

Vous êtes chargé de développer une petite API pour la gestion de réservations de vols. Cette API permettra de gérer les avions, les vols, les passagers et les billets. L'objectif est de manipuler une base MongoDB à l'aide de Mongoose et de proposer un CRUD sur chaque ressource.

1. Collection avions

```
{
  modele: String,      // ex: "Airbus A320"
  compagnie: String,   // ex: "Air France"
  capacite: Number, max 10 // nombre total de places
  placesRestantes: Number, // nombre de places encore disponibles
  enService: Boolean,   // avion actif ou non
}
```

Contraintes :

- `modele`, `compagnie` et `capacite` sont obligatoires.
- `placesRestantes` \leq `capacite`
- `enService` par défaut = `true`

2. Collection vols

```
{
  numeroVol: String,      // ex: "AF742"
  origine: String,        // ex: "Paris"
  destination: String,    // ex: "Dakar"
  dateDepart: Date,
  dateArrivee: Date,
  avion: ObjectId,        // référence vers un avion
  statut: String,         // "prévu", "retardé", "terminé", "annulé"
}
```

Contraintes :

- Le champ `avion` doit être un `_id` valide existant dans la collection **avions**.
- Les `dateDepart` et `dateArrivee` doivent être cohérentes (`dateArrivee` $>$ `dateDepart`).
- Par défaut, `statut` = `"prévu"`.

3. Collection passagers

```
{
  nom: String,
  prenom: String,
  email: String,
  pays : String,
  dateInscription: Date
}
```

Contraintes :

- nom, prenom, email, pays obligatoires. email unique.
- dateInscription = date actuelle par défaut.

4. Collection billets

```
{
  vol: ObjectId,      // référence au vol
  passager: ObjectId, // référence au passager
  numeroSiege: String, // ex: "14A"
  classe: String,      // "Economie", "Affaires", "Première"
  prix: Number,
  dateReservation: Date,
  modePaiement: String, // "CB", "PayPal", "Espèces"
  statut: String        // "confirmé", "annulé"
}
```

Contraintes :

- vol et passager doivent exister.
- prix > 0
- dateReservation par défaut = date actuelle.
- statut par défaut = "confirmé"
- classe par défaut = "Economie"

Routes à implémenter

Ressource	Méthode	Chemin	Description
Avions	GET	/api/avions	Lister tous les avions
Avions	POST	/api/avions	Créer un avion
Vols	GET	/api/vols	Lister tous les vols
Vols	POST	/api/vols	Créer un vol
Passagers	POST	/api/passagers	Créer un passager
Billets	POST	/api/billets	Créer un billet

Règles métiers

1. Lors de la création d'un billet : vérifier le vol, la disponibilité et décrémenter placesRestantes.
2. Lors de la suppression d'un billet : ré-incrémenter les placesRestantes.
3. Interdire la suppression d'un vol s'il a des billets.

1. Gestion des vols

- Combien de vols sont **prévu, retardé, terminé** ou **annulé** ?
- Quelle est la **répartition des vols par compagnie aérienne** ?
- Combien de **vols ont lieu ce mois** ?

2. Gestion des avions

- Combien d'avions sont actuellement **en service / hors service** ?
- Quelle est la **capacité totale disponible** dans la flotte ?
- Quelle compagnie possède le **plus grand nombre d'avions** ?

3. Gestion des passagers

- Combien de **passagers enregistrés** au total ?
- Combien de **nouveaux passagers** ce mois-ci ?

4. Gestion des billets

- Combien de **billets vendus** ce mois ?
- Quel est le **revenu total généré** par les ventes ?
- Quelle est la **classe de vol la plus vendue** (Économie, Affaires, Première) ?
- Combien de billets ont été **annulés** ?

5. Page principale

- Nombre total de vols actifs
- Nombre d'avions en service
- Nombre total de passagers
- Nombre total de billets vendus
- Revenu total généré
- Taux d'annulation global