



## DÉFI RAPIDE !

**Objectif :** Tu as déjà vu comment créer des classes **Guerrier** et **Magicien** qui héritent de la classe **Personnage**. Maintenant, imagine que tu veux ajouter un nouveau type de personnage à ton jeu : le **Rôdeur**. Les rôdeurs sont connus pour leur agilité et leur capacité à tirer à l'arc. Ils ont donc, en plus des attributs de **Personnage**, un attribut **agilité**.

**Ta Mission :** Crée une classe **Rôdeur** (**Prowler**) qui hérite de **Personnage** (**Character**) et ajoute-lui un attribut **agilité** (**agility**). Ensuite, crée une méthode **tirerArc()** (**archery**) qui affiche un message spécifique au rôdeur. N'oublie pas d'appeler le constructeur de la classe parente pour initialiser **pointsDeVie** (**health**) et **force** (**power**) !

Ce défi te permet de pratiquer l'héritage en POO et de voir comment il peut être utilisé pour ajouter de la diversité et de la richesse à la conception de tes personnages de jeu.

Pour ajouter une nouvelle classe **Rôdeur** (**Prowler**) avec l'attribut **agilité** (**agility**) et une méthode **tirerArc** (**archery**), suis ces étapes :

### EXEMPLE DE CODE VALIDE POUR LE DÉFI

```
class Prowler extends Character {
  constructor(health, power, agility) {
    super(health, power); // Appel du constructeur de la classe parente
    this.agility = agility;
  }
  archery() {
    console.log("Tire une flèche avec agilité !");
  }
}

// Création d'une instance de Prowler pour tester
let prowler = new Prowler(100, 50, 75);
prowler.attack(); // Appel de la méthode héritée de Character
prowler.archery(); // Appel de la méthode spécifique à Prowler
```

## EXPLICATION DU CODE

**Constructeur :** Utilise **super(health, power)** pour appeler le constructeur de la classe parente (**Character**) et initialiser les attributs **health** et **power**.

**Attribut **agility** :** Ajoute un nouvel attribut **agility**.

**Méthode **archery** :** Ajoute une nouvelle méthode **archery** qui affiche un message spécifique au **Rôdeur**.

## SCÉNARIOS DE TEST

**Initialisation :** Le **Rôdeur** est correctement initialisé avec des valeurs pour **health**, **power** et **agility**.

**Héritage :** Le **Rôdeur** peut utiliser la méthode **attack** héritée de la classe **Character**.

**Méthode spécifique :** La méthode **archery** affiche le message attendu.

Ce défi te permet de pratiquer l'héritage en programmation orientée objet et de voir comment il peut être utilisé pour ajouter de la diversité et de la richesse à la conception de tes personnages de jeu.