



# Pourquoi utiliser Docker ?

---

## Sommaire

- [Problèmes classiques sans Docker](#)
- [Pourquoi Docker ?](#)
- ☒ [Avantages](#)

## Problèmes classiques sans Docker

- **Ça marche chez moi mais pas chez toi**
- Environnements différents entre développeurs :
  - Version de Python
  - Bibliothèques installées
  - Système d'exploitation (Linux, Windows, macOS)
- Difficulté à reproduire les bugs
- Déploiement non standardisé

## Pourquoi Docker ?

Docker permet de **containeriser** une application avec **tout son environnement** (code, bibliothèques, configuration système). Cela garantit qu'elle fonctionne de manière **identique partout** : local, cloud, CI/CD...

## Avantages

- **Portabilité** : le même container fonctionne sur n'importe quelle machine
- **Reproductibilité** : même version des dépendances pour toute l'équipe
- **Isolation** : chaque service (API, base de données, etc.) tourne dans son propre container
- **Déploiement simplifié** : en cloud, sur serveur distant, ou intégré à une chaîne CI/CD