Classe Player

**Attributs :**

Health : int

* 1. Représente la vie du joueur.
  2. Varie entre 0 et 100.

Velocity : Vector2f

1. Représente la vitesse et la direction du joueur.

Acceleration : Vector2f

1. Représente l’accélération du joueur, le changement de vélocité.

Nom : string

Représente le nom du joueur.

**Constructeurs/Destructeurs :**

Player() : Player

1. Met l’attribut health à 100.
2. Met l’attribut velocity à 1.
3. Met l’attribut acceleration à 0.

Player(string nom) : Player

1. Met l’attribut nom égal à la variable nom.
2. Appelle Player().

~Player()

1. Vide pour l’instant.

**Accesseurs :**

getHealth() : int

1. Retourne la valeur de l’attribut health.

getVelocity() : Vector2f

1. Retourne la valeur de l’attribut velocity.

getAcceleration() : Vector2f

1. Retourne la valeur de l’attribut acceleration.

**Mutateurs :**

setHealth(int addition) : void

1. Ajoute addition à l’attribut health.
2. Si health devient plus bas que 0 il devient 0.
3. Si health devient plus haut que 100 il devient 100.

setVelocity(Vector2f addition) : void

1. Ajoute addition à l’attribut velocity.

setAcceleration(float addition) : void

1. Ajoute addition à l’attribut acceleration.

**Méthodes :**

move(Vector2f movement) : void

1. Ajoute le vecteur movement au vecteur acceleration.
2. Ajoute le vecteur velocity au résultat précédent.
3. Ajoute le tout à la position du joueur.

substractHealth(int substract) : void

1. Enleve substract de l’attribut health.
2. S’assure que health ne devient pas plus petit que 0.

addHealth(int addition) : void

1. Ajoute addition de l’attribut health.
2. S’assure que health ne devient pas plus grand que 100.