1. Classe abstraite : Composant

La classe Composant est la base abstraite pour tous les composants.

Attributs :

- **id (int)** : identifiant unique (doit être supérieur à 0, sinon une exception ValueError est levée).
- **nom (str)** : nom du composant (doit être une chaîne non vide, sinon ValueError).
- **marque (str)** : marque du composant.
- **prix (float)** : prix du composant (doit être un nombre positif, sinon ValueError).

Méthodes :

- **__init__(self, id, nom, marque, prix)** : initialise les attributs avec vérification.
- **afficher_details(self)** : méthode abstraite à implémenter dans les classes filles.
- **modifier_details(self, nom=None, marque=None, prix=None)** : modifie les détails du composant. Les arguments laissés à None ne sont pas modifiés.
- **__str__(self)** : méthode de représentation pour afficher les informations générales.

2. Classe : Processeur hérite de la classe composant

Attributs spécifiques :

- **frequence (float)** : fréquence du processeur en GHz (doit être > 0, sinon ValueError).
- **nombre_coeurs (int)** : nombre de cœurs (doit être >= 1, sinon ValueError).

Méthodes spécifiques :

- **__init__(self, id, nom, marque, prix, frequence, nombre_coeurs)** : initialise les attributs avec vérification.
- **afficher_details(self)** : affiche les détails spécifiques du processeur.
- **__str__(self)** : représentation spécifique.

3. Classe : Mémoire hérite de la classe composant

Attributs spécifiques :

- **capacite (int) : capacité de la mémoire en Go (doit être > 0, sinon ValueError).**
- **type_memoire (str) : type de mémoire (DDR3, DDR4, etc.).**

Méthodes spécifiques :

- **__init__(self, id, nom, marque, prix, capacite, type_memoire) : initialise les attributs avec vérification.**
- **afficher_details(self) : affiche les détails spécifiques de la mémoire.**
- **__str__(self) : représentation spécifique.**

4. Classe : CarteMere

Attributs :

- **composants (list) : liste des composants installés.**

Méthodes :

- **__init__(self) : initialise la liste des composants.**
- **ajouter_composant(self, composant) : ajoute un composant. Une exception est levée si un composant avec le même id existe déjà.**
- **supprimer_composant(self, id) : supprime un composant par son identifiant.**
- **mettre_a_jour_composant(self, id, **kwargs) : met à jour un composant spécifique.**
- **afficher_composants(self) : affiche la liste des composants.**
- **filtrer_composants(self, critere, valeur) : filtre les composants par un critère.**
- **calculer_valeur_totale(self) : calcule et retourne la valeur totale de tous les composants.**
- **rechercher_composant_par_nom(self, nom) : retourne les composants correspondant à un nom donné.**
- **Sérialisation :**
    - **sauvegarder_composants_json(self, chemin_fichier)**
    - **charger_composants_json(self, chemin_fichier)**
    - **sauvegarder_composants_csv(self, chemin_fichier)**