

Seeding #2

1. CSVTOJSON (csvtojson.js):

```
1  const csvFilePath='listings.csv';
2  const csv=require('csvtojson');
3  const fs= require('fs');
4
5  csv().fromFile(csvFilePath).then((jsonObj)=> {
6    fs.writeFile('listings.json', JSON.stringify(jsonObj),'utf8', (err) => {
7      if (err) throw err;
8      console.log('The file has been saved!');
9    });
10 })
```

Figura 1 CSVTOJSON –Por el autor

2. Ubicar el archivo “Listings.json” en la carpeta parara el Seeding

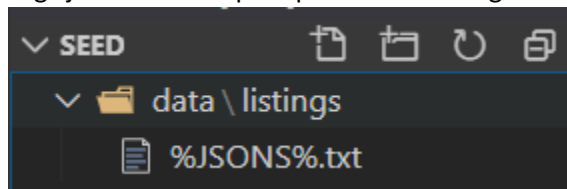


Figura 2 JSON –Por el autor

3. Run Seed (seed.js):

```
1  const { Seeder } = require('mongo-seeding');
2  const config = {
3    database: {
4      host: '127.0.0.1',
5      port: 27017,
6      name: 'airbnb',
7    },
8    dropDatabase: true,
9  };
10 const seeder = new Seeder(config);
11 const path = require('path');
12 const collections = seeder.readCollectionsFromPath(path.resolve("./data"));
13 console.log(collections);
14
15 seeder
16   .import(collections)
17   .then(() => {
18     console.log("Seed exitoso");
19   })
20   .catch(err => {
21     console.log(err);
22   });
```

Figura 3 SEED –Por el autor

4. Después se realizan las consultas (query.js):

```
//Descargar un dataset de airbnb y realizar las siguientes consultas:

// Alojamiento más barato
db.listings.find({}, {"price": true, "name": true}).limit(1).pretty().sort({"price": 1});
// Alojamiento con precio más alto
db.listings.find({}, {"price": true, "name": true}).limit(1).pretty().sort({"price": -1});
// Alojamiento con la mejor calificación en el puntaje de los comentarios (reviews)
db.listings.find({}, {"review_scores_rating": true, "name": true}).limit(1).pretty().sort({"review_scores_rating": -1});
// Alojamiento con depósito de seguridad y tarifa de limpieza más baratos (menor x, con x=depósito de seguridad + tarifa de limpieza)

// Alojamientos que cuenten con microondas, TV y precio menor a $500
db.acomodaciones.find({$and: [{amenities: /TV/}, {amenities: /Microwave/}, {price: {$lt: "$500"} }]});
// Media de precios de todos los alojamientos que cuenten con microondas
db.listings.aggregate([{$match: {}}, {$group: { } }]);
// Alojamientos con menor cantidad de noches mínimas, cantidad máxima de noches mayor a 15, que cuenten con TV y costo menor a $200
db.acomodaciones.find({$and: [{maximum_nights: {$gt: "15"}}, {amenities: /TV/}, {price: {$lt: "$200"} }]}).sort({minimum_nights: 1}) ;
```

Figura 2 Querys –Por el autor