

Node.js: effettuare una richiesta WHOIS tramite socket

Di seguito vedremo come verificare se un dominio è disponibile per la registrazione con Node.js effettuando una richiesta WHOIS.

Invece di usare il comando `whois` per ottenere questa informazione con il parametro `-h` per specificare un server WHOIS dove effettuare la richiesta, effettueremo la richiesta creando una connessione socket verso il server WHOIS.

La risposta del server WHOIS conterrà solo una determinata stringa nel caso in cui il dominio non è ancora stato registrato. Possiamo creare un nostro array di oggetti contenenti le informazioni dei vari server WHOIS. Ad esempio per i domini `.it` avremo:

```
{
  extension: '.it',
  server: 'whois.nic.it',
  response: 'AVAILABLE'
}
```

La stringa della proprietà `response` sarà diversa da server a server e la sua presenza o meno nella risposta indicherà se il dominio è disponibile per la registrazione.

Installiamo il modulo NPM `validator` per validare il formato dei nomi a dominio:

```
npm install validator --save
```

Creiamo una funzione helper che da un nome a dominio ne estragga l'estensione tenendo conto della doppia estensione.

```
const getExtension = domain => {  
  const dotFirst = domain.indexOf('.');  
  return domain.substring(dotFirst);  
};
```

Ora creiamo la funzione per eseguire la richiesta WHOIS con i parametri richiesti.

```
const sendRequest = ({ host, domain }) => {  
  const net = require('net');  
  const client = new net.Socket();  
  
  return new Promise((resolve, reject) => {  
  
    client.connect(43, host, () => {  
      client.write(`${domain}\r\n`);  
    });  
  
    client.on('data', data => {  
      resolve(data.toString());  
      client.destroy();  
    });  
  });  
};
```

Implementiamo una classe che, dato un nome a dominio, ne verificherà la validità sintattica e la presenza dell'estensione nel nostro array di server WHOIS ed infine effettuerà la richiesta restituendo lo stato del dominio.

```
'use strict';

const validator = require('validator');
const data = require('../src/data');

class WhoisClient {
  constructor(domain) {
    this.domain = domain;
  }

  async request() {
    let response = {
      result: ''
    };
    if(!validator.isFQDN(this.domain)) {
      response.result = 'Invalid domain name.';
      return response;
    }

    const extension = getExtension(this.domain);

    const datum = data.find(d => {
      return d.extension === extension;
    });

    if(!datum) {
      response.result = 'Invalid domain
extension.';
      return response;
    }

    try {
      const output = await sendRequest({
        host: datum.server,
```

```

        domain: this.domain
    });

    response.result =
output.includes(datum.response) ? 'Available' : 'Not
Available';

    } catch(err) {
        response.result = 'Error.';
    }

    return response;
}
}

module.exports = WhoisClient;

```

Esempio d'uso:

```

'use strict';

const express = require('express');
const bodyParser = require('body-parser');
const PORT = process.env.PORT || 3000;
const WhoisClient = require('./lib/WhoisClient');
const app = express();

app.use(bodyParser.json());
app.use(bodyParser.urlencoded({ extended: true
}));

app.post('/whois', async (req, res) => {
    const domain = req.body.domain ?

```

```
req.body.domain : '';
    try {
        const response = await new
WhoisClient(domain).request();
        res.json(response);
    } catch(err) {
        res.json(err);
    }
});

app.listen(PORT);
```