

来关闭此进程，之后再使用`/usr/sbin/bginx` 命令重启即可。

以下是 `HelloWorld0.conf` 的部分文件内容，我是在解读了参数作用后所写的，都是基于本项目所需要的内容，其它部分我暂时还没有深入了解：

```
server {  
  
    # http 的监听端口号，即浏览器输入 ip 地址+:8002 可对项目访问  
  
    listen      8002;  
  
    #服务器的名称，有域名填域名，没域名填 ip 地址  
  
    server_name 123.56.52.53;  
  
    #编码 utf-8 格式  
  
    charset     utf-8;  
  
    # 最大上传的容量，根据实际情况调整，我是在启动后错误提示太小  
    # 了才做的调整  
  
    client_max_body_size 75M;  
  
    # Django static 文件路径  
  
    location /static {  
  
        alias /usr/project/test/HelloWorld0/static; # 指向 django 的 static 目  
        录  
  
    }  
  
    #最后将所有不是 media（我们的项目中没有设置一个 media 目录，所  
    # 以没有配置这一相关参数）内容的请求全部发送给 uwsgi 服务器  
  
    location / {
```

```
uwsgi_pass 127.0.0.1:8001; #通过本地 8001 端口与 uwsgi 进行通信
```

```
include uwsgi_params; # the uwsgi_params file you installed  
}  
}
```

然后是 uwsgi.ini 的内容：

```
[uwsgi]
```

```
#项目路径
```

```
chdir = /usr/project/test/HelloWorld0
```

```
#项目 wsgi 文件名
```

```
module = HelloWorld.wsgi
```

```
#http 的访问地址，若是通过 socket 通信的则不需要此项，将其注释
```

```
#http=0.0.0.0:8001
```

```
# 设置用于 uwsgi 包解析的内部缓存区大小，根据实际情况进行更改
```

```
buffer-size = 65536
```

```
#启动主进程
```

```
master = true
```

```
#为预先派生模式设置工作进程的数量，根据实际情况进行更改
```

```
processes = 10
```

```
#套接字，即双向通信端口号，用于与 nginx 进行通信，必须和 nginx
```

```
配置文件中的 uwsgi_pass 一致
```

socket = 127.0.0.1:8001

以上也仅为本项目目前所需参数，以后若有新的需求还会再进行更改和增加。

配置好后我们通过该配置文件启动 uwsgi 即可：

```
uwsgi -i /usr/project/test/HelloWorld0/conf/uwsgi.ini
```

之后我们关掉终端，仍然可以通过 <http://123.56.52.53:8002> 进入我们的项目中，证明我们部署完成。