

学号	姓名	提交时间	总分	成绩
-	孙稚朝	2023-06-07 11:54:46	100	91

1. 【简答题】使用flask连接mysql数据库, 实现login.html, 实现登录功能

得分 : 91/100

学生答案 : from flask import Flask,request
from dbtool import DbTool

```

# 创建Flask类对象
app = Flask(__name__)
"""
通过装饰器将url和函数进行绑定, 在访问根url时, 将该请求交给index()函数处理
"""

@app.route('/')
def index():
    # 返回字符串, 表示在页面上显示该字符串
    # return 'Hello, World!'
    return "python从入门到放弃"

"""
在python中, 称之为装饰器; 在java称之为注解 白酒
@app.route 是路由 茅台
默认只支持get
"""

@app.route("/login",methods=['post'])
def login():
    print("执行login() 函数")
    # 获取表单数据的用户名、密码
    # request 封装表单数据
    username = request.form.get("username")
    password = request.form.get("password")
    print("用户名:",username,"密码:",password)

    db = DbTool()
    db.connect("root","123456","127.0.0.1",3306,"std_python")
    sql = "select id,username,password from user where username=%s and password=%s"
    res = db.query(sql,[username,password])
    print("res:",res)
    if len(res) == 1:
        # print("登陆成功!")
        return "登陆成功!"
    else:
        # print("用户名不存在或者密码错误!")
        return "用户名不存在或者密码错误!"
    # return 'hello world'

if __name__ == "__main__":
    # 启动
    app.run()
import mysql.connector
class DbTool:
    def connect(self,user,password,host,port,database):
        # 连接数据库
        self.conn = mysql.connector.connect(user=user,
                                           password=password, # mysql的密码
                                           host=host,
                                           port=port, # 改
                                           database=database, # 改
                                           buffered=True # 缓冲游标模式
                                           )
    def update(self,sql,params):
        # 获取游标, 执行sql
        cursor = self.conn.cursor()
        cursor.execute(sql , params )
        cursor.close()
    def query(self,sql,params):
        # 执行查询
        cursor = self.conn.cursor()
        cursor.execute(sql,params)

```

```
        res = cursor.fetchall()
        cursor.close()
        return res
    def commit(self):
        self.conn.commit()
    def close(self):
        # 断开连接
        self.conn.close()
```

参考答案：略