UNIVERSITÉ BOURGOGNE EUROPE

L1 STS — année 2025
Info 2a — programmation objet
Examen d'appel du vendredi 20 juin 2025

Modalités

- ✓ Durée : 2h.
- ✓ Vous pouvez utiliser une feuille de notes manuscrites (écrites à la main) non photocopiée recto-verso de taille quelconque — si vous pouvez l'utiliser sans déborder de l'espace qui vous est réservé.
- ✓ Rendez l'annexe jointe si vous l'utilisez. Elle compte pour un intercalaire.
- ✓ Dans l'ensemble du devoir, les collections (listes, ensembles, suites…) peuvent être codées au choix par des tableaux ou des `ArrayList`.

I. Situation préliminaire

Comme pour la première session, il s'agit de gérer les repas des clients pour des restaurants.

La carte du restaurant répertorie la liste de tous les plats qu'un client peut commander (Le client ne sera pas représenté). Elle a aussi un nom (Par exemple, Carte de saison).
Plus précisément, un plat comprend un nom (exemple : Jambon à la broche, Homard thermidor, Friture de joels, Crème brûlée à la myrte…), sa catégorie (Entrée, Plat principal, Dessert, Salade…), la carte à laquelle il appartient, et sa disponibilité — qui exprime qu'il n'y a pas de rupture d'approvisionnement pour le plat ou l'un de ses constituants. Chaque plat a aussi un prix de vente. Le nom et la catégorie d'un plat sont des propriétés dites intrinsèques, qui sont spécifiées lors de la création d'un nouveau plat et ne doivent plus pouvoir être modifiées ultérieurement. La carte associée au plat ainsi que son prix sont spécifiés à la création et peuvent évoluer. La disponibilité d'un plat peut aussi évoluer mais un nouveau plat est systématiquement considéré comme disponible.

Une table réunit un ou plusieurs clients : elle a un numéro (ex. : table 3). À chaque table, est associée la liste des commandes des clients de la table. Une commande correspond à un plat dans une quantité donnée (ex : 2 jambons à la broche ou 1 friture de joels…). Le prix associé à une commande est le prix du plat fois la quantité commandée, le prix associé à une table est la somme des prix des commandes des clients de la table : il constitue la note qui devra être payée par cette table.

Une représentation statique possible de cette situation est celle de l'annexe (en haut). Dans le diagramme, les collections sont représentées comme le sont les tableaux.

Questions

Répondez aux questions suivantes en conformité avec les spécifications de l'énoncé en considérant que les accesseurs et constructeurs sont déjà écrits — sauf lorsqu'il s'agit justement d'écrire l'une de ces méthodes.

1. Indiquez les modificateurs d'accès (`private` ou `public`) associés aux accesseurs : `getXXX()` et `setXXX()` des attributs de la classe `Plat`.
2. Écrivez le constructeur de la classe `Plat`.
3. Une commande est dite valide si et seulement si le plat qui lui est associé est disponible. Écrivez la méthode d'instance `isValide()` de la classe commande qui traduit cette propriété.
4. Selon le même principe, écrivez aussi la méthode d'instance `isValide()` de la classe `Table` qui exprime le fait que toutes les commandes de la table sont valides.
5. Écrivez la méthode `getListePlats(String categorie)` de la classe `Table` qui, pour une catégorie donnée, restitue une chaîne de caractères représentant la liste des noms des plats des commandes de la table. Si catégorie est la chaîne `"Entrée"`, cela peut donner par exemple :

   Catégorie entrée :

   Salade de poivrons façon méchouia
   Brick à l'œuf sauce meurette
   Fleur de sureau à la croque au sel et pain d'épice

II. Situation élaborée

Dans la situation précédente, les clients ne pouvaient commander qu'à la carte. On souhaite désormais qu'ils puissent aussi choisir dans un menu pour un prix forfaitaire.

Contrairement au choix qui a été fait en première session, un menu est ici vu comme un plat particulier composé lui-même d'autres plats, dits plats simples (Ceux qui ne sont pas des menus). Les plats simples ont les mêmes propriétés que les plats de la situation préliminaire, avec toujours une disponibilité qui peut évoluer. Comme tout plat, un menu a un nom : Menu du jour par exemple, un prix : 21€ et une catégorie qui a la même valeur que le nom (Ici, Menu du jour). La disponibilité d'un menu dépend de celle des plats : un menu est disponible si tous les plats qui le composent le sont aussi.

Questions

1. Faites figurer dans le deuxième diagramme de l'annexe les relations entre classes qui rendent compte de ces nouvelles modalités.
2. Ajoutez à ce diagramme les déclarations des attributs nécessaires pour représenter toutes les informations de l'énoncé dans cette nouvelle forme.
3. Écrivez les méthodes `toString` de `Plat`, `PlatSimple` et `Menu`. Dans le cas d'un plat simple, cette méthode renvoie le nom du plat suivi de son prix ou de la mention indisponible si le plat est en rupture. Par exemple, Jambon à la broche : 18.0€, ou Homard Thermidor : indisponible. Pour ce qui concerne les menus, la méthode restitue une chaîne de caractères qui, comme pour le plat, indique son nom suivi, suivant les cas, de son prix ou de la mention indisponible, puis, si le menu est disponible, de la liste des noms des plats qui le constituent. Voir les deux exemples suivants :

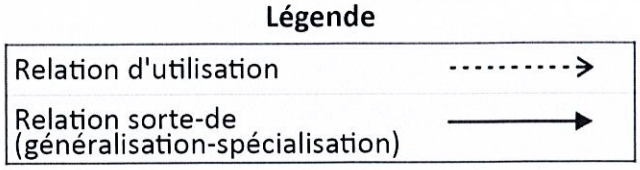
   Menu du jour : 21€

   Salade de poivrons façon méchouia
   Jambon à la broche
   Crème brûlée à la myrte

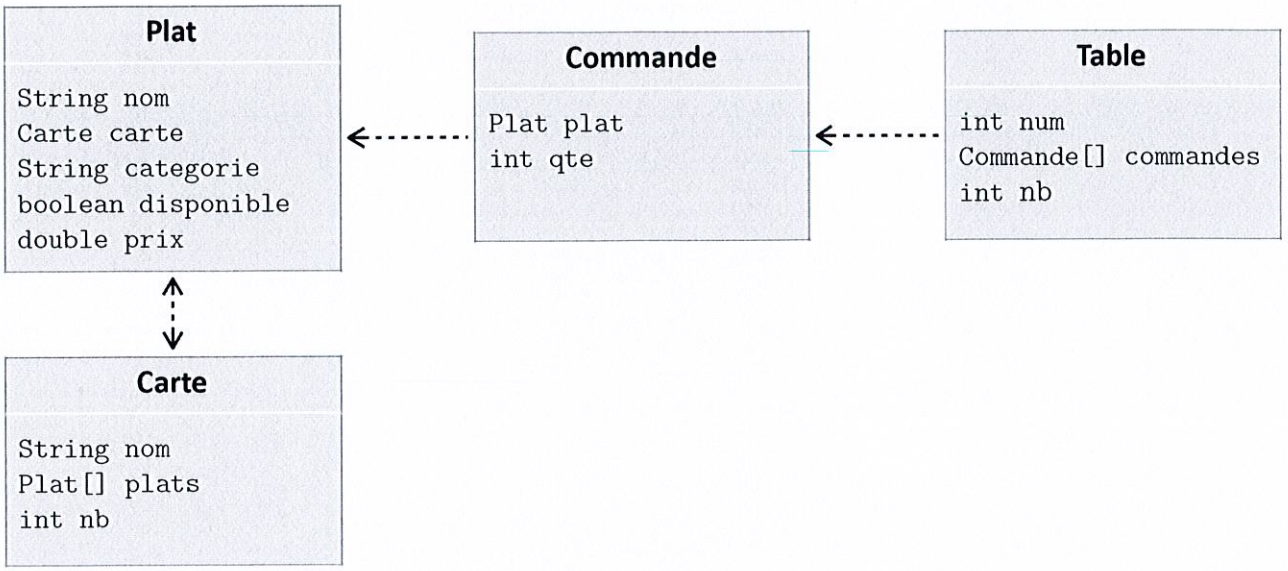
   Menu de la mer : indisponible.

4. Écrivez les méthodes `isDisponible(…)` des classes `PlatSimple`, `Plat` et `Menu`. Pour ce faire, utilisez le polymorphisme hiérarchique en expliquant comment vous employez cette notion.

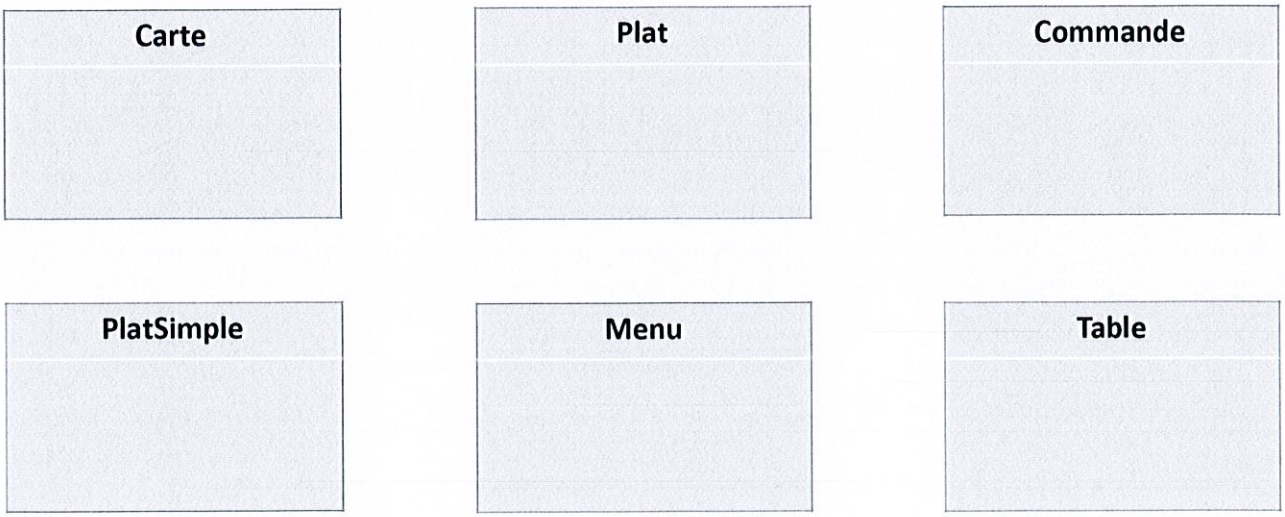
Annexe



I. Situation préliminaire



II. Situation élaborée



N° d'anonymat :