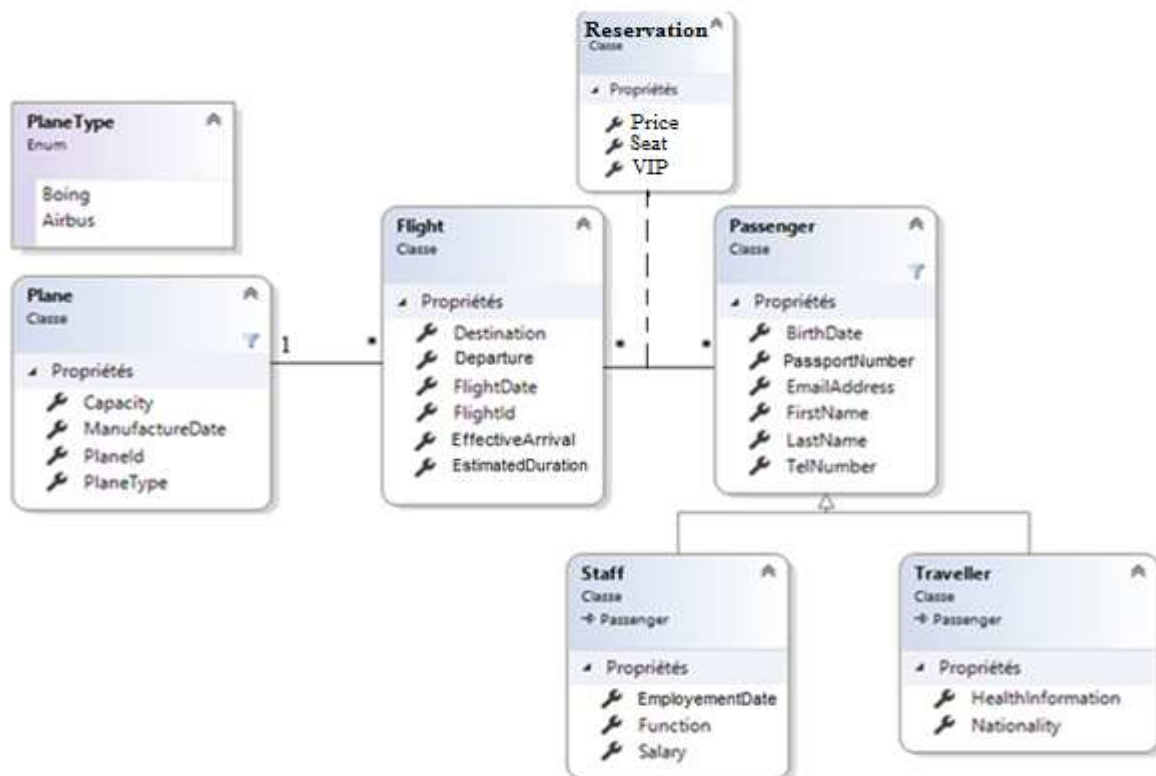


IV Stratégies d'héritage

1. Configurer l'héritage schématisé dans le diagramme de classe de façon à ce que les entités seront mappées à une seule table **Passengers** avec la colonne discriminatrice **IsTraveller** :
 - a) prend la valeur 1 si le type du passager est **Traveller**,
 - b) prend la valeur 2 si le type du passager est **Staff** et,
 - c) prend la valeur 0 sinon.
2. Mettre à jour la BDD et vérifier la nouvelle configuration.
3. Configurer l'héritage schématisé dans le diagramme de classe de façon à ce que les entités seront mappées sur 3 tables.
4. Mettre à jour la BDD et vérifier la nouvelle configuration.
5. Restaurer la BDD avec la stratégie d'héritage par défaut (**TPH** : Table Par Hiérarchie).

V Table porteuse de données

6. Dans le projet **Domain** ajouter la classe **Reservation** schématisée dans le diagramme suivant :



7. Configurer la classe **Reservation**.
8. Mettre à jour la BDD.
9. Tester en ajouter une réservation.

VI Chargement des données

10. Faites une opération d'ajout d'un vol en lui attribuant un avion. Puis afficher le vol ; et afficher l'avion en utilisant la propriété de navigation **MyPlane**.
11. Récupérer le vol déjà inséré à partir de la BDD, l'afficher ainsi que son avion en utilisant **MyPlane**. Que constatez-vous ?
12. Dans le projet **Data**, installer **EntityFrameworkCore.Proxies**.
13. Activer le **LazyLoading** dans la méthode **OnConfiguring** de la classe context.
14. Décorer toutes les propriétés de navigation avec le mot clé **virtual**.
15. Re-tester Q11.