

Getting Started Play Framework

Conseil

Pour utiliser Play, il faut lancer le script `activator.sh` sur Linux et `activator.bat` sur Windows. Ce script est en réalité un wrapper SBT, donc toutes les commandes [SBT](#) sont disponibles, mais il y a en quatre à retenir :

- `update` : Met à jour vos dépendences
- `compile` : Compile votre projet
- `run` : Lance votre projet en mode debug
- `start` : Déploie votre application en mode production

À noter que la commande `run` compile automatiquement vos sources à chaque modification du projet.

De façon général, il est conseillé d'utiliser un alias vers ce script pour deux raisons :

- La possibilité de créer un projet à n'importe quel moment
- Posséder plusieurs versions de Play que vous pourrez utiliser en fonction de vos besoins

Créer un projet selon ses besoins

Play vous génère un template lorsque vous lancez un projet. Par défaut, vous avez le choix entre trois templates déclinés sur une version JAVA et une version scala (minimal-akka, minimal et play) lorsque vous lancez la commande `new`. Cependant, pour des besoins plus spécifiques, vous pouvez télécharger d'autres templates depuis le site de [TypeSafe](#).

Par ailleurs, il y a le cas des plugins. Si vous souhaitez travailler sur les dernières versions de Play Framework, il est fortement déconseillé de les utiliser. Play évolue plus vite que ces derniers et ils deviennent alors incompatibles avec les dernières versions du framework. Nous vous conseillons plutôt d'utiliser des librairies déjà existantes qui ont fait leurs preuves (par exemple le plugin `mailer` servant à envoyer des mails est incompatible avec la version 2.3.5. À la place, l'ajout d'une dépendence vers [Apache Commons Email](#) a été réalisée dans le cadre du TP qui vous a été proposé).

Getting Started Play Framework

Se documenter est une [Play](#)

Play n'est vraiment pas technologie la mieux documentée qu'il existe. Pour éviter de longues recherches, nous vous proposons ci-dessous quelques conseils à adopter très rapidement. Pour rechercher une information, voici le protocole à suivre :

1. Vérifiez que la [documentation de Play](#) (JAVA/Scala) possède une section
2. Si cette section n'est pas suffisante, voir la même section sur l'autre langage (JAVA ► Scala ou Scala ► Java)
3. Consultez le *Migration Guide* pour voir les modifications réalisées sur la version sur laquelle vous travaillez par rapport au précédente
4. Consultez la documentation des versions précédentes
5. Vérifiez sur [Stackoverflow](#) si la réponse n'a pas déjà été postée
6. Vérifiez sur le [Google Group de Play Framework](#) si la question a déjà été posée
7. Allez sur le site de TypeSafe pour vérifier si un template utilise la partie du framework que vous souhaitez implémenter dans votre projet

Ressources annexes

Voir une liste non-exhaustive, d'articles et de vidéos que vous pouvez consulter :

[Une présentation d'introduction à Play en anglais très complète avec plein d'images de chats](#)

[Avis d'un développeur de Play à propos de la version 2](#)

[Un résumé rapide sur les nouveautés apportées par la seconde version de Play](#)

[Un tutoriel rapide sur Play 2](#)

[Une conférence à propos de Play 2 par le créateur](#)

[Une vidéo de présentation plus ancienne par le même créateur de la vidéo précédente](#)