Gear

+ combine(Gear)

+ getFullName(): String

+ getDefense(): int
+ getAttack(): int

Character

- initialAttack: final int
- initialDefense: final int
- attack: int
- defense: int
- headGear: HeadGear
- leftHandGear: HandGear
- rightHandGear: HandGear
- leftFootwear: Footwear
- rightFootwear: Footwear

+ constructor(initialAttack, initialDefense)
+ getAttack(): int
+ getDefense(): int
- computeAttack(): int
- computeDefense(): int
+ pickUp(Gear): void
+ getHeadGear() : HeadGear
+ getLeftHandGear() : HandGear
+ getRightHandGear() : HandGear
+ getLeftFootGear() : FootGear
+ getRightFootGear() : FootGear

Implement

AbstractGear

- adj: List<String>
- noun: String
- attack:int
- defense: int

+ constructor(String, String)
+ getFullName(): String
+ getAttack(): int
+ getDefense(): int

parent   parent

parent

child   child   child

HeadGear

+ constructor(String, String, int)
+ combine(Gear)

HandGear

+ constructor(String, String, int)
+ combine(Gear)

Footwear

+ constructor(String, String, int, int)
+ combine(Gear)