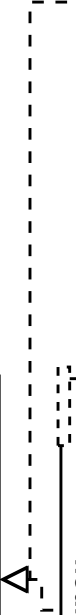**Case**

- #reserves: LinkedList<Nourriture>

- +Case()
- +Case(reserves:LinkedList<Nourriture>)
- +ajouterNourriture(nourritureAjoutee:Nourriture): void
- +supprimerNourriture(nourritureConsommee:Nourriture): void

Nourriture

- #nom: String
- #energie: int

- +donneEnergie(lofteur:Neuneu): void
- +Nourriture(nom:String,energie:int)

Loft

- #terrain: Case[][]
- #hauteur: int
- #largeur: int
- #population: ArrayList<Neuneu>

- +Loft()
- +ajouterNeuneu(arrivant:Neuneu): void
- +supprimerNeuneu(partant:Neuneu): void

Commestible

- +donneEnergie(lofteur:Neuneu): void

Neuneu

- #nom: String
- #position: Case
- #energie: int
- #environnement: Loft

- *+seDeplacer(): void*
- *+manger(nourriture:Commestible): void*
- +seReproduire(partenaire:Neuneu): Neuneu
- +donneEnergie(lofteur:Neuneu): void
- +cycleDeVie(): void

Erratique

- +Erratique(nom:String)
- +seDeplacer(): void
- +manger(nourriture:Commestible): void

Lapin

- +seDeplacer(): void
- +manger(nourriture:Commestible): void
- +Lapin(nom:String)

Vorace

- +Vorace(nom:String)
- +seDeplacer(): void

Cannibale

- +Cannibale(nom:String)
- +seDeplacer(): void