

TD3 : POO Class Abstraite et Interface

Exercice 1 : Animaux - nageurs et volants

Vous allez modéliser un système d'animaux en utilisant une classe abstraite et des interfaces.

Consignes :

1. Créez une classe abstraite 'Animal' contenant :

- un attribut 'nom' (String)
- un constructeur
- une méthode concrète 'manger()' qui affiche un message
- une méthode abstraite 'seDeplacer()'

2. Créez deux interfaces :

- 'Nageur' avec la méthode 'nager()'
- 'Volant' avec la méthode 'voler()'

3. Implémentez trois classes :

- 'Poisson' : hérite de 'Animal' et implémente 'Nageur'
- 'Oiseau' : hérite de 'Animal' et implémente 'Volant'
- 'Canard' : hérite de 'Animal' et implémente 'Nageur' et 'Volant'

4. Dans une méthode 'main', créez plusieurs animaux et testez leurs méthodes.

Exercice 2 : Gestion d'appareils intelligents

Vous allez modéliser une maison intelligente.

Consignes :

1. Créez une classe abstraite 'Appareil' avec :

- un attribut 'nom'
- une méthode 'allumer()'
- une méthode 'eteindre()'
- une méthode abstraite 'fonctionSpeciale()'

2. Créez une interface 'Connectable' avec :

- 'connecterWifi()'
- 'deconnecterWifi()'

3. Implémentez deux classes :

- ‘LampeIntelligente’ : change de couleur
- ‘AspirateurRobot’ : commence le nettoyage

Ces deux classes héritent d’‘Appareil’ et implémentent ‘Connectable’.

4. Dans ‘main’, créez plusieurs d’appareils et testez leurs méthodes.

Exercice 3 : Système de paiement multimodal

Vous allez modéliser plusieurs moyens de paiement.

Consignes :

1. Créez une classe abstraite ‘Paiement’ avec :

- un attribut ‘montant’
- un attribut ‘statut’ (succès ou échec)
- une méthode abstraite ‘effectuerPaiement()’

2. Créez une interface ‘Verifiable’ avec la méthode :

- ‘verifierIdentite()’

3. Implémentez deux classes :

- ‘CarteBancaire’ (vérifie numéro de carte)
- ‘PortefeuilleElectronique’ (vérifie email et solde)

Ces classes héritent de ‘Paiement’ et implémentent ‘Verifiable’.

4. Dans ‘main’, créez plusieurs paiements et testez-les.