```
socket.end('HTTP/1.1 400 Bad Request
                 监听客户端失败事件
                                             \r\n\r\n');
                                             });
                                           server.listen(port, function(){
                 端口监听回调函数
                                              console.log('监听在'+port )
                                                        req.on('data',function(data){
                                监听客户端数据
                                                            console.log('请求的数据:'+data)
                                                      req.on('end',function(data){
                                监听数据结束
                 请求流
                                请求的资源路径
                                                        req.url
                                请求头
                                               req.headers
                                请求方法
                                                 req.method
                                响应给客户端
                                                      res.end()
                                设置编码格式
                                                      res.setHeader('content-type','text/html;charset=utf-8')
                 响应流
                                设置location
                                                     res.setHeader('Location', 'http://baidu.com');
                                                     var date = Date.now()+ 6000;
                                设置过期时间
                                                      res.setHeader('Expires',date)
04_http
                                设置状态码
                                                   res.statusCode = 302;
                               mime:文件的内容类型,举例:text/html,text/
                 mime
                               plain,application/json,iamge/jpeg.
                                                              http.get("地址" , function (res) {
                                                                  res.on("data" , function (data) {
                                       爬取某一个网址
                                                                  res.on("end" , function () {
                                                                 })
                                                                        var images = []
                                                                        var imgReg = /<img.*src="(.+?)".*?>/g
                                      正则匹配html中的img标签
                                                                               html.replace(imgReg, function(a, b) {
                                                                                   imgs.push(b);
                 简单的小爬虫
                                                                       var filename = url.match(/[^\/]+$/)[0];
                                                                              //对图片进行请求
                                                                             http.get(url, function(res) {
                                                                                 //res:服务器响应的数据。
                                       网络图片地址保存到本地
                                                                                 res.on('data', function(data) {
                                                                                    fs.appendFile('./imgs/' + filename, data);
                                                                                 res.on('end', function() {
                                                                                    console.log(filename + '请求完成');
                                                                                 })
                                                                             })
```

server.on('clientError', (err, socket)