# Tornado知识点

1. 介绍

Tornado 是一个Python web框架和异步网络库 起初由 FriendFeed 开发. 通过使用非阻塞网络I/O。

Tornado 可以支持上万级的连接，处理 长连接, WebSockets, 和其他 需要与每个用户保持长久连接的应用。

特点：1、HTTP服务器；2、异步编程；3、WebSocket

1. Tornado工作流程：

IOLoop开启——》Linux监听socket--若有请求——》路由映射表（application）——》

找到对应Handler处理——》对应的socket

1. 简单框架分析：

#用来定义options选项变量的方法（name,默认值,描述,类型）

define("port", default=8000, help="run on the given port", type=int)

#定义一个RequestHandler类，用于处理请求

class IndexHandler(tornado.web.RequestHandler):

def get(self):#处理get请求

#获取get请求参数'greeting'的值，若这个参数没有出现在查询字符串中，get\_argument的第二个参数作为默认值。

greeting = self.get\_argument('greeting', 'Hello')

#输出信息

self.write(greeting + ', friendly user!')

if \_\_name\_\_ == "\_\_main\_\_":

#可以获取命令行参数

tornado.options.parse\_command\_line()

#处理匹配路径为/的请求

app = tornado.web.Application(handlers=[(r"/", IndexHandler)])

实例化一个HTTP服务器对象

http\_server = tornado.httpserver.HTTPServer(app)

#监听命令行输入的端口号或默认的端口号

http\_server.listen(options.port)

#让IOLoop实例运行

tornado.ioloop.IOLoop.instance().start()

1. RequestHanlder类中常用方法
2. get\_argument、get\_arguments方法：返回给定参数的值,get\_argument获得单个值，get\_arguments获得多个值
3. get\_query\_argument、get\_query\_arguments方法：同1，但仅仅是从URL查询参数中获取值
4. get\_body\_argument、get\_body\_arguments方法：同1，但仅仅是从body中获取值。
5. decode\_argument(value, name=None)方法：从请求中解码一个参数
6. set\_status(status\_code, reason=None)方法：设置响应的状态码，reason (string) – 用人类可读的原因短语来描述状态码
7. set\_header(name, value)：给响应设置指定的头部和对应的值
8. add\_header(name, value)：添加指定的响应头和对应的值
9. clear\_header(name)：清除输出头, 取消之前的 set\_header 调用，不适用于被 add\_header 设置了多个值的头
10. write\_error方法：重写自定义错误页面的实现。

10、write(chunk)：输出信息

11、flush(include\_footers=False, callback=None)：将当前输出缓存区写到网络

12、finish(chunk=None)：完成响应, 结束HTTP 请求

13、render(template\_name, \*\*kwargs)：指定响应模板

14、redirect(url, permanent=False, status=None)：重定向

15、send\_error(status\_code=500, \*\*kwargs)：发送给定的HTTP 错误码

16、clear()：重置这个响应的所有头部和内容

17、get\_cookie、set\_cookie：获取/设置cookie值

get\_secure\_cookie、set\_secure\_cookie：获取与设置安全cookie的值。

以防范浏览器中的恶意修改。为了使用这些函数，必须在应用的构造函数中指定cookie\_secret参数。

18、clear\_cookie(name, path='/', domain=None)：删除给定名称的cookie

19、clear\_all\_cookies(path='/', domain=None)：删除用户在本次请求中所有携带的cookie

20、get\_secure\_cookie\_key\_version(name, value=None)：版本号是int型的

21、create\_signed\_value(name, value, version=None)：产生用时间戳签名的字符串, 防止被伪造

22、

五、