

2.1 Insertion

- Créer un document (changer les valeurs qui vous concerne dans un éditeur) :

```
lieu = {
  "_id" : 1234567890,
  "name" : "METTRE LE NOM DE VOTRE ETABLISSEMENT",
  "reviews" : [{"wordsCount" : 32,"rating" : 5,"language" : "fr",
    "text" : "LE TEXTE QUE VOUS POUVEZ METTRE ICI"}
  ],
  "contact" : {
    "website" : "URL DE VOTRE ETABLISSEMENT",
    "GooglePlaces" : "IDEM",
    "phone" : "CONTACT",
    "foursquare" : "URL FOURSQUARE",
    "Booking" : "",
    "Facebook" : "PAGE FACEBOOK"
  },
  "description" : {
    "fr" : "DESCRIPTION DE VOTRE ETABLISSEMENT",
    "de" : "", "en" : "", "it" : "", "nl" : "", "es" : ""
  },
  "location" : {
    "coord" : {"coordinates" : [LATITUDE, LONGITUDE], "type" : "Point"},
    "address" : "ADRESSE",
    "city" : "VILLE"
  },
  "services" : [ ], "category" : "poi"
};
```

- Insérer le document dans la collection “paris” :

```
db.paris.save(lieu);
```

- Consulter le résultat :

```
db.paris.find({"_id" : 1234567890}).pretty();
```

2.2 Update/Delete

Une fois la base intégralement téléchargée, vous pourrez faire des mises à jour, avec les instructions suivantes :

```
db.paris.update (
  {"_id" : 1234567890},
  {$set : { "type" : "Etablissement" } }
);
//Premier JSon : Mapping, Second JSon : Update ($set, $unset)
```

```
db.paris.find({"type":{"$exists:1}});  
//Vérifier quels documents ont été modifiés
```

```
db.paris.remove({"_id" : 1234567890});  
//Mapping de suppression
```

```
db.paris.find({"type":{"$exists:1}});  
//Vérifier si les documents existent encore
```

- Il est possible de faire des fonctions itératives (javascript) sur le résultat du `find` :

```
db.paris.find( {"reviews.1":{"$exists:1}} ).forEach(  
  function(lieu){  
    lieu.nb_reviews = lieu.reviews.length;  
    db.paris.save(lieu);  
  }  
);  
//cela peut prendre un peu de temps pour calculer cela sur chaque document.
```

- Tester le résultat :

```
db.paris.aggregate([  
  {$group : { _id : "$nb_reviews", "nb" : {$sum :1}}},  
  {$sort : {"nb" : -1}}  
]);  
  
>>>>>  
{ "_id" : null, "nb" : 45785 }  
{ "_id" : 5, "nb" : 3145 }  
{ "_id" : 2, "nb" : 2395 }  
{ "_id" : 3, "nb" : 1541 }  
{ "_id" : 4, "nb" : 1032 }  
{ "_id" : 6, "nb" : 203 }  
{ "_id" : 7, "nb" : 171 }  
{ "_id" : 8, "nb" : 169 }  
{ "_id" : 9, "nb" : 148 }  
{ "_id" : 13, "nb" : 133 }  
{ "_id" : 12, "nb" : 131 }  
{ "_id" : 10, "nb" : 119 }  
//45785 documents n'ont pas de reviews (pas de mises à jour effectuées),  
//la plupart des lieux ont entre 4 et 5 commentaires.
```

2.3 Exercices

Pour les mises à jour suivantes, vérifier le contenu des données avant et après la mise à jour.

- 2.3.1 Mettre à jour tous les noms de lieux contenant “montparnasse” en ajoutant l’attribut “area” : “Montparnasse” (pour plusieurs mises à jour, l’option “multi :true” est à utiliser);

Correction :

```
db.paris.update ({"name": {"$regex" : "montparnasse", "$options" : "i"}},
    {$set : {"area" : "Montparnasse"}},
    {"multi": true});
//test
db.paris.find ({"area": "Montparnasse"}, {"name": 1});
```

2.3.2 Supprimer la clé “nb_reviews” de tous les articles;

Correction :

```
db.paris.update({"area" : {$exists:1}},
    {$unset : {"area" : 1}},
    {"multi": true});
//attention à ne pas oublier le $unset.
//Sinon, on remplace tous les documents par un nouveau document (celui en 2° paramètre)
```

2.3.3 Supprimer tous les lieux n’ayant aucun commentaire;

Correction :

```
db.paris.remove ({"reviews" : []});
```

2.3.4 Modifier tous les lieux pour ajouter un champ “wordsCount” qui correspond à la somme du nombre de mots de tous les commentaires (reviews.wordsCount)

Correction :

```
db.paris.find({"reviews.0":{$exists : 1}}).forEach(
    function(lieu){
        wordsCount = 0;
        lieu.reviews.forEach ( function (r){
            wordsCount += r.wordsCount;
        });
        lieu.wordsCount = wordsCount;
        db.paris.save(lieu);
    }
);
```