

## BO - HUB

---

B-INN-000

# Workshop Mod Minecraft

---

Créer son premier mod Minecraft





# Workshop Mod Minecraf

language: java  
build tool: gradle



- The totality of your source files, except all useless files (binary, temp files, obj files,...), must be included in your delivery.

## WORKSHOP MOD MINECRAFT

### PRÉ-REQUIS

Si vous n'avez pas déjà installé les outils nécessaires à ce workshop suivez les instructions situées ici :

<https://github.com/alwyn974/workshop-mc-mod/blob/master/REQUIREMENTS.md>

### CONFIGURATION DU WORKSPACE

On va commencer par modifier le `mcmmod.info` pour y mettre les informations nécessaires.  
Il faut modifier le `modid` pour mettre le vôtre (uniquement des lettres minuscules, sans espace).  
Le reste à vous de modifier en fonction:

- `modid` : le modid de votre mod, remplacez donc `examplemod` par votre `modid`
- `name` : le nom du mod, à nouveau ce que vous avez mis dans la classe principale de votre mod
- `description` : une courte description du mod, vous pouvez ne rien mettre en laissant juste “ ”.
- `version` : la version du mod, ne changez rien ici `${version}` est automatiquement remplacé par la valeur se trouvant dans le `build.gradle`
- `mcversion` : la version de minecraft, comme pour la version, le remplacement est automatique
- `url` : le lien vers la présentation de votre mod
- `updateUrl` : l'url pour télécharger le mod
- `authorList` : la liste des auteurs. [“auteur1”, “auteur2”, “auteur3”] pour plusieurs auteurs, [“auteur”] pour un seul auteur
- `credits` : créditez ici les éventuels contributeurs.
- `logoFile` : le chemin à partir du début de l'archive vers le logo du mod. Si votre logo se trouve directement dans `src/main/resouces` et s'appelle `logo.png` il faudra mettre `logo.png`



- screenshots : une liste de screenshots, fonctionne comme pour le logo (chemin à partir du début de l'archive)
- dependencies : les dépendances, je conseille plutôt d'utiliser le String dépendance de l'annotation `@Mod`.

En suite, on va rename le package de base créer par le MDK, renommé donc `com.example.examplemod` par votre package.

Renommez aussi la classe `ExampleMod` par votre `ModidMod`



Pensez aussi à faire les modifications dans le fichier `ExampleMod.java`

## CRÉATION D'UN ITEM SIMPLE

---

On va devoir déclarer chaque item dans une classe spéciale, pour cela on va créer `ModidItems.java` qui contiendra:

- Une ArrayList en publique, statique et finale
-