

Vols(id int, #airport_depart_id int, #airport_arrivee_id int, #compagnie_id int, #avion_id int, date_depart date, date_arrive date, created_at timestamp, updated_at timestamp).
avions(id int, # compagnie_id int, libelle varchar(50), capacite int, created_at timestamp, updated_at timestamp)
compagnies(id int, #airport_base_id int, iata char(2), nom_compagnie varchar(30), created_at timestamp, updated_at timestamp)
airports(id int, iata char(2), icao char(4), libelle varchar(50), created_at timestamp, updated_at timestamp)

Partie laravel 22pts

- 1) Ecrire la commande artisan pour créer le modèles **Vol** avec son controlleur, ressource, factory, seeder, migration, Request
- 2) Donner les lignes de codes à ajouter au modèle **Vol** afin que tous les attributs puissent être affectés lors de la création ou de la mise à jour (indication : utiliser la propriété **fillable**)
- 3) Ecrire les méthodes suivantes
 - a. **airportDepart** dans le model **Vol** qui permettra d'affecter un vol à un aéroport de départ (belongsto)
 - b. **arrivalVols** dans le model **airport** qui permettra d'associer les arrivées des vols à un airport (hasmany)
- 4) Donner un ordre correcte à suivre pour créer les migrations de toutes les tables, afin que la commande **php artisan:migrate** passe sans erreurs par la suite (exemple de réponse attendue: 1-TABLENAME1 2-TABLENAME2)
- 5) Donner la ligne de code dans la méthode up de migration qui décrira les colonnes suivantes
 - a. airport_base_id de la table avions
 - b. icao de la table airports
- 6) La création d'un controlleur de type resource permet de créer les méthodes de crud, écrire le code à mettre dans la page web.php afin de générer les différentes routes du controlleur **AvionController**

2 sur 6

Sachant que la commande **php artisan:route list** affiche ceci :

```
POST      avions ..... avions.store › AvionController@store
GET/HEAD  avions/create ..... avions.create › AvionController@create
GET/HEAD  avions/{avion} ..... avions.show › AvionController@show
PUT/PATCH avions/{avion} ..... avions.update › AvionController@update
DELETE    avions/{avion} ..... avions.destroy › AvionController@destroy
GET/HEAD  avions/{avions }/edit ..... avions.edit › AvionController@edit
```

Ecrire le code des méthodes du controlleur **AvionController** suivantes :

- 7) **"index"** qui renvoie la vue **"index"** avec la liste des Avions
- 8) Ecrire le code de la vue **"index"** qui permet :
 - a. D'étendre le layout présent sous le chemin **ressources/views/avions/layout.blade.php**
 - b. D'afficher tous les avions dans un tableau dans la section : **"content"**
 - c. De renvoyer à travers un bouton ajouter vers le formulaire d'ajout (create)
- 9) Ecrire la ligne de code qui permettre de valider que le champs aéroport de départ est requis et existe bel et bien dans la liste des aéroports
- 10) Sachant qu'on a un middleware d'authentification qui s'appelle **AuthMiddleware**, appliquer ce middleware sur la route créée dans la question 6
- 9) Ecrire la ligne de code qui permettre de valider que le champs aéroport de départ est requis et existe bel et bien dans la liste des aéroports
- 10) Sachant qu'on a un middleware d'authentification qui s'appelle **AuthMiddleware**, appliquer ce middleware sur la route créée dans la question 6
- 11) Ecrire une méthode **getNumberPartnerVols (\$nom)** qui aura le même comportement de la question 3)b. avec Eloquent, DBquery et/ou DBraw

3 sur 6