

# Simplon Dev VR

## Exercice: Health Bar

### Partie 1:

1. Créer un nouveau projet
  - *Initialiser un repo Git*
2. Créer une classe **Player** dérivant de MonoBehaviour
  - *Contient une variable privée: **int initialHealth** modifiable dans l'inspecteur (100 par défaut)*
  - *Contient une variable privée **int health***
  - *Contient une propriété **int Health** permettant de récupérer la valeur de health, mais interdisant la modification de celle-ci*
  - *Dans le start initialiser health à la valeur de initialHealth*
  - **commit !**
3. Créer un gameobject avec le script **Player** avec un mesh de votre choix
  - *Exemple: cube, sphère, ou bien n'importe quel modèle 3d*
4. Ecrire un script **HealthDisplay** permettant d'afficher les points de vie (Health) de Player à l'écran
  - *il faut qu'on puisse voir Player HP = 100 dans le coin en haut à gauche de l'écran*
  - **commit !**
5. Créer une classe **Bomb**
  - Contient une valeur **int damage** modifiable dans l'inspecteur (10 par défaut)
  - **commit !**
6. Ajouter le script **Bomb** à un gameObject avec un mesh de votre choix
  - *Faites simplement en sorte qu'on puisse bien le distinguer de Player*
7. Faites en sorte qu'on puisse déplacer l'objet avec la souris en faisant un drag and drop
  - **commit !**
8. Coder le comportement suivant:
  - *si on drag and drop Bomb sur le Player, mettre à jour les points de vie du Player en soustrayant la valeur de damage de la bombe à la valeur health du Player*
  - *Après le drag and drop sur le Player, détruire l'objet Bomb*
  - *Attention: vérifier les points de vie affichés !*
  - **commit !**
9. **Optionnel:** afficher les points de vie du Player sous la forme d'une barre de vie
  - *Il faut quand même garder l'affichage du nombre de points de vie sous forme de texte à côté. Par exemple:*



Player HP = 90

- 
- **commit !**
- 10. Créer un nouvel objet et lui associer une nouvelle classe **Ally**, similaire au **Player**
  - *les scripts sont les mêmes seuls les noms changent*
  - *Ally doit aussi subir les dégâts de Bomb*
  - **commit !**
- 11. Modifier **Player** pour que celui-ci puisse être déplacé en faisant un drag and drop
  - **commit !**
- 12. Modifier **Ally** pour que celui-ci suive les déplacements de Player
  - Définir une distance de suivi (par défaut 2f)
  - Si Ally est trop loin ( > distance de suivi ), il se rapproche de la position de Player
  - Si Ally est trop près il s'arrête (<= distance de suivi)
  - **commit !**
- 13. Afficher les points de vie de **Ally** dans le coin en haut à droite de l'écran
  - *Le script écrit au point 4 est-il réutilisable sans modification ?*
  - *Ecrire un script **HealthBarAlly** pour afficher les points de vie d'Ally*
  - **commit !**

Dans la suite nous allons voir une méthode possible pour afficher les points de vie de Player et Ally en utilisant un seul script à l'aide d'un ScriptableObject

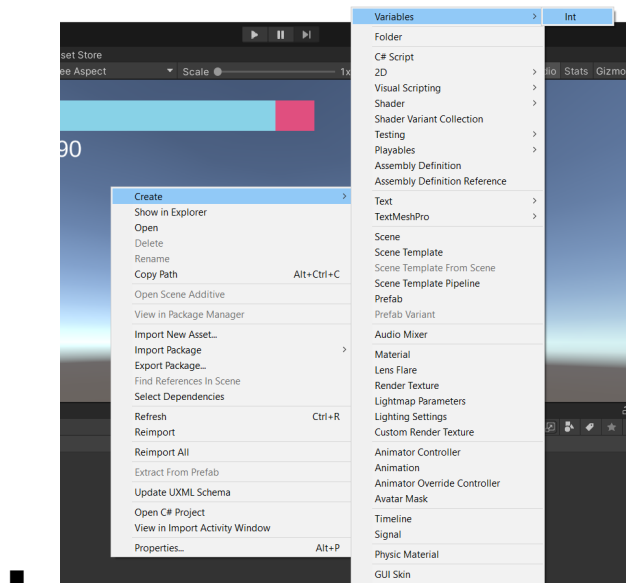
## Partie 2: ScriptableObject

14. Créer un script **IntVariable** avec le code suivant

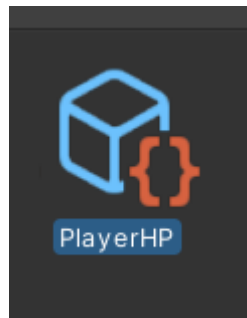
```
using UnityEngine;

[CreateAssetMenu(menuName = "Variables/Int")]
0 references
public class IntVariable : ScriptableObject
{
    0 references
    public int Value;
}
```

- 
- Vous pourrez ensuite créer des instances de ce ScriptableObject comme ceci:
  - Clic droit dans la fenêtre Project > Create > Variables > Int



**15. Créer un `IntVariable` que vous nommerez **PlayerHP****



- 
- Fixer la valeur de **PlayerHP** à 75

**16. Modifier le script **HealthDisplay** pour afficher la valeur de **PlayerHP****

- **commit !**

**17. Modifier Player pour faire en sorte que `private int health` deviennent un `[SerializeField] IntVariable health`**

- **commit !**

**18. Dans l'inspecteur de Player, utiliser **PlayerHP** pour le champ `IntVariable` Health**

- **commit !**

**19. Les points de vie affichés sont-ils corrects lorsque le Player se prend de dégâts ?**

- **commit !**

**20. Modifier Ally pour que celui-ci utilise aussi un `IntVariable`**

- **commit !**

**21. Afficher les points de vie d'Ally en utilisant le script **HealthDisplay** et non **HealthDisplayAlly****

- **commit !**