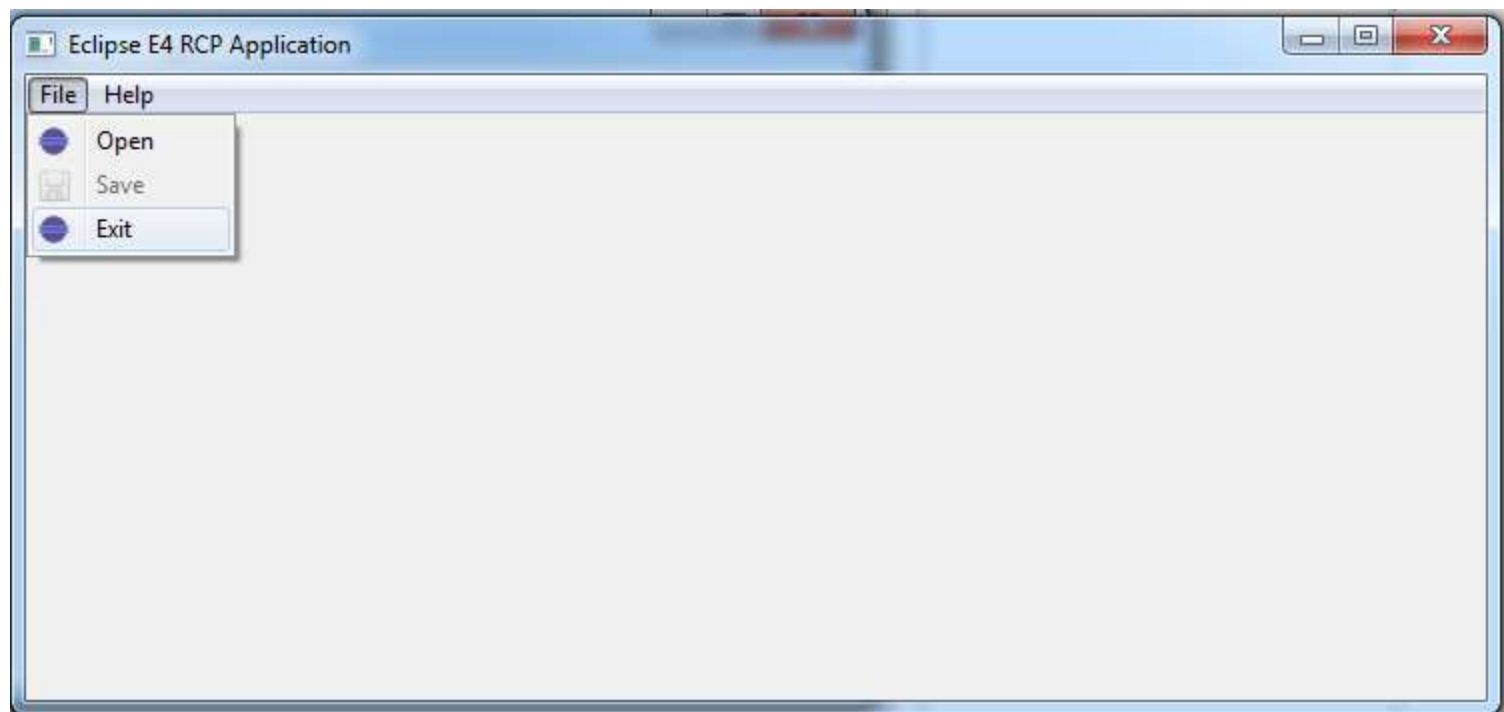
- Open
- Save
- Exit



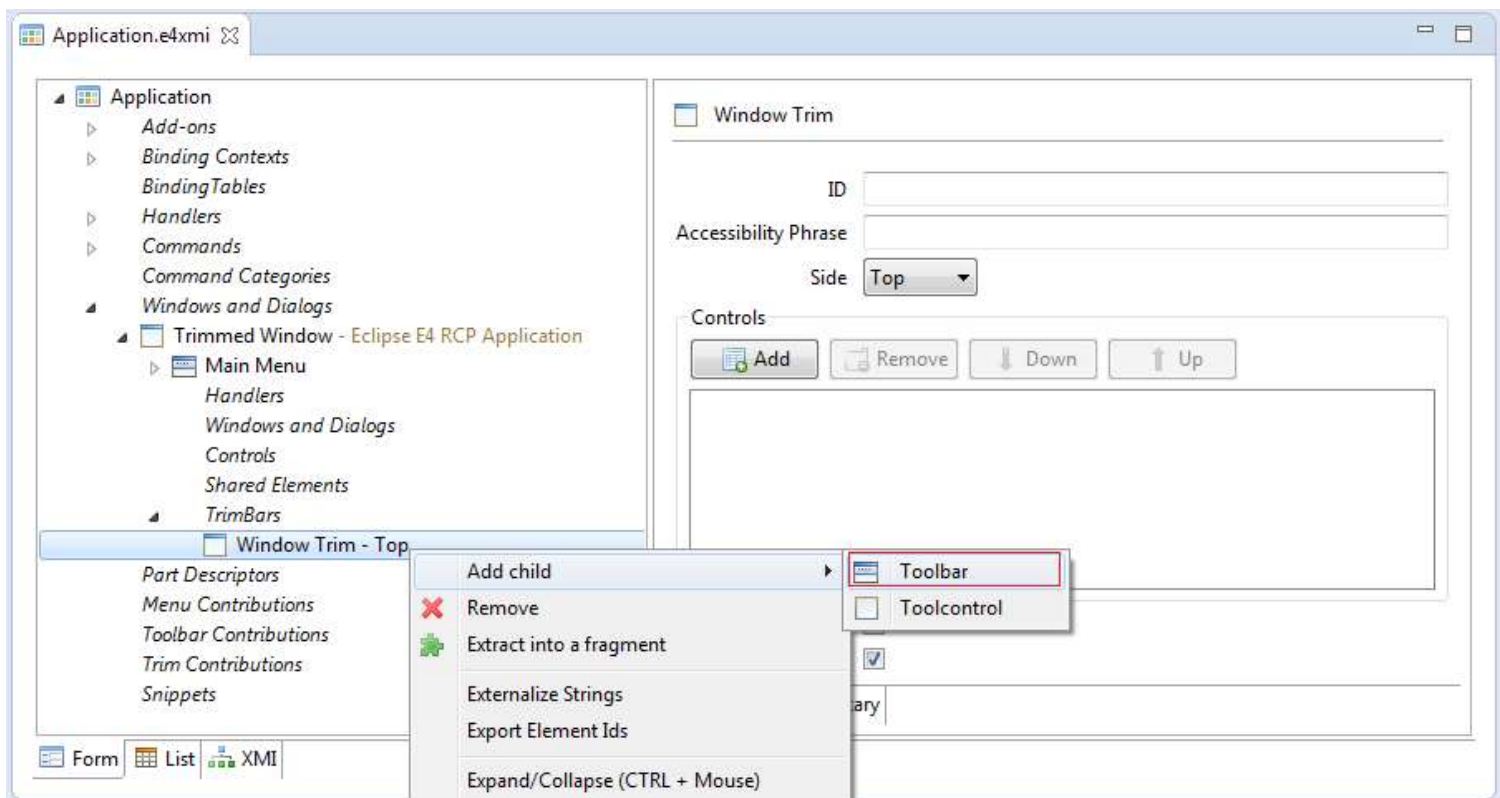
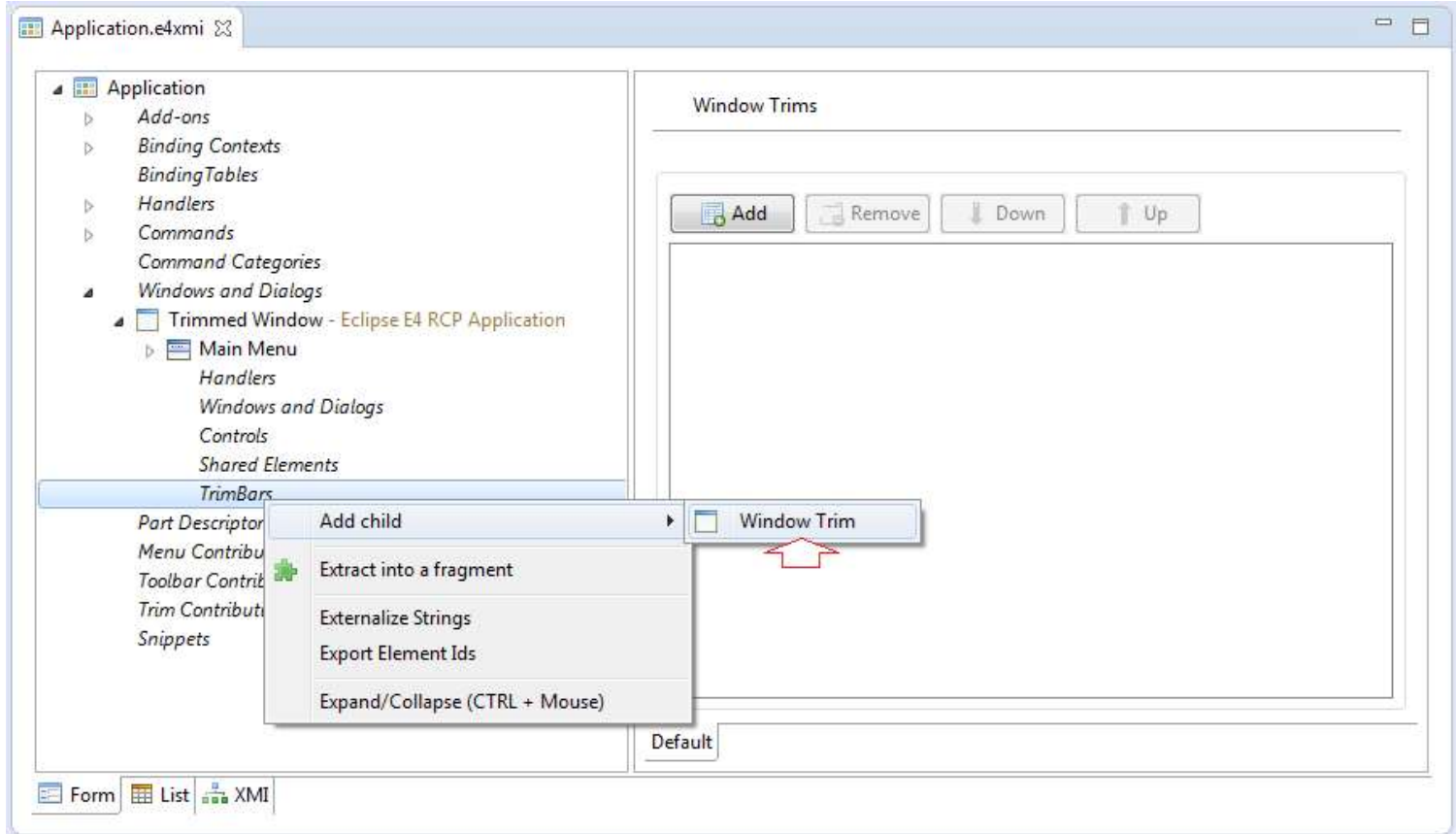


Réexécutez votre application :





7- Create Toolbar

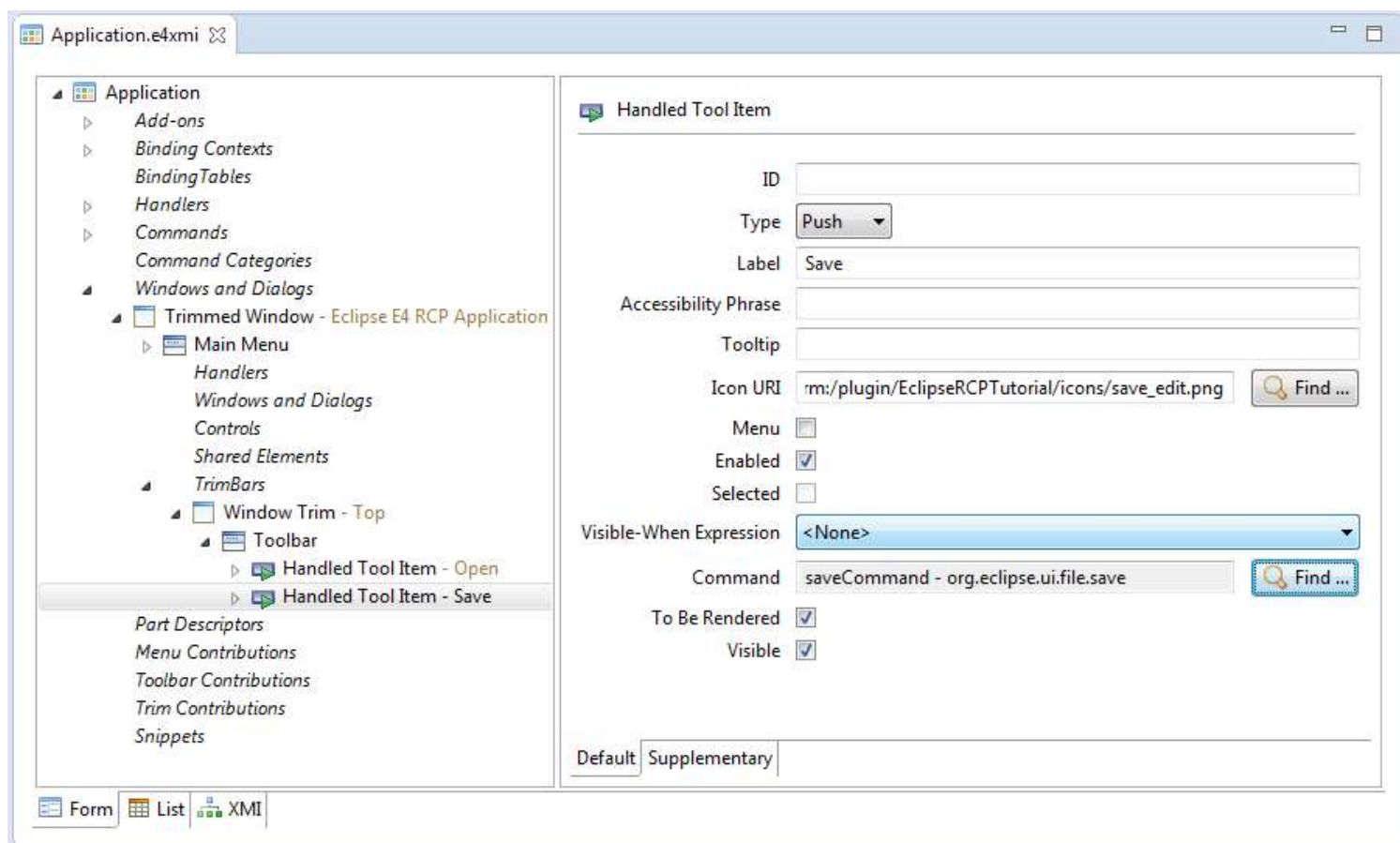
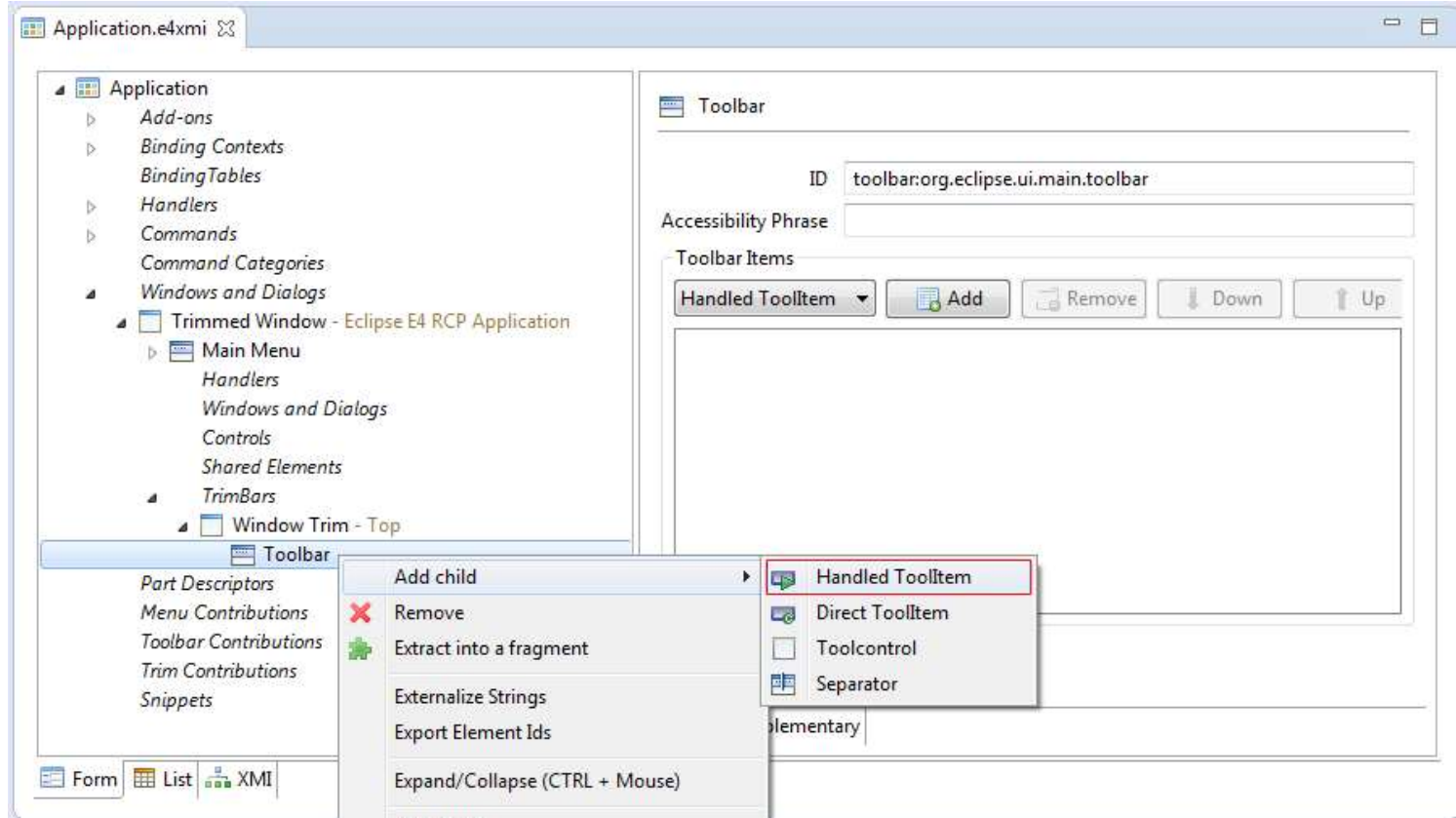


Créez Main Toolbar, avec ID :

- **ID:** toolbar:org.eclipse.ui.main.toolbar

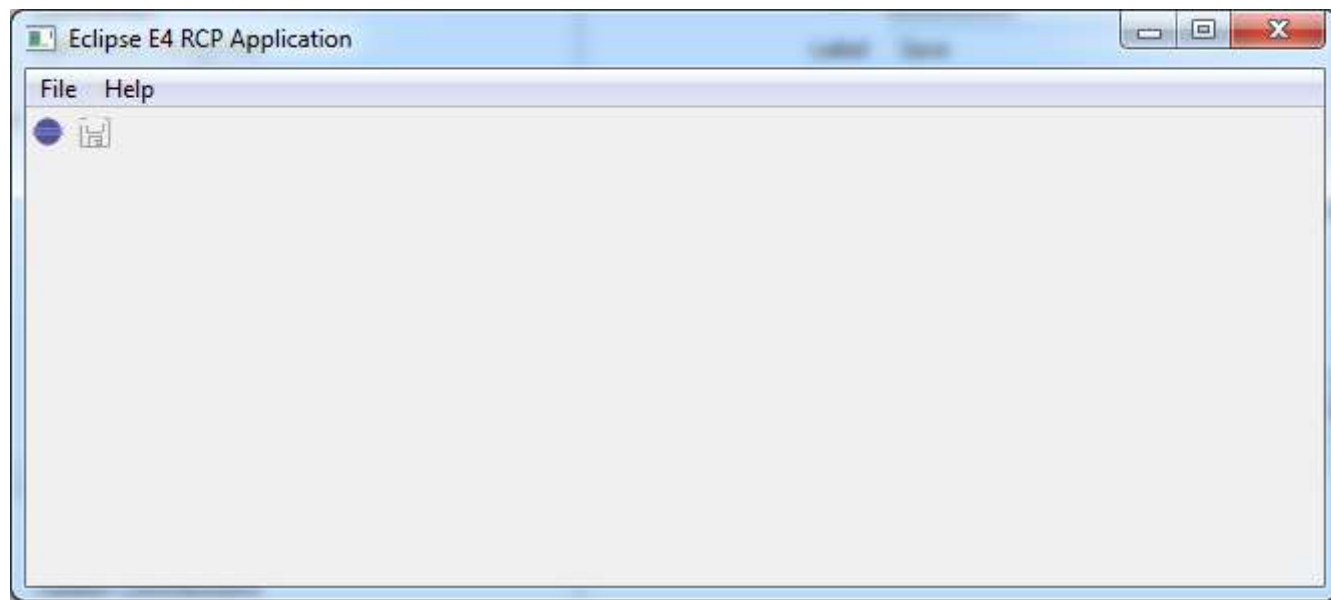
Ajoutez deux Toolitem : open & save





Relancez votre application :





8- Part

Inkl. 40 GB und jedes Jahr
10 GB mtl. ohne Aufpreis dazu*

Für nur
29,99 € mtl.*
inkl. Allnet-Flat

0 € Ans

Jetzt

SamplePart.java

```
/*****
```

```
* Copyright (c) 2010 - 2013 IBM Corporation and others.  
* All rights reserved. This program and the accompanying materials  
* are made available under the terms of the Eclipse Public License v1.0  
* which accompanies this distribution, and is available at  
* http://www.eclipse.org/legal/epl-v10.html  
*
```

```
* Contributors:
```

```
*   IBM Corporation - initial API and implementation  
*   Lars Vogel <lars.Vogel@gmail.com> - Bug 419770
```

```
*****/
```

```
package org.o7planning.tutorial.rcp.part;
```

```
import javax.annotation.PostConstruct;
```

```
import javax.inject.Inject;
```

```
import org.eclipse.e4.ui.di.Focus;
```

```
import org.eclipse.e4.ui.di.Persist;
```




```
import org.eclipse.e4.ui.model.application.ui.MDirtyable;
import org.eclipse.jface.viewers.TableViewer;
import org.eclipse.swt.SWT;
import org.eclipse.swt.events.ModifyEvent;
import org.eclipse.swt.events.ModifyListener;
import org.eclipse.swt.layout.GridData;
import org.eclipse.swt.layout.GridLayout;
import org.eclipse.swt.widgets.Composite;
import org.eclipse.swt.widgets.Text;
```

```
public class SamplePart {
```

```
    private Text txtInput;
```

```
    private TableViewer tableViewer;
```

```
    @Inject
```

```
    private MDirtyable dirty;
```

```
    @PostConstruct
```

```
    public void createComposite(Composite parent) {
        parent.setLayout(new GridLayout(1, false));
```

```
        txtInput = new Text(parent, SWT.BORDER);
```

```
        txtInput.setMessage("Enter text to mark part as dirty");
```

```
        txtInput.addModifyListener(new ModifyListener() {
```

```
            @Override
```

```
            public void modifyText(ModifyEvent e) {
```

```
                dirty.setDirty(true);
```

```
            }
```

```
        });
```

```
        txtInput.setLayoutData(new GridData(GridData.FILL_HORIZONTAL));
```

```
        tableViewer = new TableViewer(parent);
```

```
        tableViewer.add("Sample item 1");
```

```
        tableViewer.add("Sample item 2");
```

```
        tableViewer.add("Sample item 3");
```

```
        tableViewer.add("Sample item 4");
```

```
        tableViewer.add("Sample item 5");
```

```
        tableViewer.getTable().setLayoutData(new GridData(GridData.FILL_BOTH));
```

```
    }
```

```
    @Focus
```

```
    public void setFocus() {
```

```
        tableViewer.getTable().setFocus();
```

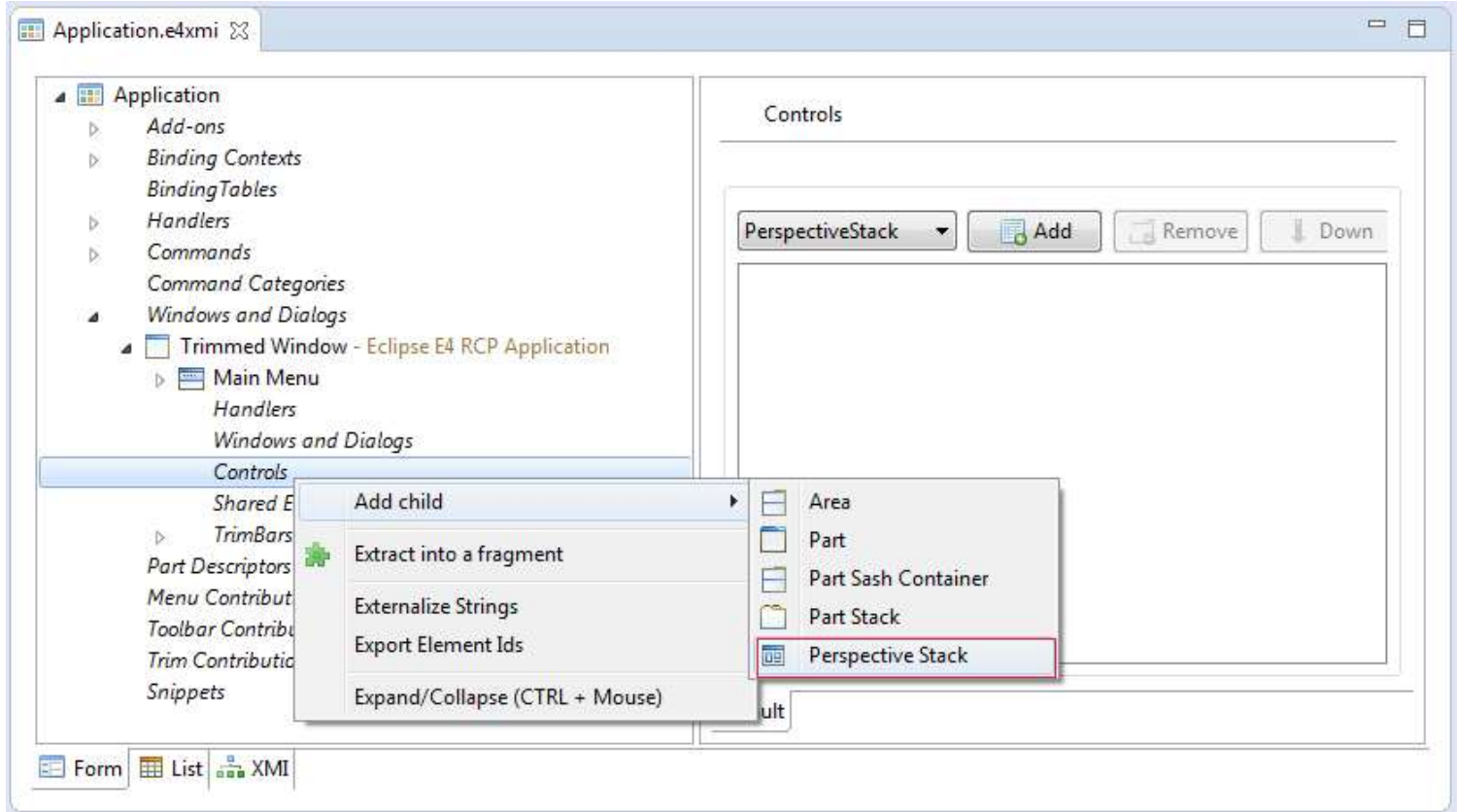
```
    }
```



@Persist

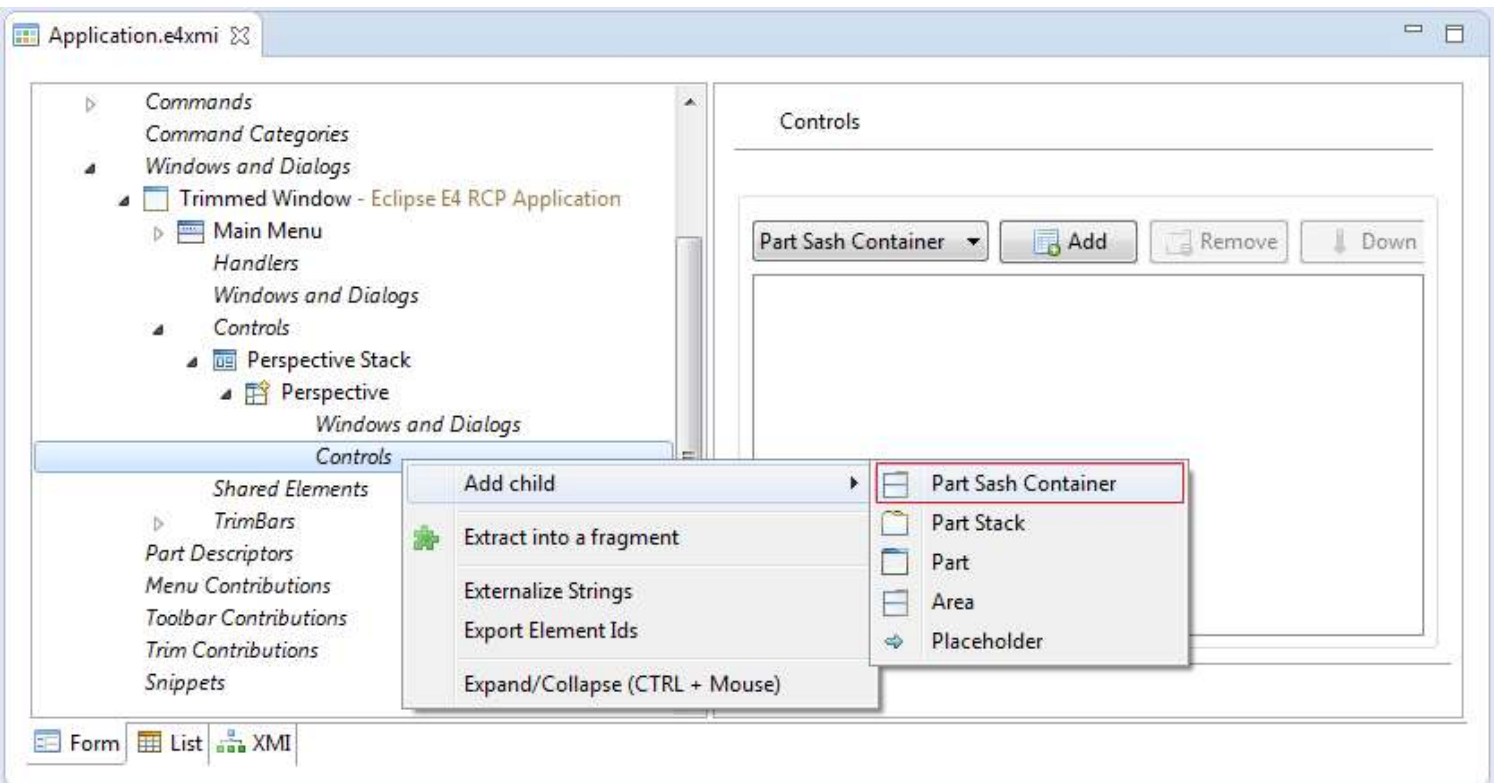
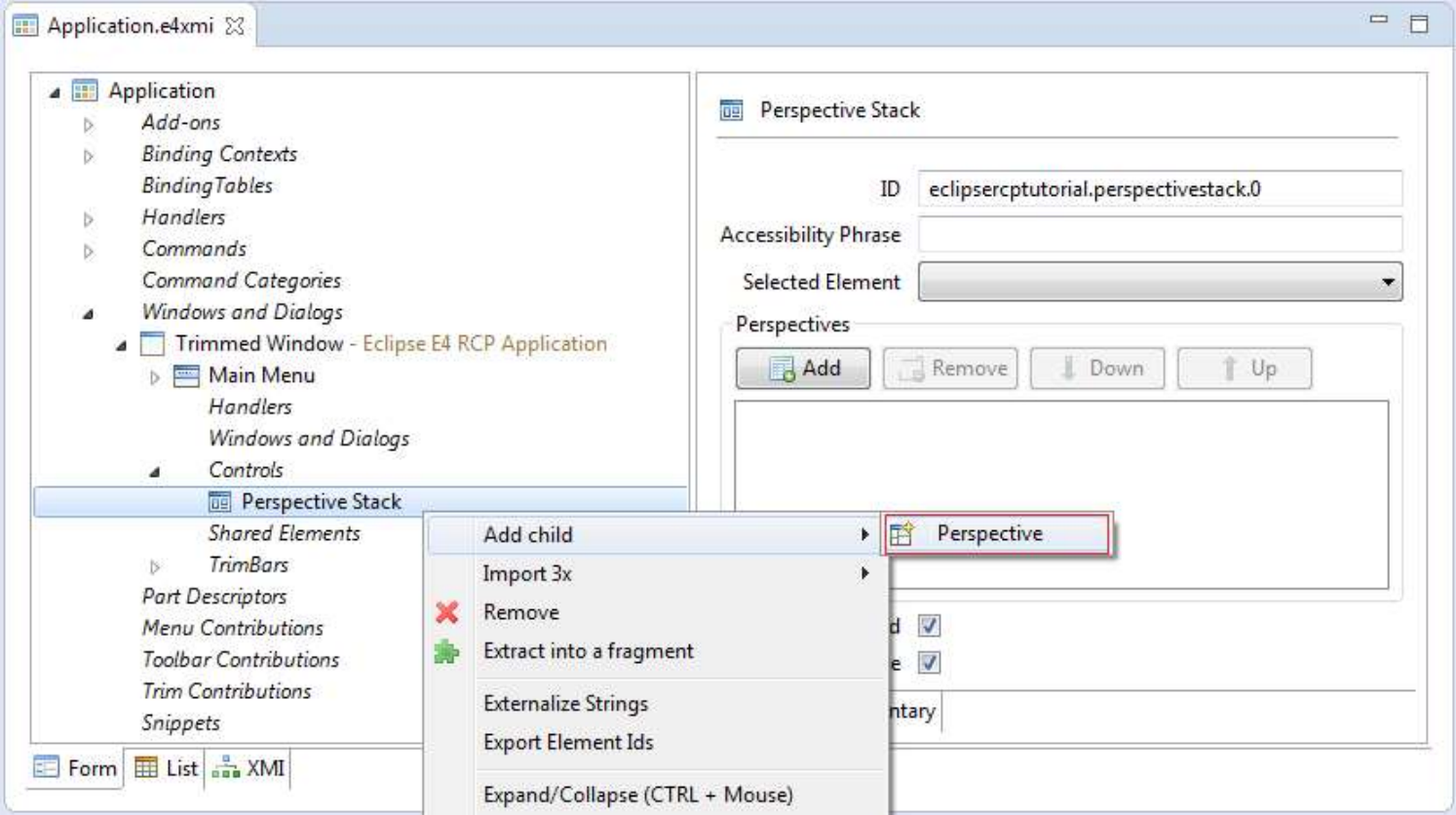
```
public void save() {  
    dirty.setDirty(false);  
}  
}
```

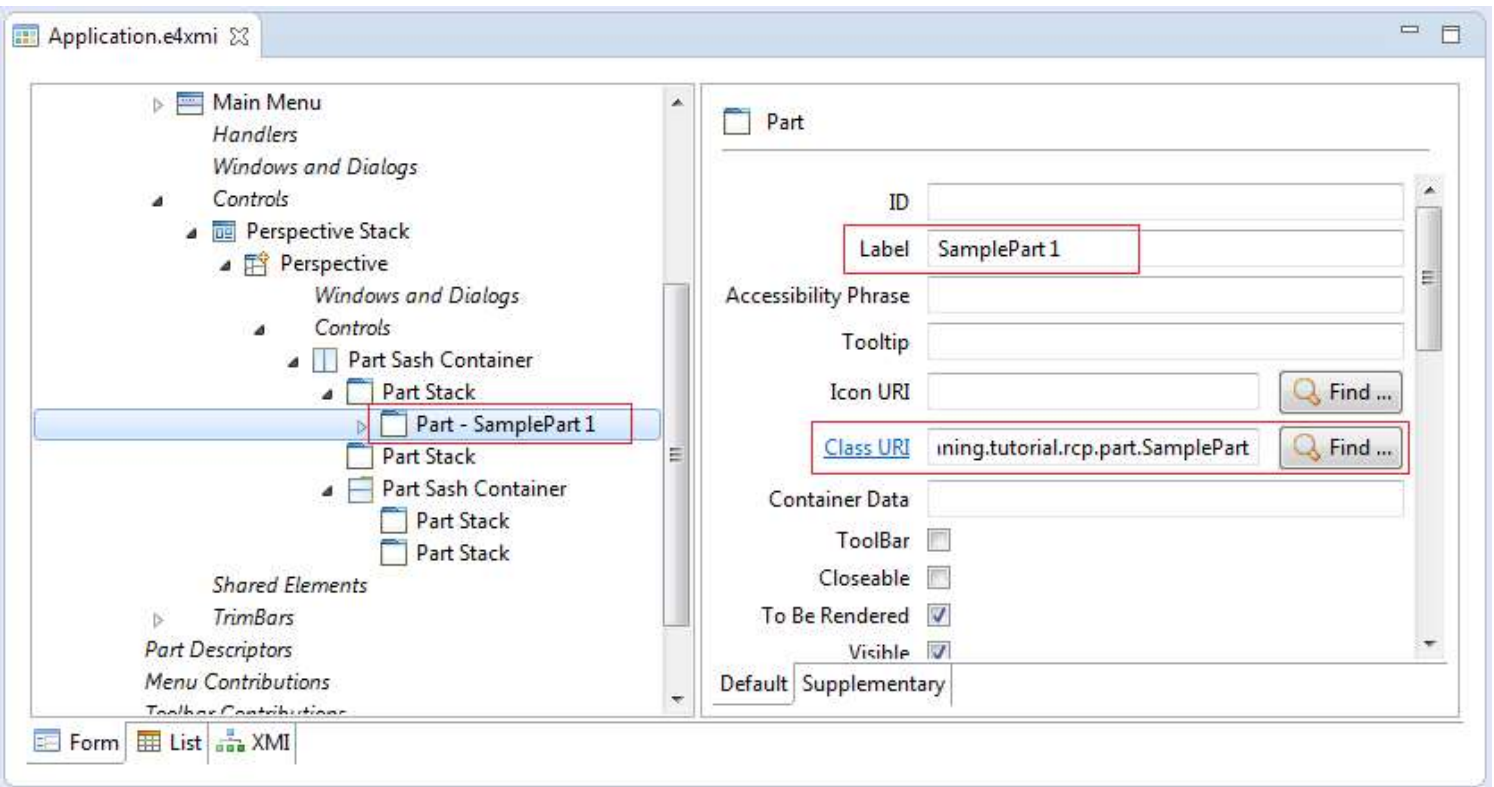
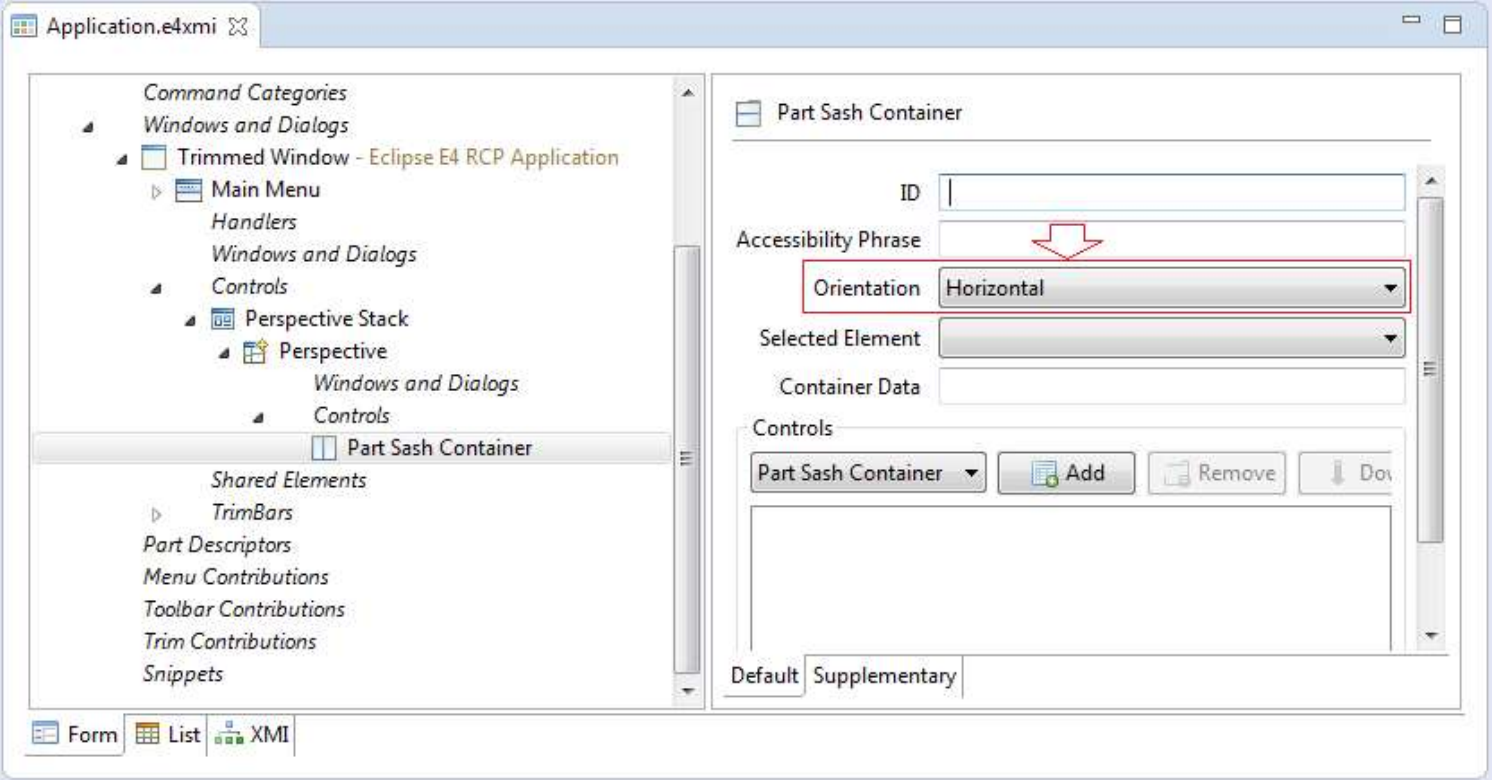
Ajoutez un nouveau **PerspectiveStack**. Ceci est un stack qui contient des perspectives.

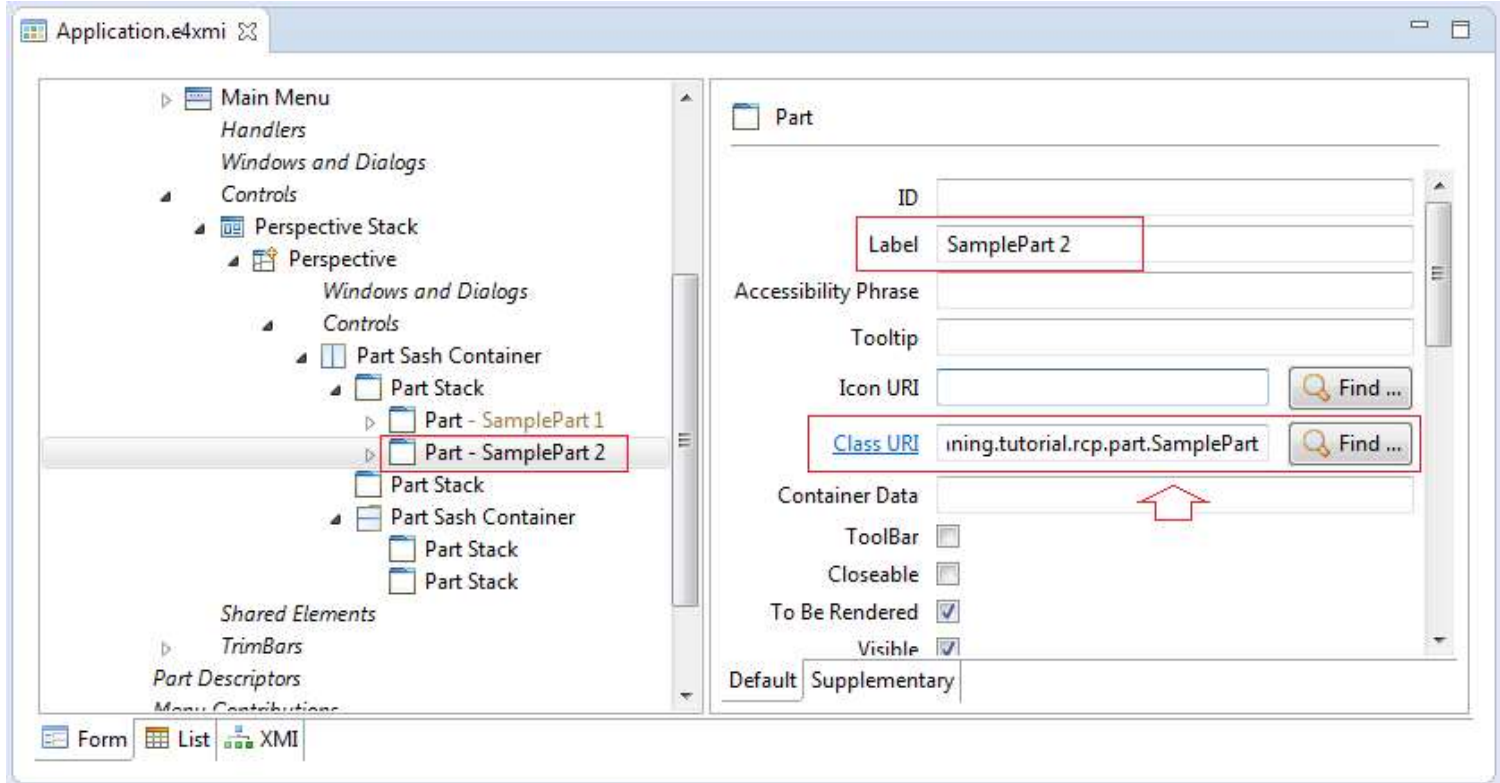


Ajoutez une nouvelle perspective :

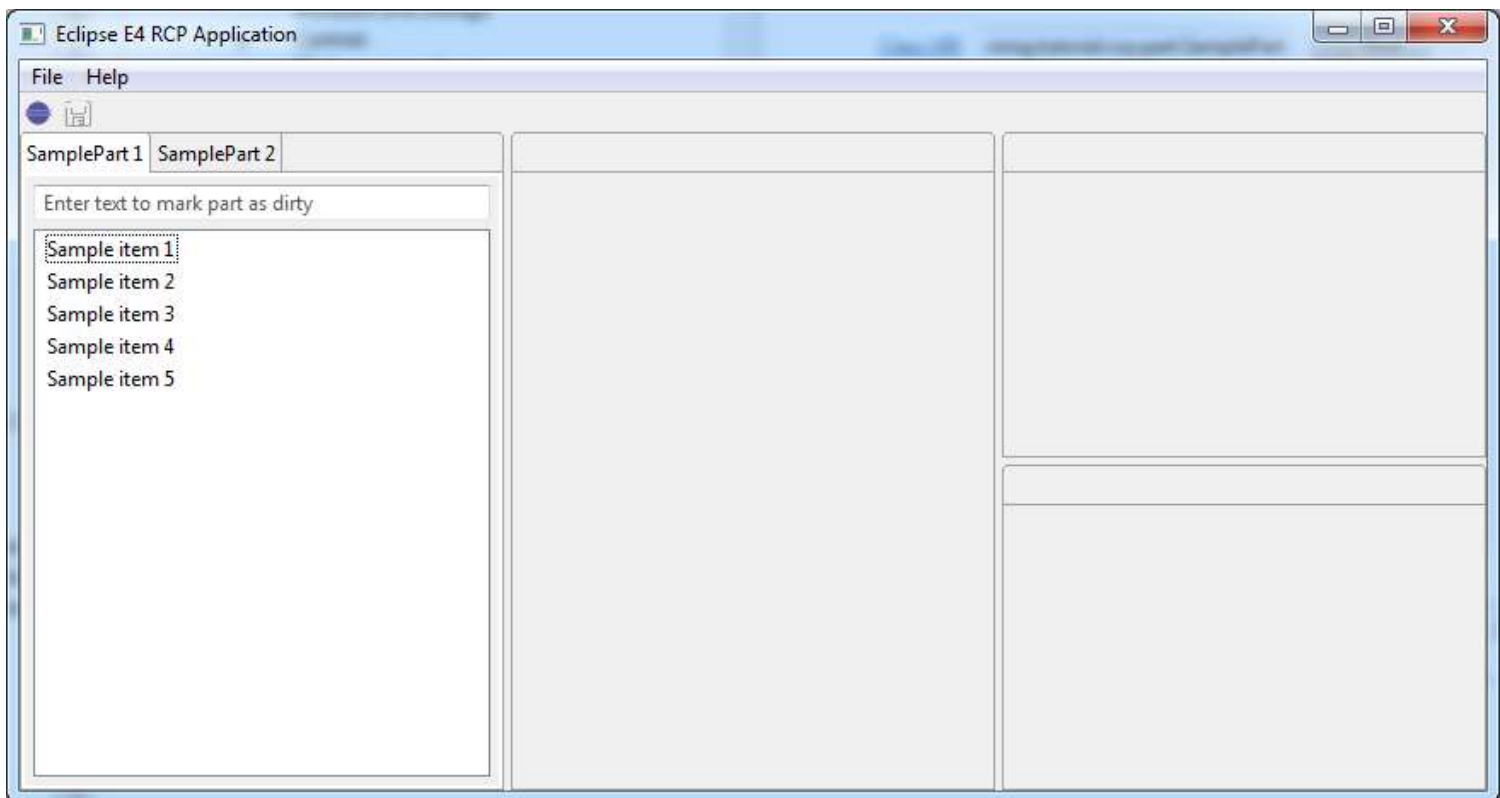






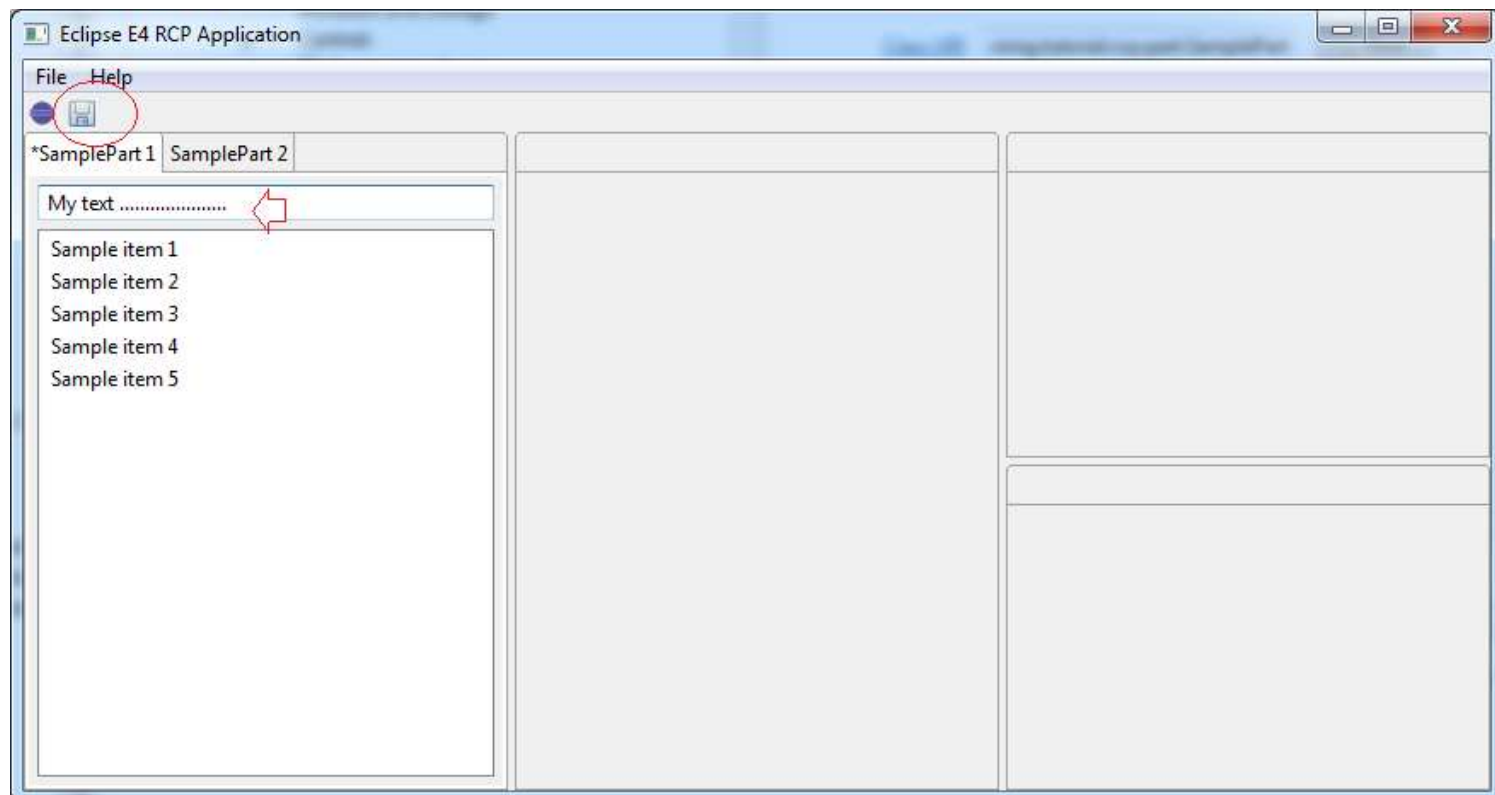


Réexécutez l'application :



Les changements sur Part pourraient faire le bouton *Save* clair














View more Tutorials:

Tutoriels Eclipse Technology

































Eclipse RCP

Peut-être que vous êtes intéressé

Voici des leçons en ligne à part du site web o7planning que nous recommandons. La liste comprend des leçons en ligne et celles en promo.

-  Udemy Professional Ruby on Rails Developer with Rails 5
-  Udemy Struts 2 Framework for Beginners
-  Udemy iOS 11 App Development: Create a Spotify Clone
-  Udemy Advanced Design Patterns with React
-  Udemy Swift - Learn by doing
-  Udemy Starting 2D Game Development in Unity with C#
-  Udemy The Complete Python & PostgreSQL Developer Course
-  Udemy Créer Votre Premier Site Responsive avec le HTML5 et le CSS3
-  Udemy Mobile E-Commerce with Flutter, Redux, and Stripe



-   Learn Bootstrap By Building Projects - Includes Bootstrap 4
-   Servlets and JSPs Tutorial: Learn Web Applications With Java
-   The Complete HTML5 & CSS3 Course Build Professional Websites
-   An Introduction to C# for the Rest of Us
-   AngularJs with Bootstrap in Action
-   Microservices with Spring Cloud
-   Bootstrap 4.0 In Action: Build A Responsive Website
-   jQuery 2018 and beyond course
-   Oracle SQL - A Complete Introduction
-   Learn SSRS SQL Reporting & Business Intelligence Essentials
-   Create Android and iOS App using HTML, CSS and JS
-   Mastering Thymeleaf with Spring Boot
-   Python 3000: Tactical File I/O
-   JavaScript Promises: Applications in ES6 and AngularJS
-   Deploying Cloud-native AWS components
-   iOS11 & Swift 4 - Le Cours Complet



Websites

-  o7planning.org
-  devstory.net
-  codestory.de
-  betacode.net
-  openplanning.net

About Us

Le site Web a été créé en mars 2014 par un groupe de programmeurs et d'auteurs du Vietnam. Actuellement, le projet prend en charge 5 langues, dont l'anglais, le français, l'allemand, le russe et le vietnamien

DMCA PROTECTED

