

Node.js: usare MongoDB in Fastify

In questo articolo vedremo come connettere Fastify a MongoDB in Node.js.

Installiamo i plugin richiesti:

```
npm install fastify-plugin fastify-mongodb --save
```

Creiamo un file per registrare l'istanza della nostra connessione al database MongoDB.

```
'use strict';

const fastifyPlugin = require('fastify-plugin');

async function dbConnector (fastify, options) {
  fastify.register(require('fastify-mongodb'), {
    url: 'mongodb://localhost:27017/test'
  });
}

module.exports = fastifyPlugin(dbConnector);
```

Ora includiamo tale istanza nella nostra applicazione:

```
'use strict';

const fastify = require('fastify')({
  logger: true
});

const PORT = process.env.PORT || 3000;
```

```

fastify.register(require('./lib/dbconnector'));
fastify.register(require('./routes/'));

fastify.listen(PORT, (err, address) => {
  if (err) {
    fastify.log.error(err);
    process.exit(1);
  }
});

```

L'utilizzo dell'istanza della connessione viene utilizzato come segue nelle route:

```

'use strict';

async function routes (fastify, options) {
  const posts =
fastify.mongo.db.collection('posts');

  fastify.get('/posts', async (request, reply) => {
    const result = await posts.find().toArray();
    if (result.length === 0) {
      throw new Error('No posts found.');
```

```

    }
    return result;
  });
}

module.exports = routes;

```

Come si può notare, l'istanza della connessione viene registrata dal framework direttamente nell'oggetto `fastify` e quindi disponibile in tutte le

nostre route.