1. 打印输出flask sqlalchemy慢日志记录，设定SQL执行超时时间的门限值设定为0.6秒。

参考答案为：

from flask\_sqlalchemy import get\_debug\_queries

app = Flask(\_\_name\_\_)

SQLALCHEMY\_COMMIT\_ON\_TEARDOWN =True# 断开设置

FLASKY\_DB\_QUERY\_TIMEOUT=0.6#数据库查询时间的门限值

SQLALCHEMY\_RECORD\_QUERIES=True# 启用慢查询记录功能

@app.after\_request

def after\_request(response):

for query in get\_debug\_queries():

if query.duration >= app.config['FLASKY\_DB\_QUERY\_TIMEOUT']:

print('#####Slow query:%s \nParameters:%s \nDuration:%fs\nContext:%s\n #####' %

(query.statement, query.parameters, query.duration,query.context)) # 打印超时sql执行信息

1. 使用Flask的logging日志记录模块，记录sql执行超时时间的门限为值为0.0005秒，形成日志文件格式输出。

from flask\_sqlalchemy import get\_debug\_queries

from logging.handlers import RotatingFileHandler

import logging

@bp.after\_app\_request

def after\_request(response):

formatter = logging.Formatter( # 设定日志格式

"[%(asctime)s] {%(pathname)s:%(lineno)d} %(levelname)s - %(message)s")

handler = RotatingFileHandler('slow\_query.log', maxBytes=10000, backupCount=10)

handler.setLevel(logging.WARN)

handler.setFormatter(formatter)

logger = logging.getLogger("logger")

logger.addHandler(handler)

for query in get\_debug\_queries():

if query.duration >= FLASKY\_DB\_QUERY\_TIMEOUT:

logger.warn(

('\nContext:{}\nSLOW QUERY: {}\nParameters: {}\nSTART\_TIME: {}\nDuration: {}\n').format(query.context,

query.statement,

query.parameters,

query.start\_time,

query.duration))

return response

config.py文件必须要做如下设置：

FLASKY\_DB\_QUERY\_TIMEOUT=0.0005#数据库查询时间的门限值

SQLALCHEMY\_RECORD\_QUERIES=True