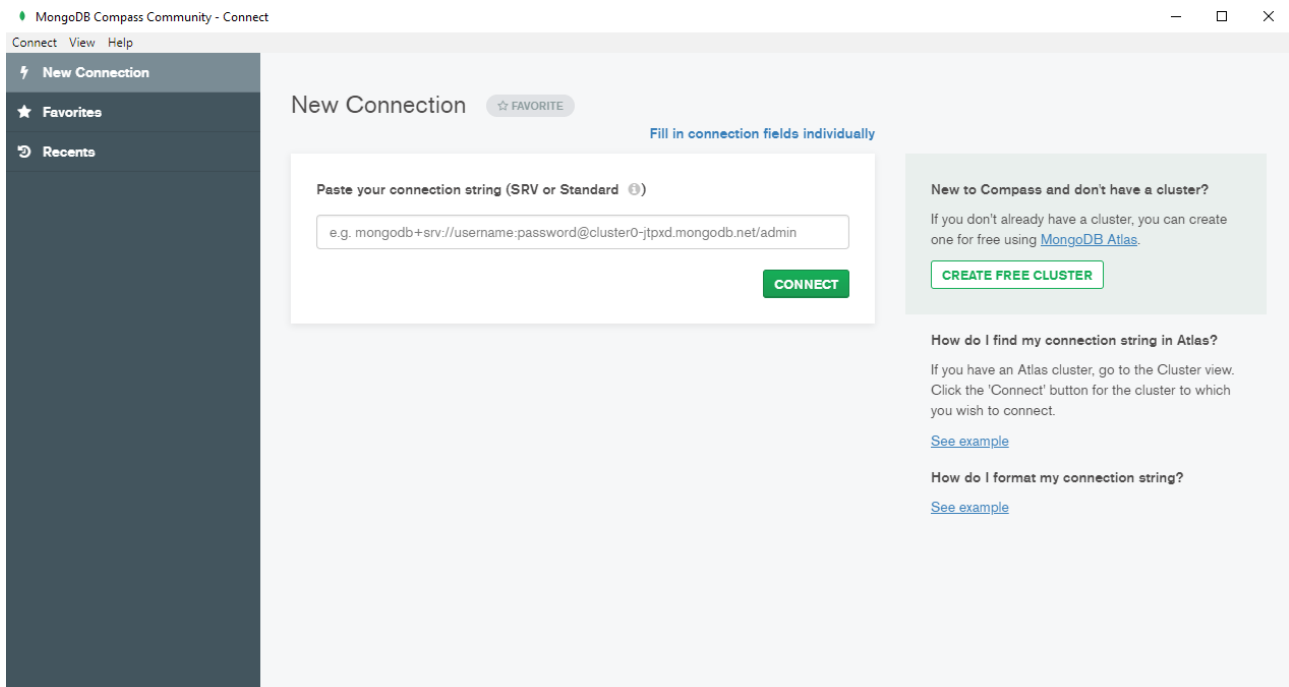
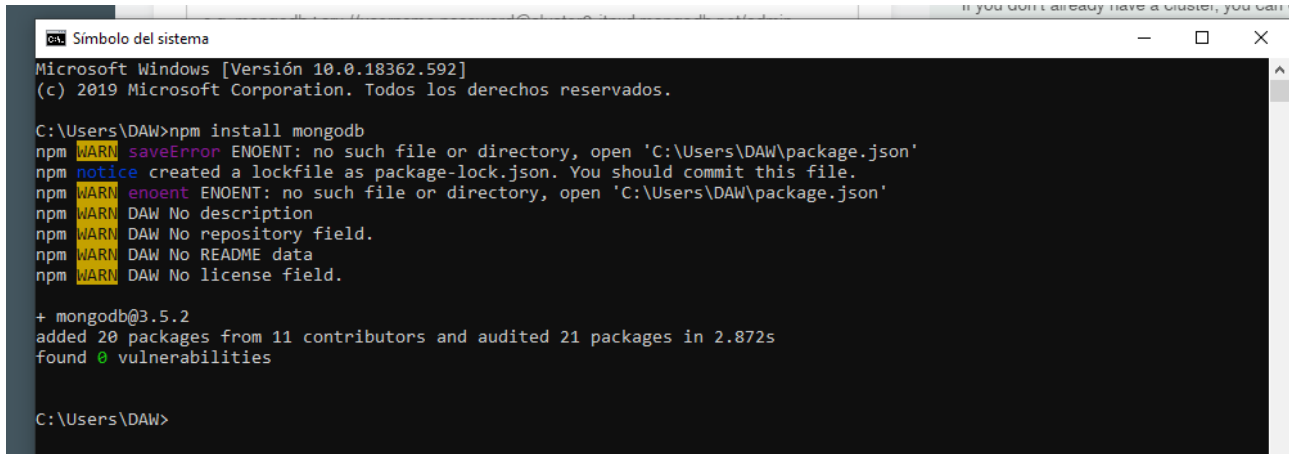


Primeros pasos con Node.js y MongoDB

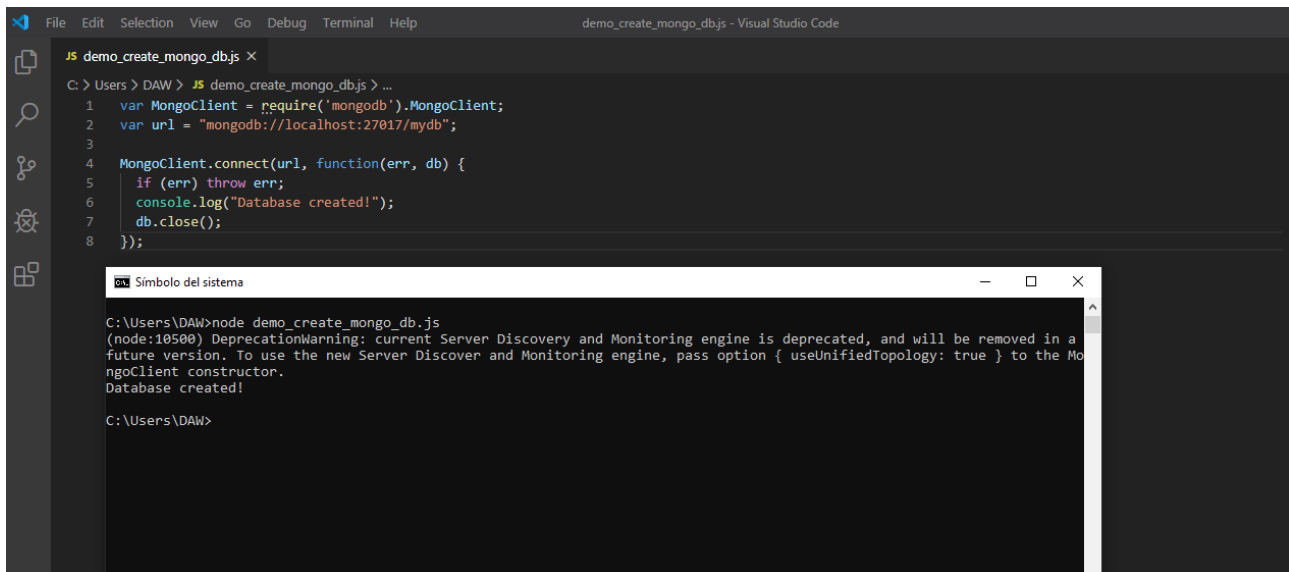
Instalamos MongoDB



Instalamos el controlador de MongoDB



A continuación creamos una base de datos



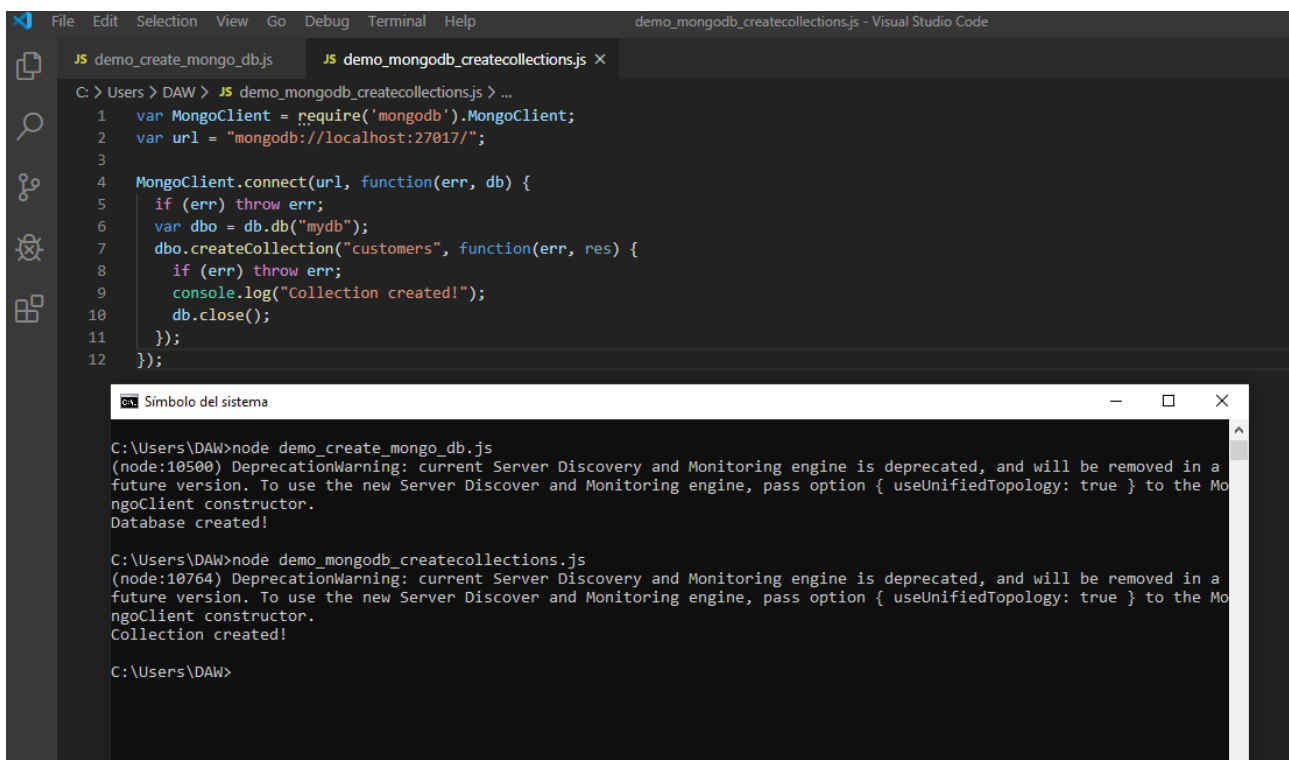
The screenshot shows the Visual Studio Code interface with a file named `demo_create_mongo_db.js`. The code in the editor is as follows:

```
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/mydb";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   console.log("Database created!");
7   db.close();
8 });
```

Below the editor, a terminal window titled "Símbolo del sistema" shows the command `C:\Users\DAW>node demo_create_mongo_db.js` being executed. The output is:

```
(node:10500) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
Database created!
C:\Users\DAW>
```

Ahora creamos una colección:



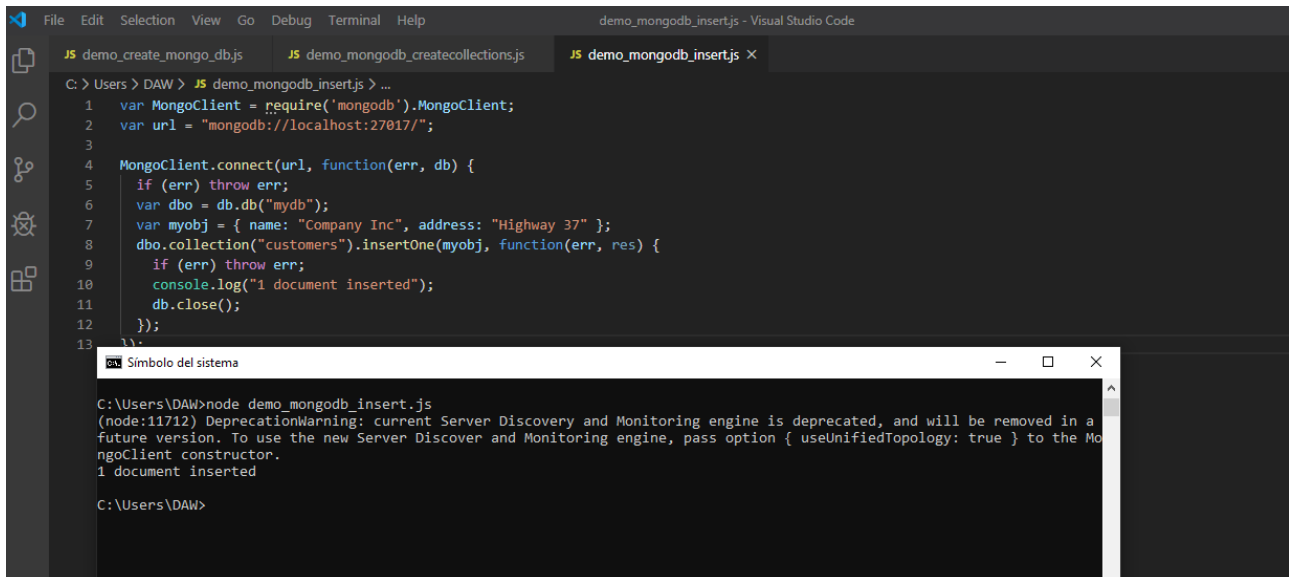
The screenshot shows the Visual Studio Code interface with two files: `demo_create_mongo_db.js` and `demo_mongodb_createcollections.js`. The code in `demo_mongodb_createcollections.js` is as follows:

```
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("mydb");
7   dbo.createCollection("customers", function(err, res) {
8     if (err) throw err;
9     console.log("Collection created!");
10    db.close();
11  });
12 });
```

Below the editor, a terminal window titled "Símbolo del sistema" shows the command `C:\Users\DAW>node demo_mongodb_createcollections.js` being executed. The output is:

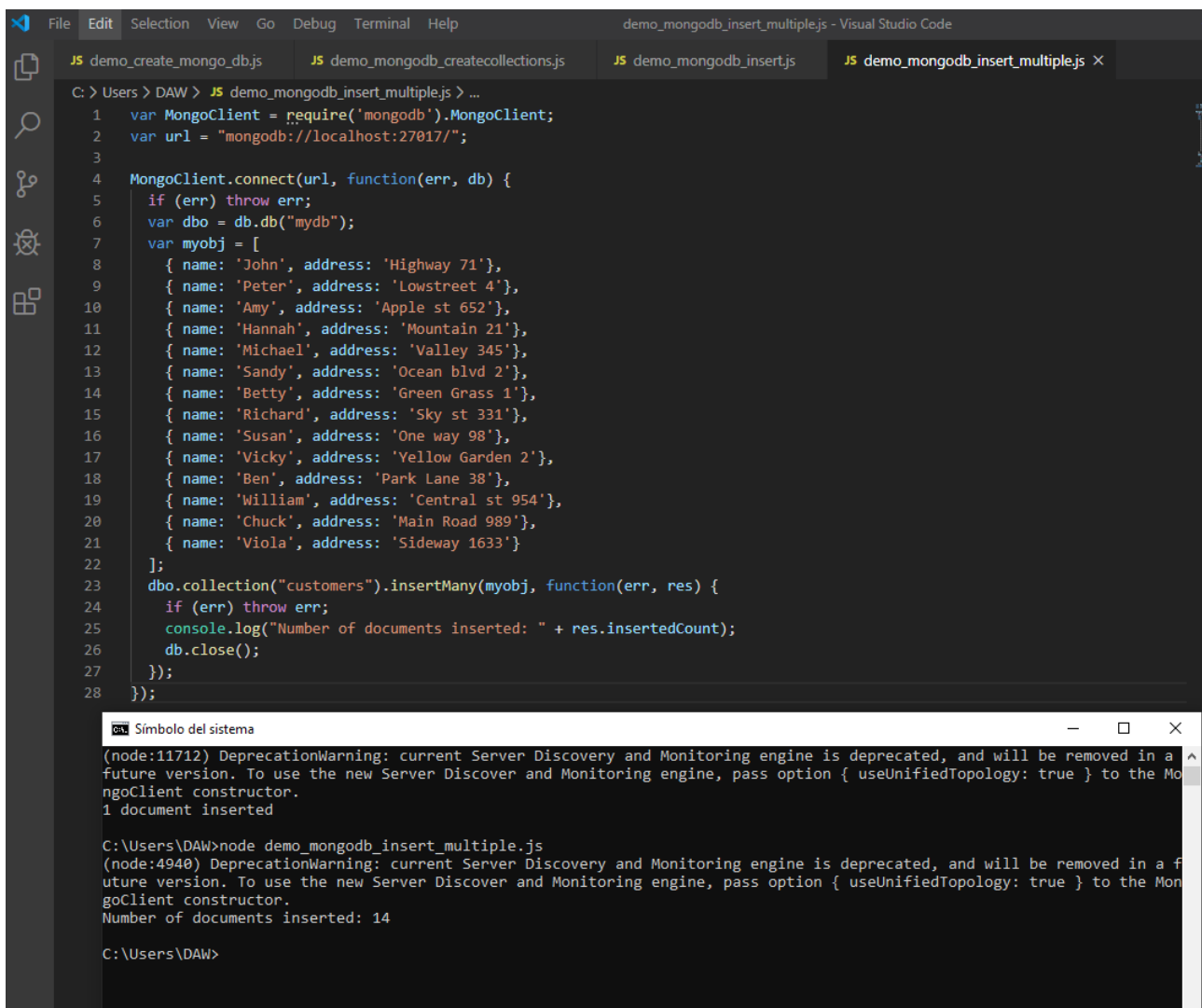
```
(node:10764) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
Collection created!
C:\Users\DAW>
```

Creamos también los inserts para introducir datos en la colección



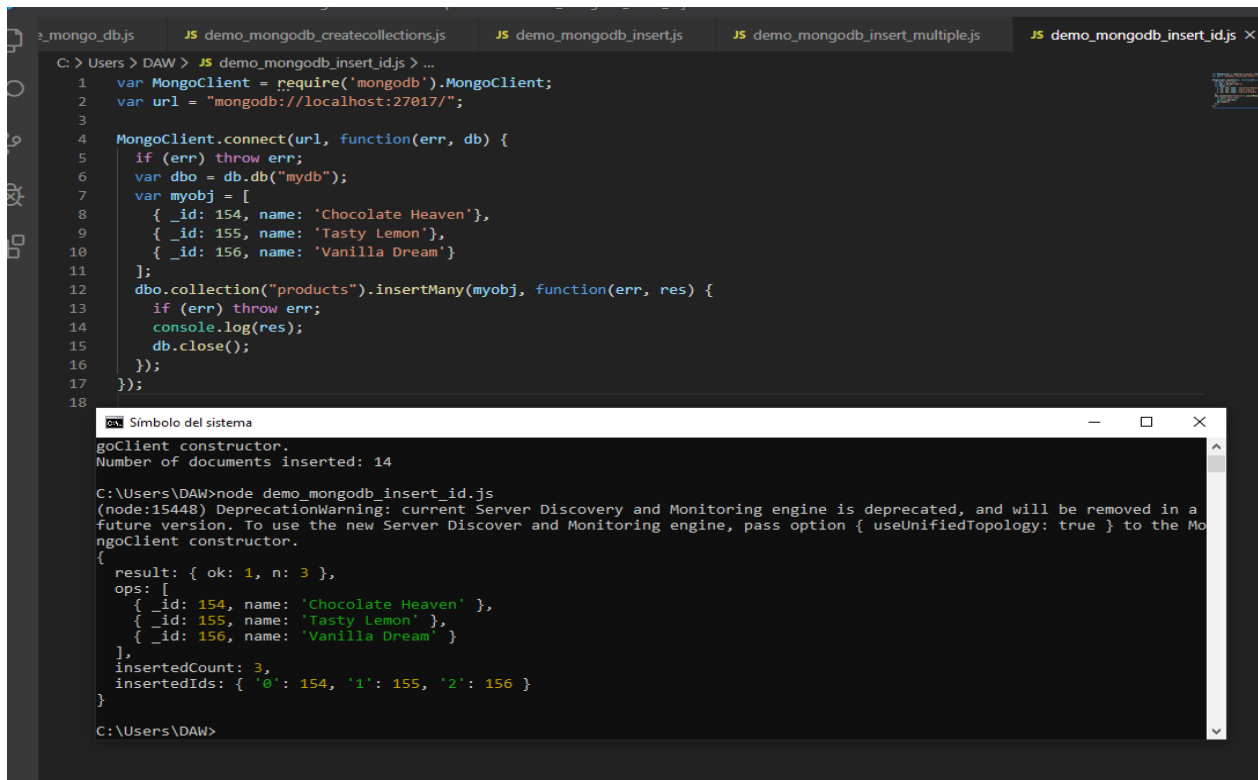
```
File Edit Selection View Go Debug Terminal Help
demo_mongodb_insert.js - Visual Studio Code
JS demo_create_mongo_db.js JS demo_mongodb_createcollections.js JS demo_mongodb_insert.js X
C:\Users\DAW> node demo_mongodb_insert.js
1 var MongoClient = require('mongodb').MongoClient;
2 var url = 'mongodb://localhost:27017/';
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("mydb");
7   var myobj = { name: "Company Inc", address: "Highway 37" };
8   dbo.collection("customers").insertOne(myobj, function(err, res) {
9     if (err) throw err;
10    console.log("1 document inserted");
11    db.close();
12  });
13 })
14
C:\Users\DAW>
```

Insertamos múltiples documentos



```
File Edit Selection View Go Debug Terminal Help
demo_mongodb_insert_multiple.js - Visual Studio Code
JS demo_create_mongo_db.js JS demo_mongodb_createcollections.js JS demo_mongodb_insert.js JS demo_mongodb_insert_multiple.js X
C:\Users\DAW> node demo_mongodb_insert_multiple.js
1 var MongoClient = require('mongodb').MongoClient;
2 var url = 'mongodb://localhost:27017/';
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("mydb");
7   var myobj = [
8     { name: 'John', address: 'Highway 71'},
9     { name: 'Peter', address: 'Lowstreet 4'},
10    { name: 'Amy', address: 'Apple st 652'},
11    { name: 'Hannah', address: 'Mountain 21'},
12    { name: 'Michael', address: 'Valley 345'},
13    { name: 'Sandy', address: 'Ocean blvd 2'},
14    { name: 'Betty', address: 'Green Grass 1'},
15    { name: 'Richard', address: 'Sky st 331'},
16    { name: 'Susan', address: 'One way 98'},
17    { name: 'Vicky', address: 'Yellow Garden 2'},
18    { name: 'Ben', address: 'Park Lane 38'},
19    { name: 'William', address: 'Central st 954'},
20    { name: 'Chuck', address: 'Main Road 989'},
21    { name: 'Viola', address: 'Sideway 1633'}
22  ];
23  dbo.collection("customers").insertMany(myobj, function(err, res) {
24    if (err) throw err;
25    console.log("Number of documents inserted: " + res.insertedCount);
26    db.close();
27  });
28 })
29
C:\Users\DAW> node demo_mongodb_insert_multiple.js
(node:4940) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
Number of documents inserted: 14
C:\Users\DAW>
```

Especificamos el campo ID



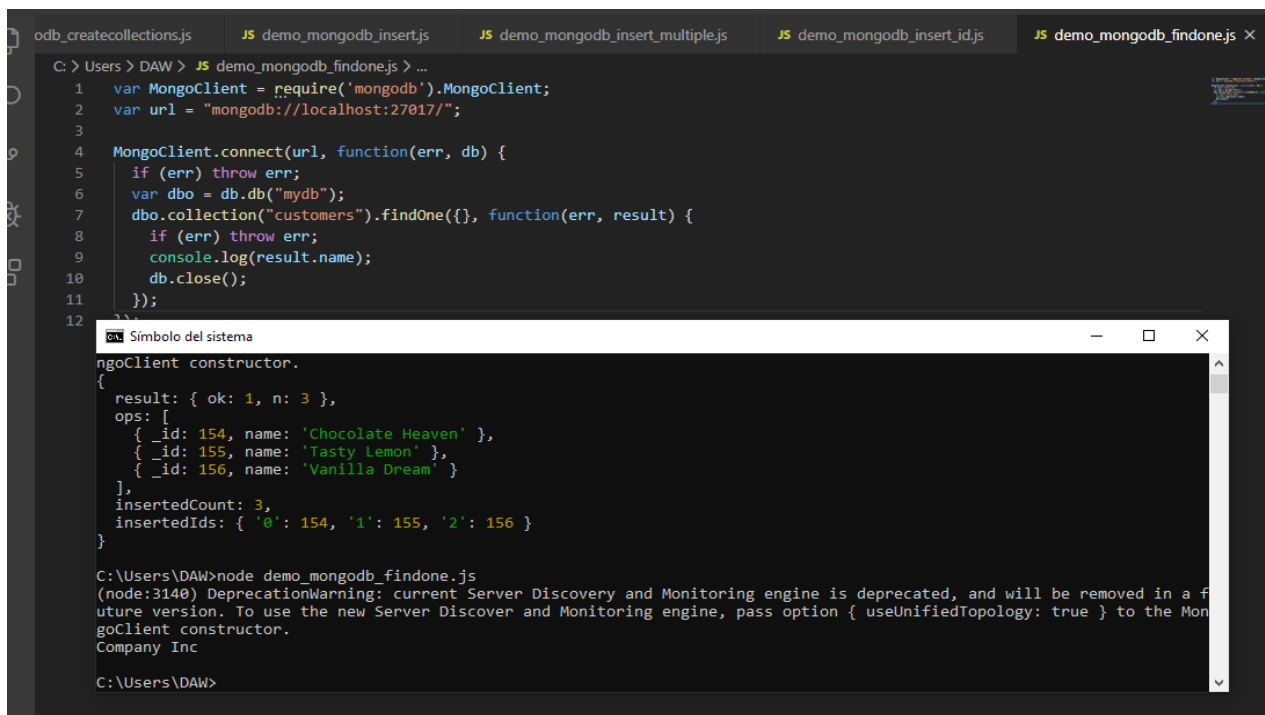
```
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("mydb");
7   var myobj = [
8     { _id: 154, name: 'Chocolate Heaven' },
9     { _id: 155, name: 'Tasty Lemon' },
10    { _id: 156, name: 'Vanilla Dream' }
11  ];
12  dbo.collection("products").insertMany(myobj, function(err, res) {
13    if (err) throw err;
14    console.log(res);
15    db.close();
16  });
17 });
```

```
goClient constructor.
Number of documents inserted: 14

C:\Users\DAW>node demo_mongodb_insert_id.js
(node:15448) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
{
  result: { ok: 1, n: 3 },
  ops: [
    { _id: 154, name: 'Chocolate Heaven' },
    { _id: 155, name: 'Tasty Lemon' },
    { _id: 156, name: 'Vanilla Dream' }
  ],
  insertedCount: 3,
  insertedIds: { '0': 154, '1': 155, '2': 156 }
}

C:\Users\DAW>
```

Buscamos el objeto



```
1 var MongoClient = require('mongodb').MongoClient;
2 var url = "mongodb://localhost:27017/";
3
4 MongoClient.connect(url, function(err, db) {
5   if (err) throw err;
6   var dbo = db.db("mydb");
7   dbo.collection("customers").findOne({}, function(err, result) {
8     if (err) throw err;
9     console.log(result.name);
10    db.close();
11  });
12 });
```

```
goClient constructor.
{
  result: { ok: 1, n: 3 },
  ops: [
    { _id: 154, name: 'Chocolate Heaven' },
    { _id: 155, name: 'Tasty Lemon' },
    { _id: 156, name: 'Vanilla Dream' }
  ],
  insertedCount: 3,
  insertedIds: { '0': 154, '1': 155, '2': 156 }
}

C:\Users\DAW>node demo_mongodb_findone.js
(node:3140) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
Chocolate Heaven

C:\Users\DAW>
```

Buscamos varios objetos

```
C:\Users\DAW> node demo_mongodb_find.js > ...
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("mydb");
7    dbo.collection("customers").find({}).toArray(function(err, result) {
8      if (err) throw err;
9      console.log(result);
10     db.close();
11   });
12 }
```

Simbolo del sistema

```
C:\Users\DAW>node demo_mongodb_find.js
(node:9412) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
[
  {
    _id: 5e273e57297c492dc0d3a14d,
    name: 'Company Inc',
    address: 'Highway 37'
  },
  {
    _id: 5e273ef9a58d40134cc7089d,
    name: 'John',
    address: 'Highway 71'
  },
  {
    _id: 5e273ef9a58d40134cc7089e,
    name: 'Peter',
    address: 'Lowstreet 4'
  },
  {
    _id: 5e273ef9a58d40134cc7089f,
    name: 'Amy',
    address: 'Apple st 652'
  },
  {
    _id: 5e273ef9a58d40134cc708a0,
    name: 'Hannah',
    address: 'Mountain 21'
  },
]
```

Buscamos todos los objetos

```
o_mongodb_insert_multiple.js JS demo_mongodb_insert_id.js JS demo_mongodb_findone.js JS demo_mongodb_find.js JS demo_mongodb_find_fields.js X
C:\Users\DAW> JS demo_mongodb_find_fields.js > ...
1  var MongoClient = require('mongodb').MongoClient;
2  var url = "mongodb://localhost:27017/";
3
4  MongoClient.connect(url, function(err, db) {
5    if (err) throw err;
6    var dbo = db.db("mydb");
7    dbo.collection("customers").find({}, { projection: { _id: 0, name: 1, address: 1 } }).toArray(function(err, result) {
8      if (err) throw err;
9      console.log(result);
10     db.close();
11   });
12 }

```

```

},
{
  _id: 5e273ef9a58d40134cc708aa,
  name: 'Viola',
  address: 'Sideway 1633'
}
]
C:\Users\DAW>node demo_mongodb_find_fields.js
(node:8108) DeprecationWarning: current Server Discovery and Monitoring engine is deprecated, and will be removed in a future version. To use the new Server Discover and Monitoring engine, pass option { useUnifiedTopology: true } to the MongoClient constructor.
[
  { name: 'Company Inc', address: 'Highway 37' },
  { name: 'John', address: 'Highway 71' },
  { name: 'Peter', address: 'Lowstreet 4' },
  { name: 'Amy', address: 'Apple st 652' },
  { name: 'Hannah', address: 'Mountain 21' },
  { name: 'Michael', address: 'Valley 345' },
  { name: 'Sandy', address: 'Ocean blvd 2' },
  { name: 'Betty', address: 'Green Grass 1' },
  { name: 'Richard', address: 'Sky st 331' },
  { name: 'Susan', address: 'One way 98' },
  { name: 'Vicky', address: 'Yellow Garden 2' },
  { name: 'Ben', address: 'Park Lane 38' },
  { name: 'William', address: 'Central st 954' },
  { name: 'Chuck', address: 'Main Road 989' },
  { name: 'Viola', address: 'Sideway 1633' }
]
C:\Users\DAW>

```