

# Node.js: ottenere il cambio di una valuta in Bitcoin

In questo articolo vedremo come ottenere il valore di cambio in Bitcoin per una data valuta in ExpressJS.

Per prima cosa registriamoci su Coinlib e otteniamo una chiave API. Una volta ottenuta tale chiave, salviamola in un file di configurazione.

```
'use strict';

module.exports = {
  apiKey: 'your-api-key'
};
```

A questo punto creiamo una funzione di utility che effettui la richiesta GET remota all'endpoint delle API di Coinlib.

```
'use strict';

const superagent = require('superagent');
const { apiKey } = require('../config');

module.exports = {
  apiRequest( currency ) {
    const url = `https://coinlib.io/api/v1/coin?key=${apiKey}&pref=${currency}&symbol=BTC`;
    return superagent.get(url);
  }
};
```

Stiamo usando il modulo `superagent` al posto del più comune `request` in quanto quest'ultimo è stato deprecato e non viene più sviluppato.

`superagent` fa uso delle `Promise`, quindi rispetto al modulo deprecato risulta essere più flessibile da gestire.

A questo punto possiamo usare questa funzione in una route della nostra applicazione che gestisce una richiesta AJAX da parte del codice lato client.

```
router.post('/value', async (req, res, next) => {
  const { currency } = req.body;
  const validCurrencies = ['USD', 'EUR'];

  if(validCurrencies.includes(currency)) {
    try {
      const response = await
apiRequest(currency);
      const data = JSON.parse(response.text);
      res.json(data);
    } catch(err) {
      res.json({ error: 'Unable to get value.',
info: err });
    }
  } else {
    res.json({ error: 'You must provide a
currency.' });
  }
});

module.exports = router;
```

La route effettua una semplice validazione sul valore della valuta contenuto nella richiesta POST. Nel caso in cui il dato sia corretto, viene effettuata la

richiesta alle API remote. In caso contrario, viene restituito un messaggio di errore.

## **Codice sorgente**

GitHub