

1.6超时

超时

你可以告诉 `requests` 在经过以 `timeout` 参数设定的秒数时间之后停止等待响应:

```
>>> requests.get('http://github.com', timeout=0.001)
```

```
Traceback (most recent call last):
```

```
File "<stdin>", line 1, in <module>
```

```
requests.exceptions.Timeout: HTTPConnectionPool(host='github.com', port=80): Request timed out. (timeout=0.001)
```

注意

`timeout` 仅对连接过程有效，与响应体的下载无关。`timeout` 并不是整个下载响应的时间限制，而是如果服务器在 `timeout` 秒内没有应答，将会引发一个异常（更精确地说，是在 `timeout` 秒内没有从基础套接字上接收到任何字节的数据时）

错误与异常

遇到网络问题（如：DNS 查询失败、拒绝连接等）时，`Requests` 会抛出一个 `ConnectionError` 异常。

如果 HTTP 请求返回了不成功的状态码，`Response.raise_for_status()` 会抛出一个 `HTTPError` 异常。

若请求超时，则抛出一个 `Timeout` 异常。

若请求超过了设定的最大重定向次数，则会抛出一个 `TooManyRedirects` 异常。

所有 `Requests` 显式抛出的异常都继承自 `requests.exceptions.RequestException`