var express = require('express');  
var proxy = require('http-proxy-middleware');  
var app = express();  
app.use('/', proxy({ target: 'http://www.baidu.com', changeOrigin: true }));  
app.listen(3000);

原生写法：

const express = require('express');

const path = require('path');

const app = express();

const request = require('request');

// 配置静态文件服务中间件

let serverUrl='http://220.231.2.29:7668';//server地址

app.use(express.static(path.join(\_\_dirname, './')));//静态资源index.html和node代码在一个目录下

app.use('/', function(req, res) {

  let url = serverUrl + req.url;

  req.pipe(request(url)).pipe(res);

});

app.listen(3000,'127.0.0.1', function () {//前端ajax地址写 http://127.0.0.1:3000/

  console.log('server is running at port 3000');

});

<https://www.npmjs.com/package/http-proxy>

据说是优秀的写法：

var http = require("http");var server = http.createServer(function(req, res) {

var opts = {

host: "xxxxx",

port: xxx,

path: req.url,

method: req.method,

headers: req.headers

};

// 如果后端服务需要check host头的话，代理请求头就不能直接透明代理了

// req.headers["host"] = opts.port ? (opts.host+":"+opts.port) : (opts.host)

var proxy = http.request(opts, function(resProxy) {

res.writeHead(resProxy.statusCode, resProxy.headers);

resProxy.pipe(res);

});

req.pipe(proxy); });

server.listen(8080);