**4.1多环境配置·Flask加载配置的三种方式**

前面进行数据库配置的时候，我们可以看到如下代码：

app.config['SQLALCHEMY\_DATABASE\_URI'] = ""  
app.config['SQLALCHEMY\_TRACK\_MODIFICATIONS'] = True  
app.config['SQLALCHEMY\_POOL\_RECYCLE'] = 7200  
app.config['SQLALCHEMY\_RECORD\_QUERIES'] = True  
app.config['SQLALCHEMY\_POOL\_SIZE'] = 5  
app.config['SQLALCHEMY\_POOL\_TIMEOUT'] = 90  
app.config['SQLALCHEMY\_MAX\_OVERFLOW'] = 1024

像上面需要一个个去设置配置参数，并不是很方便，Flask提供了三种更加优雅的设置配置参数的方式，下面我们简单介绍一下。

# 从配置对象中加载(推荐)

新建一个配置文件config.py

class BaseConfig(ｏｂｊｅｃｔ):  
 DEBUG = True  
 APP\_AUTHOR = "mldong"  
 SQLALCHEMY\_TRACK\_MODIFICATIONS = True  
 SQLALCHEMY\_POOL\_RECYCLE = 7200  
 SQLALCHEMY\_RECORD\_QUERIES = True  
 SQLALCHEMY\_POOL\_SIZE = 5  
 SQLALCHEMY\_POOL\_TIMEOUT = 90  
 SQLALCHEMY\_MAX\_OVERFLOW = 1024

导入配置，代码片段

app.config.from\_ｏｂｊｅｃｔ(BaseConfig)

# 通过配置文件加载

新建一个配置文件config.ini

DEBUG = True  
APP\_AUTHOR = mldong  
SQLALCHEMY\_TRACK\_MODIFICATIONS = True  
SQLALCHEMY\_POOL\_RECYCLE = 7200  
SQLALCHEMY\_RECORD\_QUERIES = True  
SQLALCHEMY\_POOL\_SIZE = 5  
SQLALCHEMY\_POOL\_TIMEOUT = 90  
SQLALCHEMY\_MAX\_OVERFLOW = 1024

导入配置，代码片段

app.config.from\_pyfile('config.ini')

# 通过环境变量加载

新建一个配置文件config.py

DEBUG = True  
APP\_AUTHOR = "mldong"  
SQLALCHEMY\_TRACK\_MODIFICATIONS = True  
SQLALCHEMY\_POOL\_RECYCLE = 7200  
SQLALCHEMY\_RECORD\_QUERIES = True  
SQLALCHEMY\_POOL\_SIZE = 5  
SQLALCHEMY\_POOL\_TIMEOUT = 90  
SQLALCHEMY\_MAX\_OVERFLOW = 1024

设置一个环境变量

export CONFIG\_FILE\_PATH=./config.py

导入配置，代码片段

app.config.from\_envvar('CONFIG\_FILE\_PATH')