



Labo 1 – First steps

101.1 Programmation impérative

Objectifs et donnée du laboratoire

Les objectifs de ce laboratoire sont :

1. se familiariser avec les outils utilisés dans le cours *101.1 Programmation impérative*;
2. faire vos premiers pas en *Scala* (si ce n'est pas déjà fait);

La durée de laboratoire est de **2 périodes**. Vous aurez l'occasion durant ce premier laboratoire de créer votre premier projet *Scala* et de faire connaissance avec l'environnement de développement *IntelliJ*.

Partie 1 – Les outils et la documentation

Plateforme ISC Learn

Tous les cours en ISC utilisent la même plateforme pour assurer l'échange d'informations entre étudiant-e-s et professeur-e-s. Celle-ci est disponible sur le lien <https://isc.hevs.ch/learn> et vous devriez avoir reçu vos informations de login par email normalement.

Sur le cours *101.1 Programmation impérative*, vous trouverez des informations sur le cours, des anciens examens, les dernières nouvelles... Cette plate-forme est accessible tant à l'intérieur qu'à l'extérieur de la HES-SO Valais mais avec votre login.

Tâche 1 : Installation des outils

Pour ce cours, nous allons utiliser le langage Scala avec comme éditeur de code *IntelliJ*. Ces deux outils sont disponibles gratuitement en téléchargement pour les principaux systèmes d'exploitation Windows, MacOS ou Linux.

Voici comment procéder à l'installation : 1. Téléchargez et installez IntelliJ Community Edition. Durant l'installation, vous pouvez choisir quels plugins installer. Choisissez le **plugin Scala**. 1. Si vous avez loupé l'installation du plugin ou vous avez déjà une version d'IntelliJ fonctionnelle, vous pouvez télécharger et installer le plugin Scala en suivant les instructions suivantes (cherchez "*Scala*" dans le menu des plugins)

Tâche 2 : Créer votre premier projet

1. Lancez IntelliJ et cliquez sur **File =>New =>Project**
2. Donnez lui un nom, comme par exemple **First project**
3. Choisissez où vous souhaitez mettre ce projet.
4. Dans le langage, choisissez **Scala** et comme **Build system** mettez **IntelliJ**.

Vous devez ensuite choisir le JDK, c'est-à-dire la machine virtuelle Java. Cliquez sur la flèche, puis **Download**. Nous allons toujours utiliser une version 17 du JDK et comme machine la version **GraalVM**. Pour le Scala SDK (c'est-à-dire la version de Scala) choisissez **impérativement** la dernière version 2.13 et surtout pas une version 3.0.

Vous devriez avoir la même chose que sur l'image suivante

Name the project HelloWorld Assuming this is your first time creating a Scala project with IntelliJ, you'll need to install a Scala SDK. To the right of the Scala SDK field, click the Create button. Select a 2.12 Scala SDK number (e.g. 2.12.XX) and click Download. This might take a few minutes but subsequent projects can use the same SDK. Please, do not choose a 2.13 version as it is not compatible with some examples shown in class. Once the SDK is created and you're back to the "New Project" window click Finish.

Je sais pas vraiment s'il faut installer les outils command line ou pas... J'hésite franchement.

1. GraalVM install depuis IntelliJ semble possible –1. Explication dans intellij

```
1 cs java-home --jvm graalvm-java17
2 cs java --jvm graalvm-java17 --setup
```

[https://scala-cli.virtuslab.org/scripting/ ???](https://scala-cli.virtuslab.org/scripting/???)

1. Allez sur la page (<https://docs.scala-lang.org/getting-started/index.html>)
2. Télécharger l'installateur [Scala](#) pour votre système d'exploitation. Cet installateur de commande s'appelle [Coursier](#) et c'est lui qui va gérer l'installation des outils pour pouvoir faire du [Scala](#).
3. Lancez une ligne de commande (TODO expliquer)
4. Dans la ligne de commande, tapez exactement la ligne suivante :

```
1 cs java --jvm 17
```

Cela va télécharger la machine virtuelle Java qui sert à exécuter les programmes écrits non seulement en Java (qui est un langage de programmation assez célèbre) mais également les programmes écrits en [Scala](#), [Clojure](#), [Kotlin](#), [Groovy](#), ...

1. Vous devriez maintenant pouvoir commencer à faire du Scala. Pour vérifier que tout fonctionne correctement, tapez `scala` dans la ligne de commande. Vous devriez voir la chose suivante :

```
1 Welcome to Scala 3.2.0 (11.0.15, Java OpenJDK 64-Bit Server VM).
2 Type in expressions for evaluation. Or try :help.
3
4 scala>
```

L'outil [Coursier](#) est un programme mais qui n'a pas d'interface utilisateur. Vous allez en rencontrer beaucoup dans vos études et dans votre vie de programmeur-euse ! En réalité, la majorité des programmes sont comme cela, sans information visible.

Tâche 1

Effectuez les tâches suivantes :

1. Essayer de retrouver sur le site le cours du jour ainsi que ce laboratoire.
2. Téléchargez les fichiers [RoomCalc](#), [scala](#) et [Input.java](#) et retrouvez ces fichiers sur le disque.

Partie 2. Environnement de développement

```
1 package labs.ideas
2
3 var length = 30
4 var symbol = "██"
5
6 def progress(value: Int) = {
7   require(value >= 1)
8
9   for (i <- 1 to value) {
10    print("[")
11    print(symbol * i)
12    print(" " * (length - i) + "]\n")
13    Thread.sleep(200)
14    print("\b" * (length + 2))
15  }
16 }
17
18 @main
```

```
19 def main() = {  
20   progress(10)  
21 }
```