Node.js: creare redirect HTTP

I redirect HTTP sono un modo per i server web di informare un client che la risorsa richiesta è stata spostata in una nuova location. I redirect vengono eseguiti inviando una risposta con un codice di stato che indica che la risorsa richiesta è stata spostata, insieme ad un header Location che specifica la nuova posizione. In Node.js, i redirect HTTP possono essere implementati usando i moduli `http` e `https`.

Esistono diversi tipi di redirect HTTP, ognuno con un codice di stato e un significato differente. I tipi di redirect più comuni sono 301 (Spostato permanentemente), 302 (Trovato) e 307 (Redirect temporaneo). In Node.js, è possibile utilizzare la proprietà `statusCode` dell'oggetto di risposta per specificare il codice di stato del redirect.

Ecco un esempio di come implementare un redirect 301 in Node.js usando il modulo `http`:

```
'use strict';

const http = require('http');

const server = http.createServer((req, res) => {
  res.statusCode = 301;
  res.setHeader('Location',
    'http://www.example.com');
  res.end();
});
server.listen(3000);
```

In questo esempio, quando un client effettua la richiesta riceverà una risposta di redirect 301 con un header HTTP Location impostato su `http://www.example.com`. Ciò farà sì che il client richieda automaticamente la nuova location.

Se si desidera eseguire un redirect temporaneo, è possibile utilizzare invece il codice di stato 307:

```
'use strict';

const http = require('http');

const server = http.createServer((req, res) => {
  res.statusCode = 307;
  res.setHeader('Location',
    'http://www.example.com');
  res.end();
});
server.listen(3000);
```

I redirect HTTP sono spesso utilizzati per migliorare la user experience reindirizzando gli utenti sugli URL corretti. Ad esempio, è possibile utilizzare i redirect per reindirizzare gli utenti da vecchi URL abbandonati ai nuovi, o per reindirizzarli da pagine senza HTTPS a pagine con HTTPS.