# Installation de l'environnement pour développer en JavaFX

Pas de contrainte 32bits ou 64bits (par contre, il faut prendre tous les composants en cohérence avec l'architecture choisie compatible avec votre matériel, soit tout en 32bits, soit tout en 64bits).

## 1. Installation de Java

**Java 8 et Java 17 (ou 21)**

Télécharger les deux JDK 8 et JDK 17 (ou 21). Attention à bien prendre des JDK et pas des JRE.

Si Java 8 (aussi appelée Java 1.8) et Java 17 sont déjà installés, on peut passer directement au paragraphe suivant.

Java 8 : https://www.oracle.com/java/technologies/javase-jdk8-downloads.html

Java 17 : https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html

Java 21 : https://www.oracle.com/java/technologies/downloads/#jdk21-windows

La dernière version en 1.8 étant la JDK 8u333.

La dernière version en 17 est la JDK 17.0.9 (une version précédente de JDK 17 convient aussi).

A noter qu'il faut un compte Oracle (si vous êtes embêtés sur ce point, je peux vous mettre à disposition les JDK 8, 17 ou 21).

Vous pouvez aussi utiliser OpenJDK 17 : https://openjdk.org/projects/jdk/17/ https://jdk.java.net/java-se-ri/17

Ou bien OpenJDK 21 : https://openjdk.org/projects/jdk/21/ https://jdk.java.net/21/

## 2. Installation de JavaFX et de l'outil SceneBuilder

### 2.1 - SDK JavaFX

Java 8 : le plus simple, JavaFX faisant partie de la distribution JDK 8, il est donc installé en même temps.

Depuis Java 11, JavaFX n'est plus présent en standard mais doit être inclus en tant que module Open Source distinct : il faut télécharger les modules JavaFX à part, sous la forme d'un SDK, à récupérer à l'URL suivante : https://gluonhq.com/products/javafx/

Bien choisir au niveau des filtres : la version, le système d'exploitation et le type souhaité :


Télécharger JavaFX SDK et décompressez-le dans un dossier de votre choix.

Pour l'exemple, supposons que JavaFX SDK est installée dans le dossier : C:\Utils\OpenJFX\ javafx-sdk-21.0.1

https://openjfx.io/openjfx-docs/#install-javafx

## 2.2 – Scene Builder

Télécharger l'outil Scene Builder à l'adresse suivante (respecter l'architecture 32bits ou 64bits choisie au départ de l'installation :

http://gluonhq.com/products/scene-builder/

Attention de bien rester cohérent :

- Si vous utilisez **Java 8**, télécharger Scene Builder 8.5.0 pour Java 8.
- Si vous utilisez **Java 21 ou 17**, télécharger Scene Builder 21.0.0 pour Java 17 et plus.

On téléchargera les deux versions pour les étudiants.


Pour Windows, je choisis le Windows Installer. Ensuite lancer simplement l'une des 2 installations de Scene Builder (8.5.0 ou 21.0.0).

Prendre soin de bien retenir les chemins d'installation (on en a besoin par la suite, cf. paragraphe suivant).

# 3. IDE Eclipse

**IDE Eclipse :**

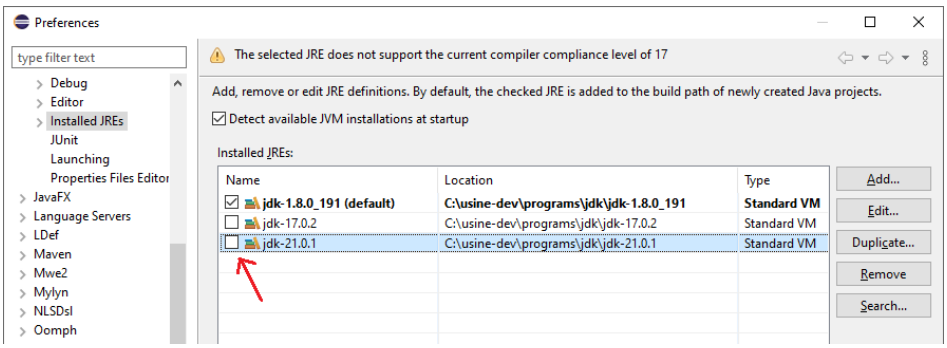
Une fois Java 8, 17 (ou 21) installée(s), il faut télécharger un IDE Eclipse à jour (si c'est déjà le cas, on peut passer au paragraphe suivant).

https://www.eclipse.org/downloads/

Il faut choisir le package « Eclipse IDE for Java Developers », au minimum Eclipse IDE version 2021-12 (4.22), la dernière étant la **2023-09 (4.29)**.

Une fois installé, lancer Eclipse et vérifier que les JDK 1.8 ou 17 ou 21 sont bien reconnus par Eclipse :

Ouvrir le menu Window> Preferences, puis sélectionner à gauche : Java > Installed JREs.



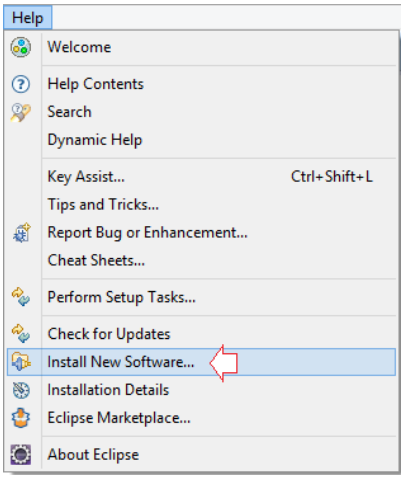

Si ce n'est pas le cas, il faudra lui indiquer manuellement via le bouton Add…

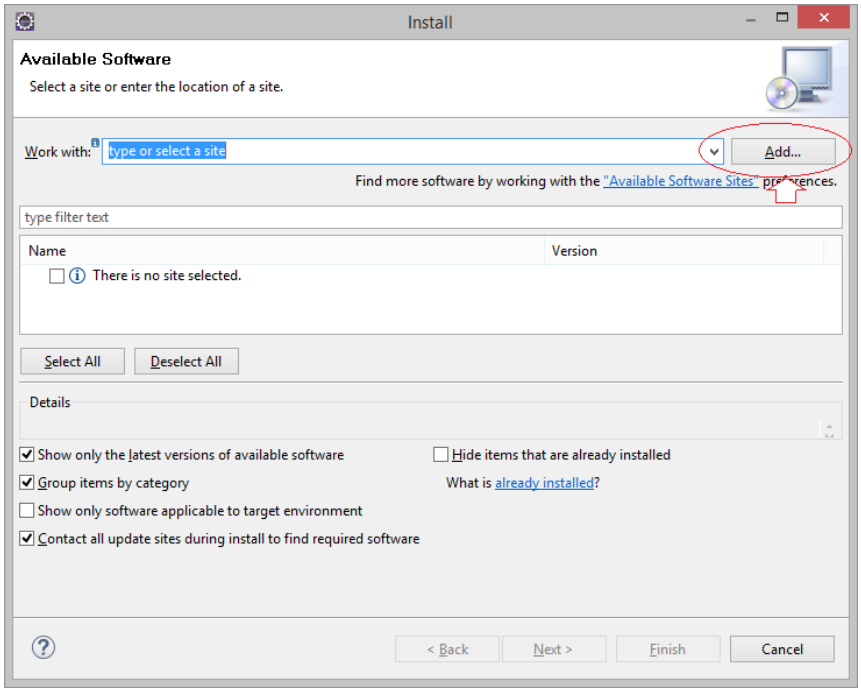
**Sélectionner la JDK 8 (ou alors la JDK 17 ou 21) par défaut.**

## 4. Plugin Eclipse e(fx)clipse

Ensuite il faut installer un plugin pour Eclipse nommé : **e(fx)clipse**.

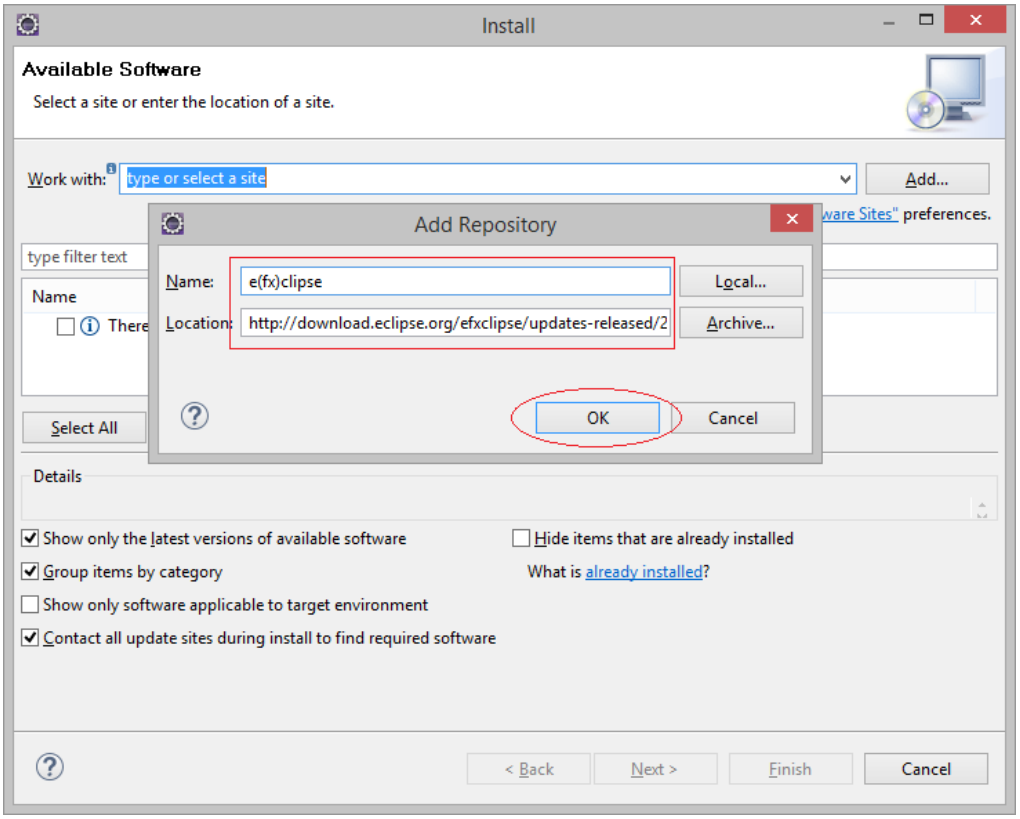
Dans le menu Help > Install New Software… Dans la fenêtre d'installation, cliquer sur le bouton Add…

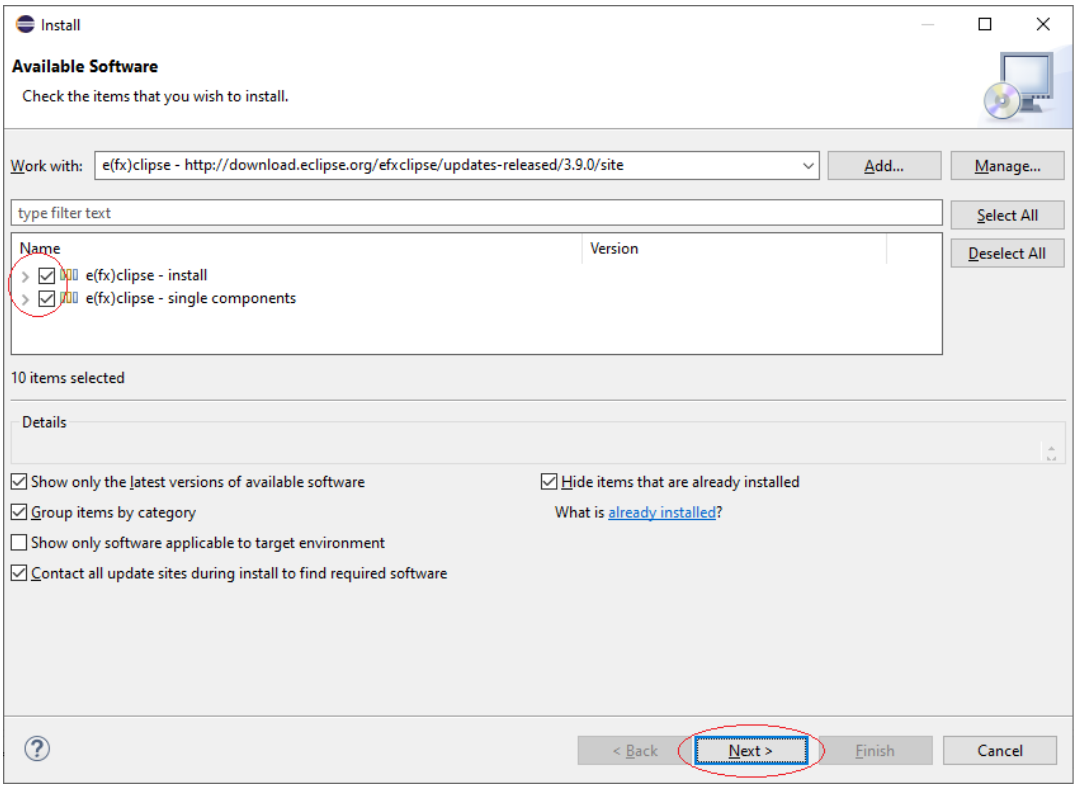


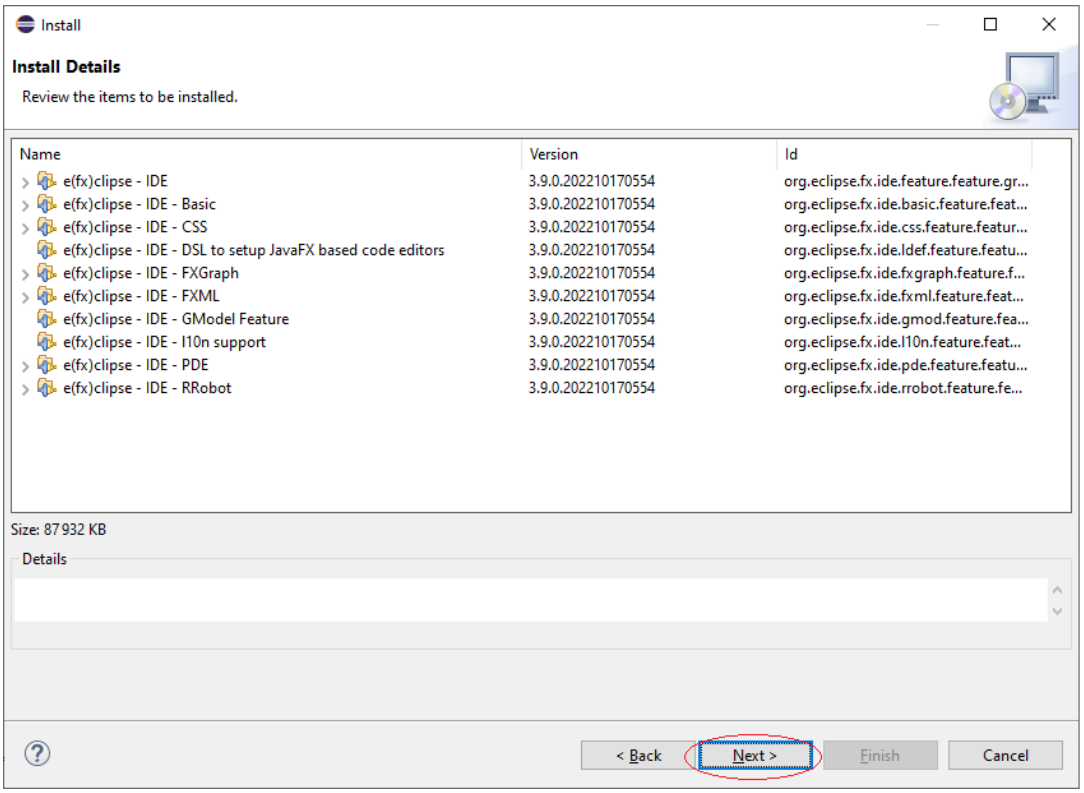




Saisir les informations suivantes :

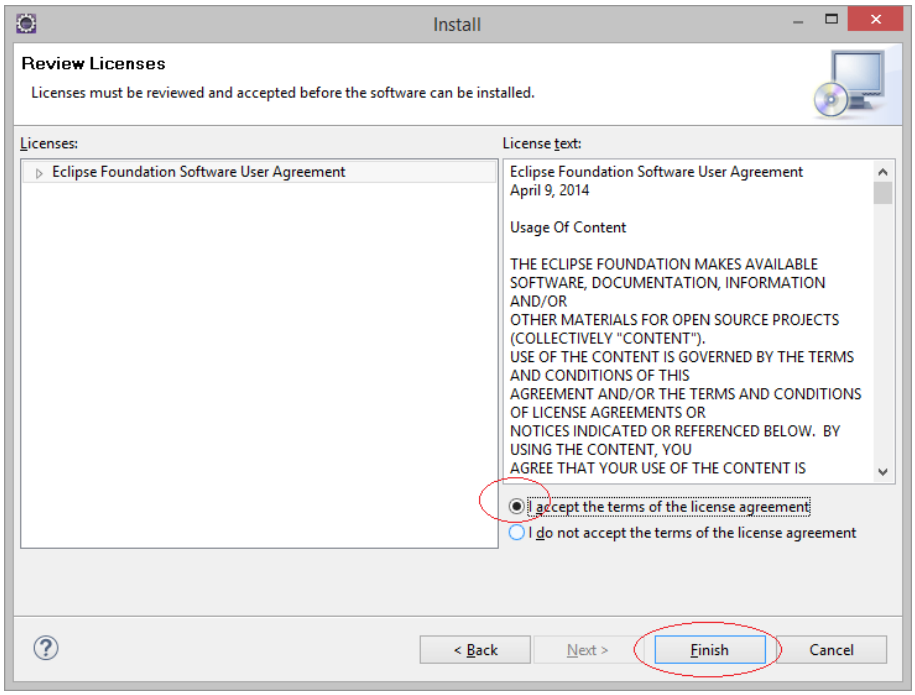
- Name : **e(fx)clipse**
- Location : http://download.eclipse.org/efxclipse/updates-released/3.9.0/site



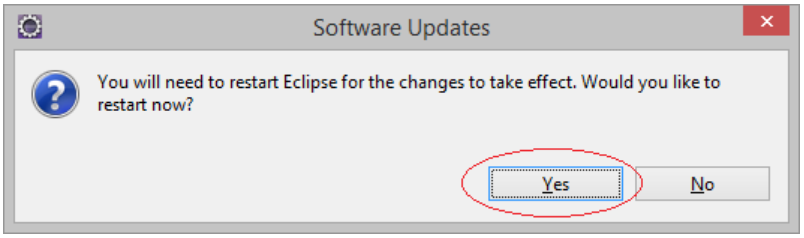

Install
Available Software
Check the items that you wish to install.
Work with:
e(fx)clipse - http://download.eclipse.org/efxclipse/updates-released/3.9.0/site
Add...
Manage...
type filter text
Select All
Deselect All
Name
Version
e(fx)clipse - install
e(fx)clipse - single components
10 items selected
Details
Show only the latest versions of available software
Group items by category
Show only software applicable to target environment
Contact all update sites during install to find required software
Hide items that are already installed
What is already installed?
< Back
Next >
Finish
Cancel


Install
Install Details
Review the items to be installed.
Name
Version
Id
e(fx)clipse - IDE 3.9.0.202210170554 org.eclipse.fx.ide.feature.feature.gr...
e(fx)clipse - IDE - Basic 3.9.0.202210170554 org.eclipse.fx.ide.basic.feature.feat...
e(fx)clipse - IDE - CSS 3.9.0.202210170554 org.eclipse.fx.ide.css.feature.featur...
e(fx)clipse - IDE - DSL to setup JavaFX based code editors 3.9.0.202210170554 org.eclipse.fx.ide.ldef.feature.featu...
e(fx)clipse - IDE - FXGraph 3.9.0.202210170554 org.eclipse.fx.ide.fxgraph.feature.f...
e(fx)clipse - IDE - FXML 3.9.0.202210170554 org.eclipse.fx.ide.fxml.feature.feat...
e(fx)clipse - IDE - GModel Feature 3.9.0.202210170554 org.eclipse.fx.ide.gmod.feature.fea...
e(fx)clipse - IDE - I10n support 3.9.0.202210170554 org.eclipse.fx.ide.l10n.feature.feat...
e(fx)clipse - IDE - PDE 3.9.0.202210170554 org.eclipse.fx.ide.pde.feature.featu...
e(fx)clipse - IDE - RRobot 3.9.0.202210170554 org.eclipse.fx.ide.rrobot.feature.fe...
Size: 87932 KB
Details
< Back
Next >
Finish
Cancel

Accepter la licence.

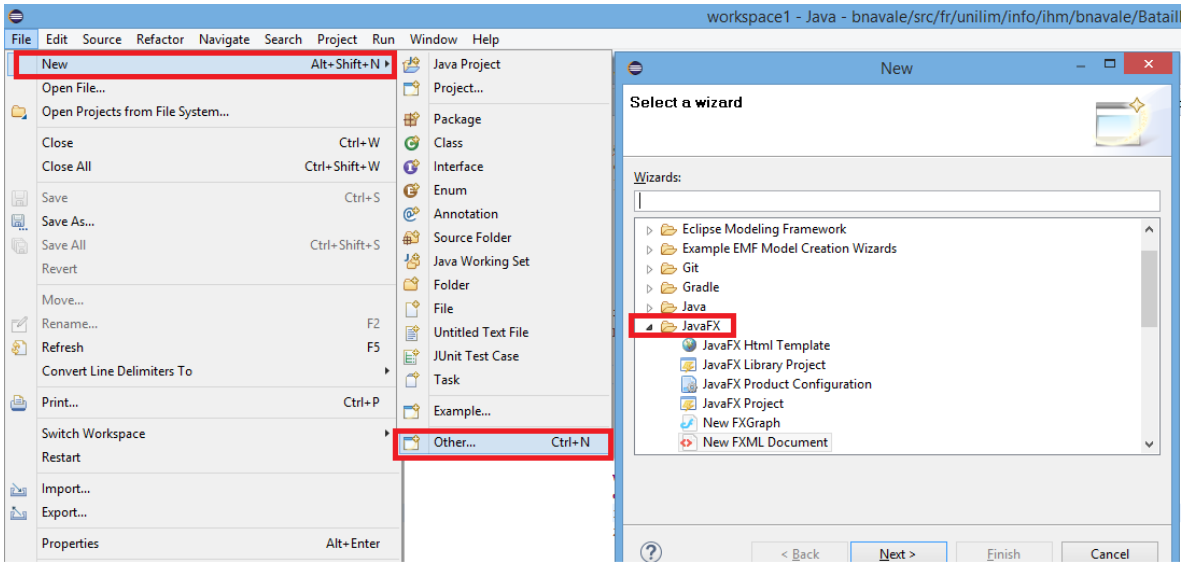



Re-démarrer Eclipse quand on vous le propose (l'installation prend quelques minutes tout de même).




Vérifier que le plugin est bien installé : cliquez sur File > New > Other...

Puis vérifier la présence de l'option : JavaFX.

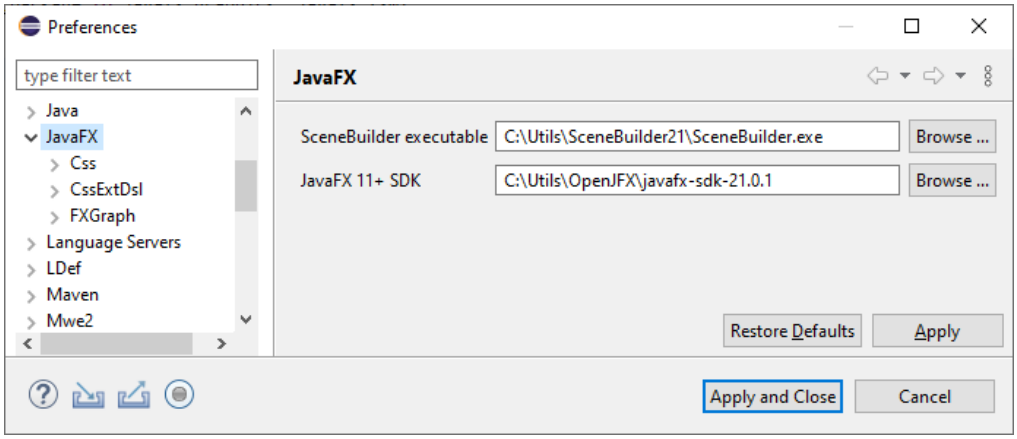



Si l'option est bien présente, cliquer sur Annuler (Cancel) pour fermer la fenêtre.

**Lier Eclipse avec JavaFX et Scene Builder**

Une fois que le SDK JavaFX et l'outil Scene Builder sont installés, depuis Eclipse, ouvrir le menu Window> Preferences > JavaFX

Indiquer les chemins de l'exécutable de SceneBuilder (chemin d'installation de Scene Builder 21), et de la SDK de JavaFX, comme suit :



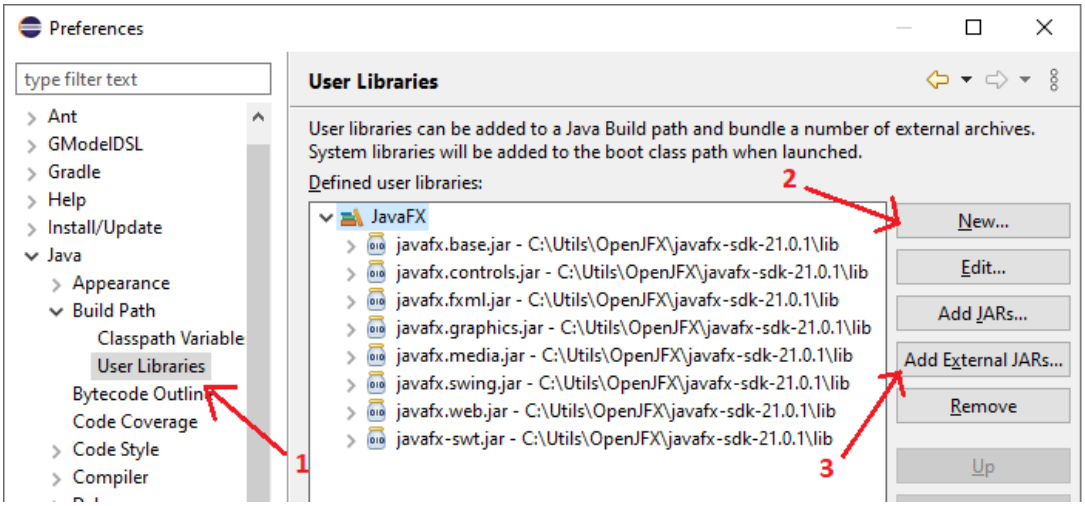

## 5. Configurer un projet JavaFX en Java 17 (ou 21) sous Eclipse

**Etape 1 : Créer une « User Library » dans Eclipse**

Ouvrir le menu Window> Preferences, puis sélectionner à gauche : Java > Build Path > User Libraries.

Cliquer "New…" pour créer une nouvelle librairie : Nommez-là JavaFX.

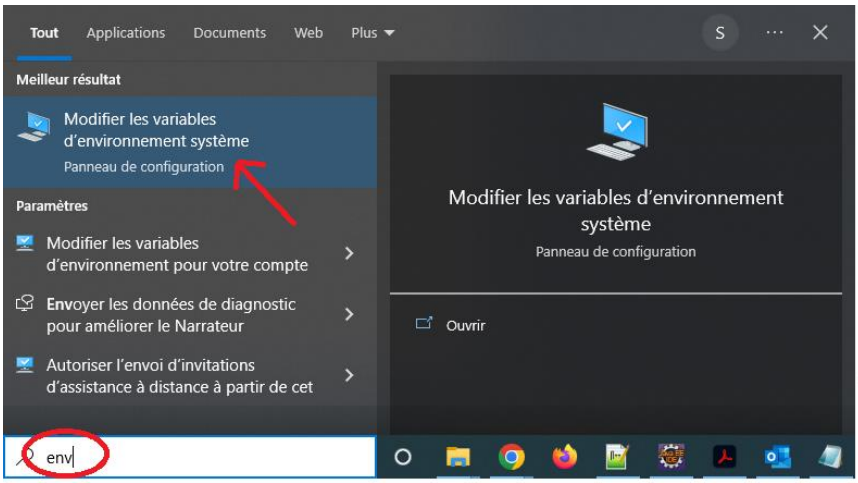
Cliquer ensuite sur "Add External JARS…" : rechercher le dossier lib de votre installation de JavaFX SDK puis sélectionner tous les jar présents.

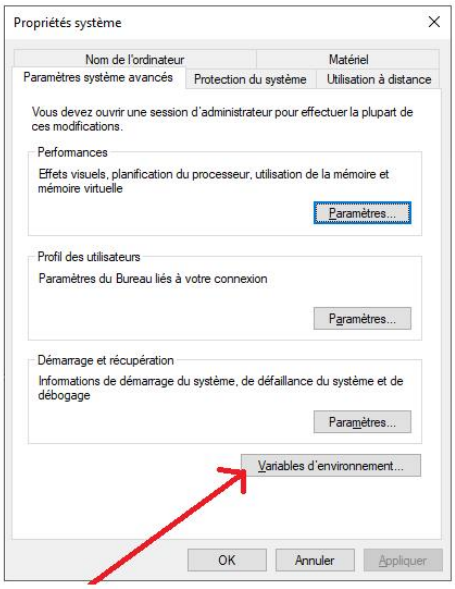


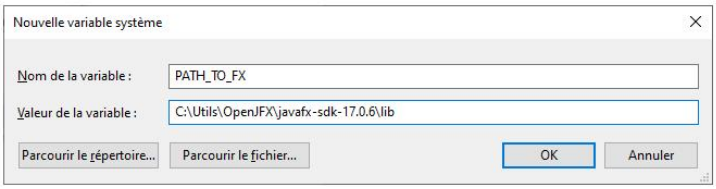

Terminer cette étape en cliquant sur "Apply and Close".

**Etape 2 (facultative) :** définir une variable d'environnement.

**Sous Windows :**




Dans notre exemple, ce serait : `set PATH_TO_FX="C:\Utils\OpenJFX\javafx-sdk-21.0.1\lib"`

**Sous Unix**, en fonction de votre shell :

- **Korn shell (ksh) :**

  ```
  PATH_TO_FX=/path/to/javafx-sdk-21.0.1/lib
  export PATH_TO_FX
  ```

- **Bourne shell (sh et bash) :**

  ```
  export PATH_TO_FX=/path/to/javafx-sdk-21.0.1/lib
  ```

- **C shell (csh ou tcsh) :**

  ```
  setenv PATH_TO_FX /path/to/javafx-sdk-21.0.1/lib
  ```

**Sous Mac :**

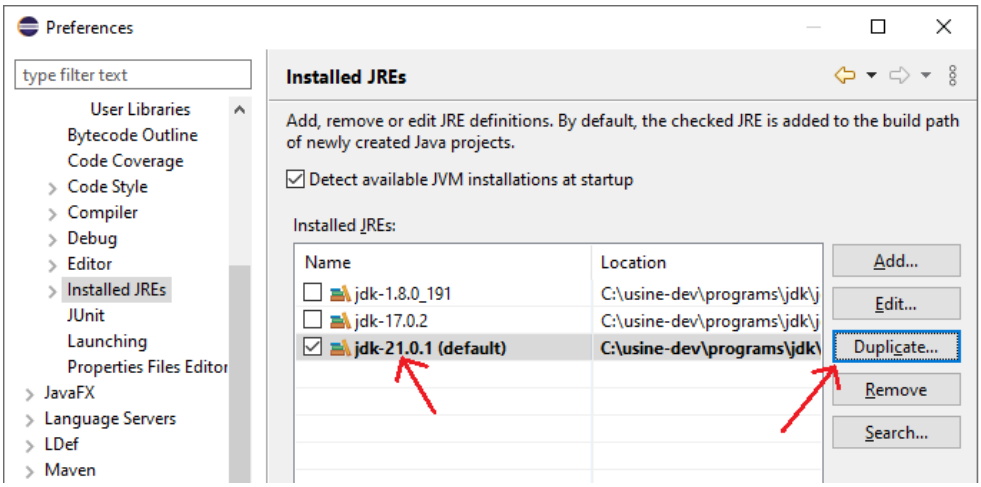
Chercher et modifier votre fichier .bash_profile :

- Chercher le fichier : `~/.bash_profile`
- Editer le fichier et rajouter à la fin : `export PATH_TO_FX=/path/to/javafx-sdk-21.0.1/lib`
- Recharger le fichier : `source ~/.bash_profile`

Redémarrer Eclipse (pour que la variable d'environnement soit bien prise en compte).

**Etape 3 :** Cloner une JRE avec les arguments requis par la VM.

Ouvrir le menu Window> Preferences, puis sélectionner à gauche : Java > Installed JREs.

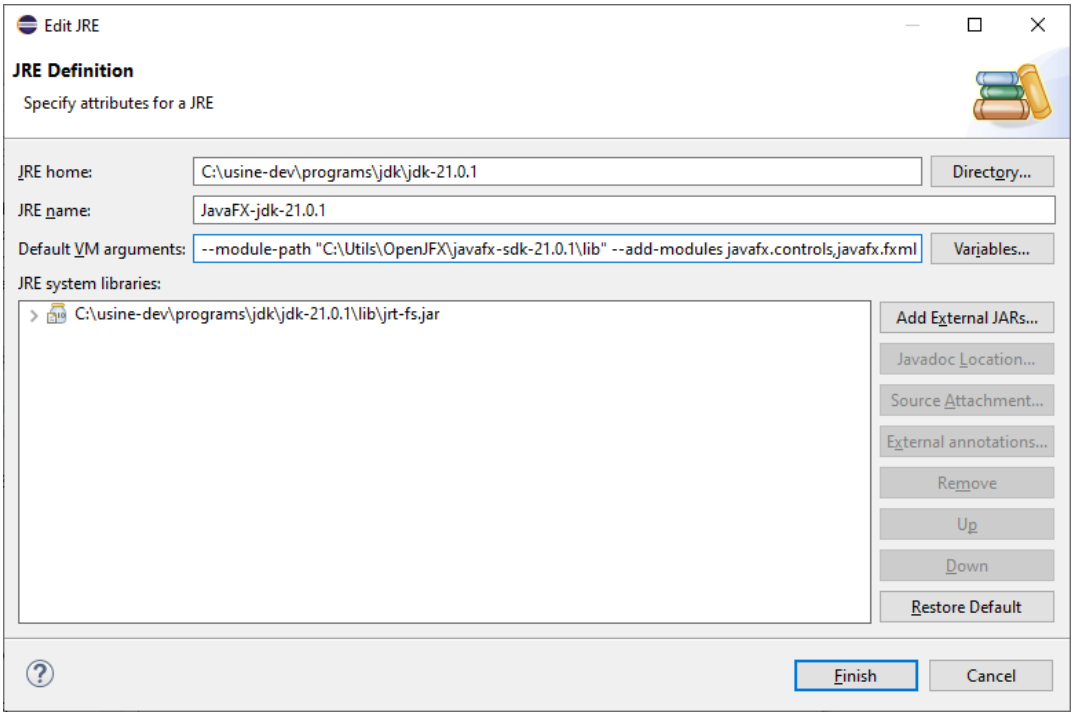



Dupliquer la JRE que vous souhaitez utiliser.

Renseigner les arguments de la VM (virtual machine) avec la ligne suivante en adaptant le chemin d'installation de JavaFX SDK (attention au copier-coller, saisisser les arguments à la main) :

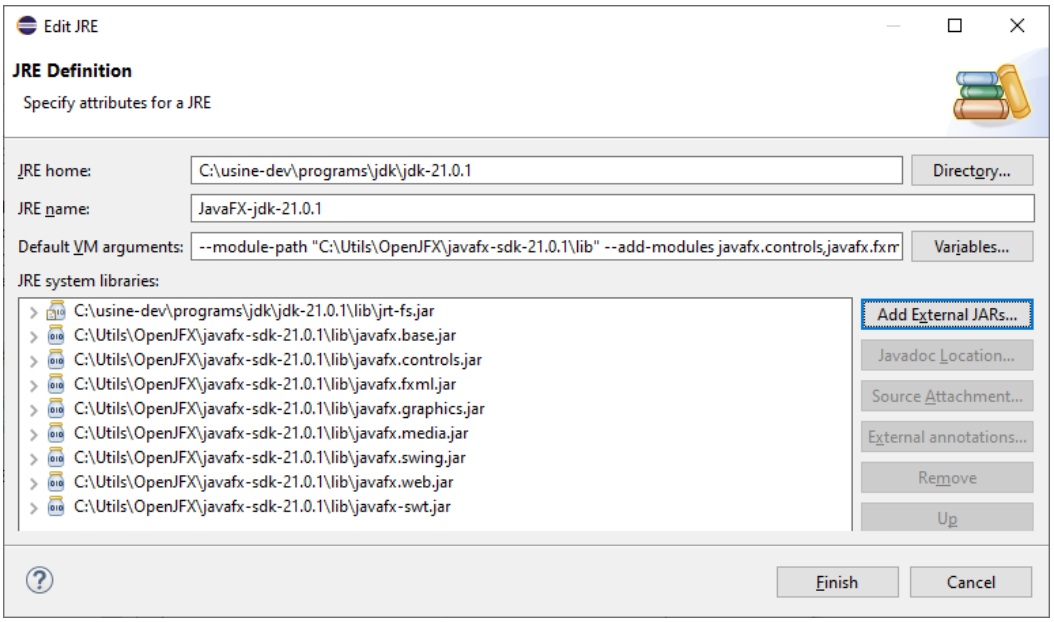
```
--module-path "\path\to\javafx-sdk-21\lib" --add-modules javafx.controls,javafx.fxml
```

Si vous avez défini une variable d'environnement, vous pouvez l'utiliser en remplaçant `"\path\to\javafx-sdk-21\lib"` par `%PATH_TO_FX%` (ou `$PATH_TO_FX` selon votre plateforme).



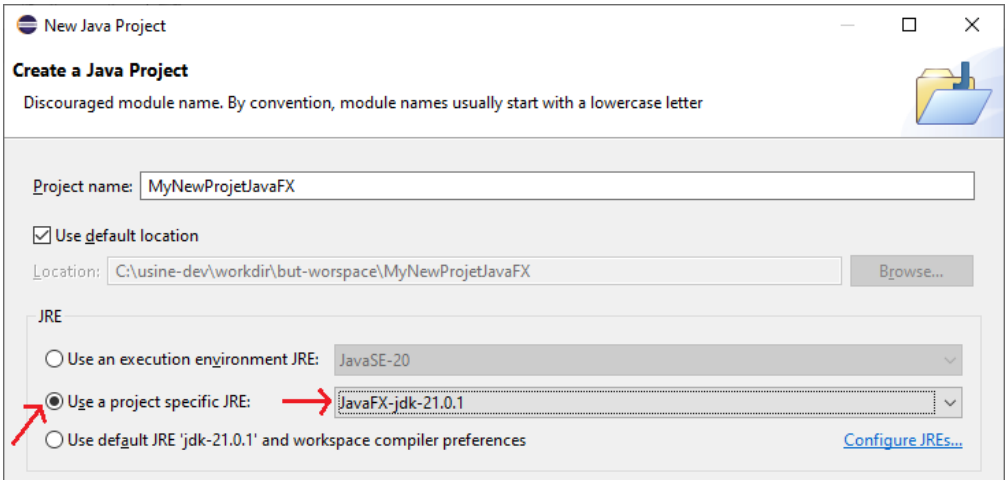

Cliquer ensuite sur "Add External JARS..." : rechercher le dossier lib de votre installation de JavaFX SDK puis sélectionner tous les jar présents.

Indiquer un nom explicite à votre JRE. La définition de votre JDK doit maintenant ressembler à ceci :




Cliquer sur Finish.

**Mis en pratique :** Créer un nouveau projet avec votre JDK clonée.




Ne pas cocher la case « Create module-info.java file » (si jamais vous l'avez cochée ce n'est pas grave, il faudra supprimer le fichier module-info.java qui aura été généré à la racine de vos sources).

Nb : si vous voulez vraiment utiliser des modules (non nécessaire et non conseillé dans le cadre de ce cours), votre fichier "module-info.java" devra ressembler à ceci :

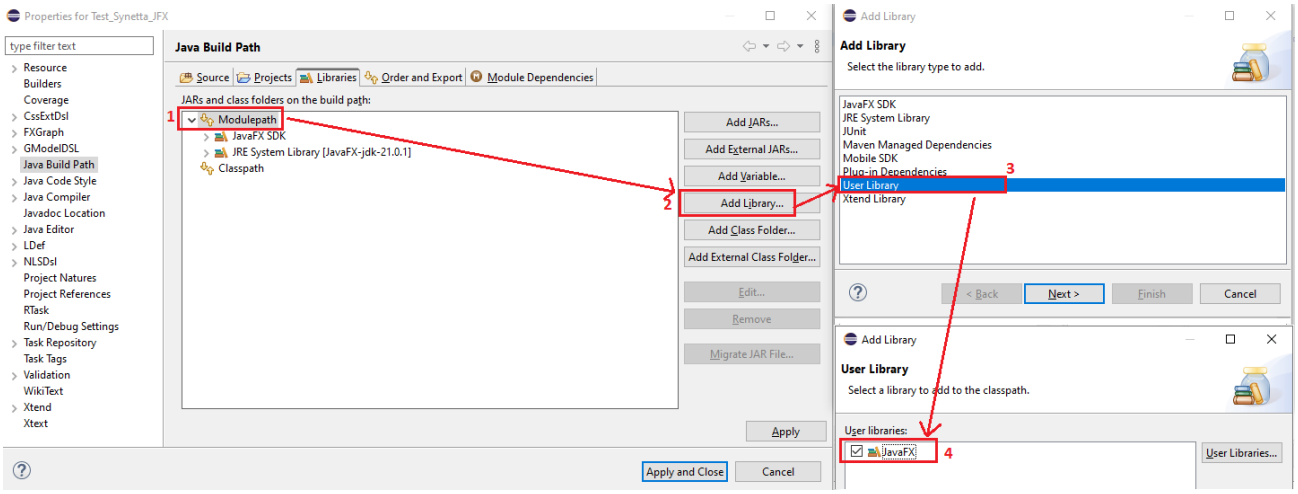
```
module votre_module {
    requires javafx.controls;

    opens votre_package to javafx.graphics, javafx.fxml;
}
```

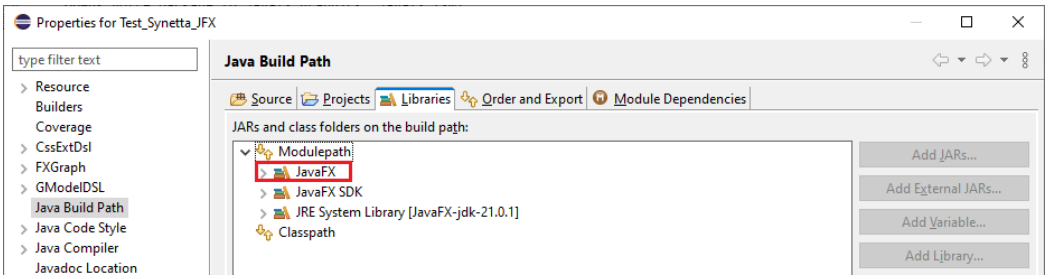
Implémenter une classe JavaFX.

Si les packages et/ou classes JavaFX ne sont pas reconnus dans votre projet, sélectionner votre projet puis ouvrir le menu Project > Properties, puis sélectionner à gauche, Java Build Path.

Au centre, sélectionner l'entrée Modulepath, puis cliquer sur le bouton "Add Library…", sélectionner le type "User Library", enfin cocher la librairie personnalisée nommée JavaFX que l'on a créé précédemment.




A présent, votre projet doit ressembler à ceci et doit fonctionner :




Si ça ne fonctionne pas, supprimer l'entrée JavaFX du Modulepath, et ajouter la librairie au Classpath.

# 6. Utiliser Maven

## Installer Apache Maven

Pour finir, télécharger l'outil Apache Maven (la dernière version en date étant la 3.9.5) :

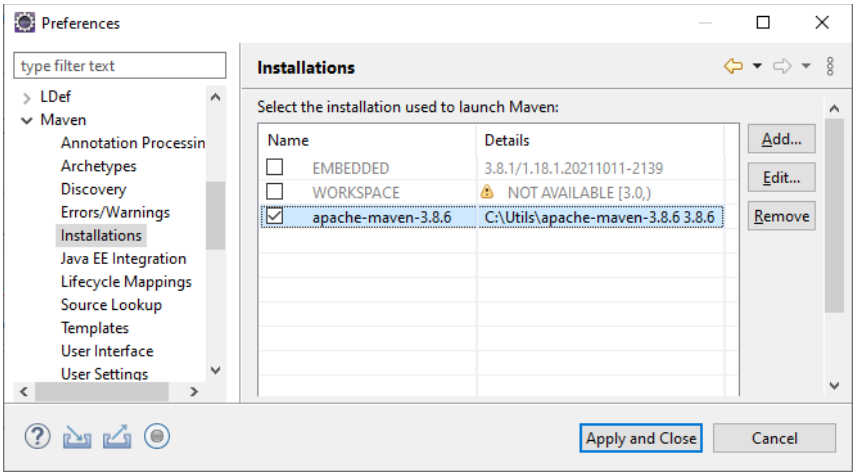
https://maven.apache.org/download.cgi

Récupérer le binaire au format zip, et décompresser-le. Prendre soin de bien retenir le chemin d'installation (on en a besoin par la suite, cf. paragraphe suivant).

## Lier Eclipse et Maven

Depuis Eclipse, ouvrir le menu Window> Preferences > Maven > Installations

Cliquer sur le bouton Add… et indiquer le chemin d'installation de Maven. Valider avec le bouton Apply and Close.

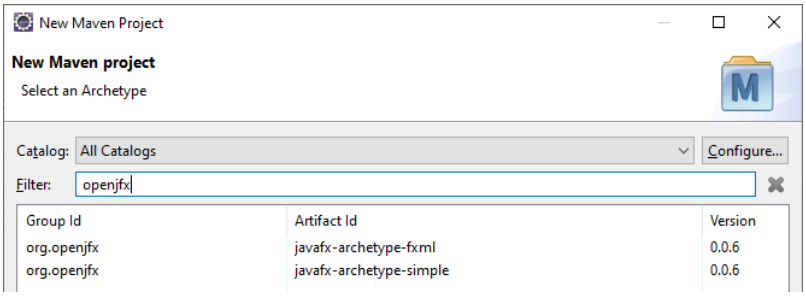



## Créer un projet JavaFX avec Maven

Conseil pour créer un projet JavaFX (modulaire ou pas) via Maven, utiliser le plugin OpenJFX :

https://openjfx.io/openjfx-docs/#maven

https://github.com/openjfx/javafx-maven-plugin




Déclarer votre dépendance à JavaFX :

```
<dependency>
    <groupId>org.openjfx</groupId>
    <artifactId>javafx-controls</artifactId>
    <version>21.0.1</version>
</dependency>
```

Déclarer le plugin maven :

```
<plugin>
    <groupId>org.openjfx</groupId>
    <artifactId>javafx-maven-plugin</artifactId>
    <version>0.0.8</version>
    <configuration>
        <mainClass>hellofx/org.openjfx.App</mainClass>
    </configuration>
</plugin>
```

Pour compiler et exécuter votre programme : ***mvn clean javafx:run***