

TP01 – MÉTÉOMANIA – 17,5%

MISE EN SITUATION

Fort de votre expérience de quelques semaines dans le développement de l'API *Space-explorer*, votre superviseur vous demande de débiter le développement des services Web pour l'acquisition des données météo des différentes stations à travers la galaxie. Le nom de code du projet est **Météomania**.

MANDAT ET REMISE

Vous devez développer les routes présentes dans le document, ces routes permettront de sauvegarder et de retrouver les données des observations des stations météo.

Vous devez remettre un projet node.js complet qui inclut seulement les routes demandées dans ce travail.

Votre travail **devra être remis avant 23 h 59, le 16 octobre 2020** sur Léa dans une archive zip. Votre archive devra contenir :

- Un fichier **README.md** avec votre prénom, nom et matricule étudiant
- Un fichier **package.json** avec les informations à propos de votre projet
- Tous les fichiers de code source, nécessaire à l'exécution de votre serveur
- Un fichier **tests.json**, exporté de Postman, permettant de tester vos différentes routes.
- Les **node_modules** ne doivent pas être remis.

RESSOURCES ET REPRÉSENTATION

OBSERVATION MÉTÉO

REPRÉSENTATION JSON DE REQUÊTE

Les stations météo positionnées un peu partout dans la galaxie communiquent de façon régulière avec le serveur les informations présentes dans la représentation JSON suivante:

```
{
  "location": {
    "station": "XL908",
    "coord": {
      "lon": -122.08,
      "lat": 37.39
    }
  },
  "temperature": 15,
  "pressure": 1011,
  "humidity": 0.55,
  "feelslike": 15,
  "uvIndex": 5,
  "wind": {
    "speed": 12,
    "degree": 165
  },
  "clouds": {
    "cloudcover": 0.72
  },
  "observationDate": "2020-06-28T11:39:13.934000Z",
  "hexMatrix": [ "0xA2", "0xC4", "0xE2", "0x12", "0xB6" ]
}
```

Information supplémentaire à propos de la représentation d'une observation météo

Information	Validation
Nom de la station	Doit être indexé et en majuscule
Température	En degré Celsius ¹
Température ressentie	En degré Celsius ¹
Index ultra-violet	Dois être compris entre 0 et 11
Direction du vent	Dois être comprise entre 0 et 360
Couverture nuageuse	Dois être comprise entre 0 et 1
Date d'observation	Valeur par défaut: la date courante

REPRÉSENTATION JSON DE RÉPONSE

Votre service d'échange doit répondre aux différents clients avec la représentation JSON suivante :

```
{
  "location": {
    "coord": {
      "lon": -122.08,
      "lat": 37.39
    },
    "station": "XL908"
  },
  "wind": {
    "speed": 12,
    "degree": 165,
    "direction": "SE"
  },
  "clouds": {
    "cloudcover": 0.72
  },
  "_id": "5f543f935dd6856534c654bb",
  "temperature": 15,
  "pressure": 1011,
  "humidity": 0.55,
  "feelslike": 15,
  "uvIndex": 5,
  "observationDate": "2020-06-28T11:39:13.934Z",
  "hex": {
    "alpha": 784,
    "beta": 11754209376000,
    "gamma": 14992614000,
    "delta": 0
  }
}
```

BASE DE DONNÉES

Ajouter une nouvelle collection nommée : `weather-observations` qui permettra de sauvegarder les observations météo dans votre base de données NOSQL (<https://cloud.mongodb.com/>). Votre collection doit contenir au minimum 10 observations météo lors de la remise.

¹ Vous n'avez pas de validation particulière à faire, mais ceci aura une importance pour certaines routes

SERVICES D'ÉCHANGE DE DONNÉES

Vous devez développer les routes suivantes pour la ressource `weather-observations` dans votre application serveur

WEATHER-OBSERVATIONS

Ajouter une observation météo

URL	/weather-observations
Paramètres d'URL	Aucun
Corps de requête	La représentation JSON du document de l'observation météo à ajouter
Succès	201 – Created La représentation JSON du document de l'observation météo venant d'être ajoutée
Échec	400 – Bad Request 500 – Internal Server Error

Supprimer une observation spécifique

URL	/weather-observations/:idWeatherObservation
Paramètres d'URL	Aucun
Corps de requête	Aucun
Succès	Aucun
Échec	405 – Method Not Allowed

Obtenir la liste des observations d'une station

URL	/weather-observations/:stationName		
Paramètres d'URL	unit	Les valeurs représentant des températures (temperature, feelslike) doivent être retournées au client dans l'unité choisi.	
		Valeur	Signification
		Unité	
	m	Métrique	Celsius
	s	Scientifique	Kelvin
	f	Fahrenheit	Fahrenheit
Corps de requête	Aucun		
Succès	200 – OK La représentation JSON de la collection des observations météo de la station demandée		
Échec	400 – Bad Request		
	500 – Internal Server Error		

Obtenir une observation spécifique d'une station

URL	/weather-observations/:stationName/:idWeatherObservation		
Paramètres d'URL	unit	Les valeurs représentant des températures (temperature, feelslike) doivent être retournées au client dans l'unité choisi.	
	Valeur	Signification	Unité
	m	Métrique	Celsius
	s	Scientifique	Kelvin
	f	Fahrenheit	Fahrenheit
Corps de requête	Aucun		
Succès	200 – OK La représentation JSON du document de l'observation météo spécifique demandé		
Échec	400 – Bad Request 404 – Not Found 500 – Internal Server Error		

TRANSFORMATION DE RÉPONSE

UNIT

Le paramètre d'URL *unit* permet au client de choisir l'unité de mesure qu'il désire recevoir pour les valeurs représentant des températures (*temperature*, *feelslike*). Selon le petit tableau suivant :

Valeur	Signification	Unité
m	Métrique	Celsius
s	Scientifique	Kelvin
f	Fahrenheit	Fahrenheit

Votre service d'échange de données devra donc offrir cette possibilité, il faudra donc programmer les conversions entre les unités. Les données en lien avec les températures sont sauvegardées en degré Celsius. Il faut donc trouver les équations pour transformer des degrés Celsius en Kelvin et de transformer de degrés Celsius en Fahrenheit.

À vous de trouver les formules nécessaires, vous pouvez les valider avec moi si vous n'êtes pas certain.

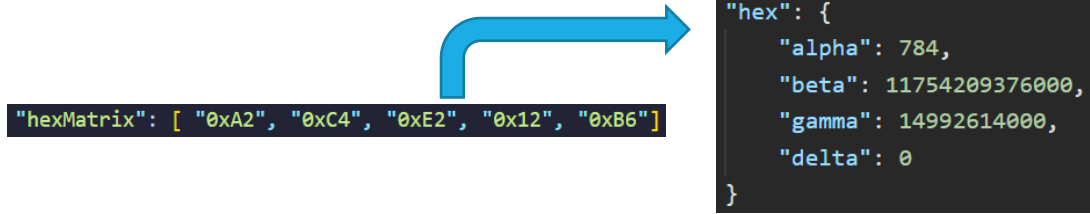
WIND DIRECTION

Votre réponse doit comprendre la direction du vent dans la propriété *direction* de l'objet *wind* de la représentation de réponse. Les valeurs possibles en fonction de l'orientation du vent sont définies dans la colonne abréviation dans le tableau suivant :

Point cardinal	Abréviation	Intervalle
Nord	N	[0;45[
Nord-Est	NE	[45;90[
Est	E	[90;135[
Sud-Est	SE	[135;180[
Sud	S	[180;225[
Sud-Ouest	SW	[225;270[
Ouest	W	[270;315[
Nord-Ouest	NW	[315;360[

HEX-MATRIX

La propriété `hexMatrix` est un tableau de chaînes de caractères représentant des nombres en hexadécimal. Votre serveur ne doit jamais retourner la propriété `hexMatrix` d'une observation météo, mais plutôt la transformer en objet `hex` en suivant les règles suivantes :



`alpha` : La somme de tous les éléments du `hexMatrix`

`beta` : Le produit de tous les éléments du `hexMatrix`

`gamma` : Le quotient de `beta` sur `alpha`

`delta` : Le reste de `beta` sur `alpha`

TESTS

Vous devez remettre un fichier provenant d'une collection Postman, permettant de tester toutes vos routes de manière individuelle. Vous devez tester le succès et les différentes erreurs de chacune des routes. Le fichier doit se nommer **tests.json** et être à la racine de votre projet.

PONDÉRATION

Ce travail compte pour **17,5%** de la note finale du cours, les points seront distribués de cette manière :

Critères de correction	
Respect intégral de la représentation de la ressource weather-observations	/10
Fonctionnement de la route Ajouter une observation météo conforme aux exigences	/15
Fonctionnement de la route Supprimer une observation spécifique conforme aux exigences	/5
Fonctionnement de la route Obtenir la liste des observations d'une station conforme aux exigences	/20
Fonctionnement de la route Obtenir une observation spécifique d'une station conforme aux exigences	/15
Création adéquate de la collection dans la base de données	/2,5
Présence des données initiales dans la collection de la base de données conformes aux exigences	/2,5
Qualité et niveau de complétude des tests fournis (tests.json)	/5
Qualité de la structure du projet, du code et des commentaires	/12,5
Total	/87,5
Total pondéré (Total ÷ 5)	/17,5