



UOS LIKELION Python Study

2018. 06. 05.

Made by doorbw



Flask 가지고 놀기_6월 5일

- 1. 지난 수업 복습**
- 2. 메일 보내기**
- 3. Goorm ide 사용하기**
- 4. pymysql 이용하기**



Flask 가지고 놀기_6월 5일

지난 수업 복습

```
1 from flask import Flask  
2 app = Flask(__name__)  
3  
4 @app.route('/')  
5 def hello_world():  
6     return 'Hello World!'  
7  
8 if __name__ == '__main__':  
9     app.run()
```

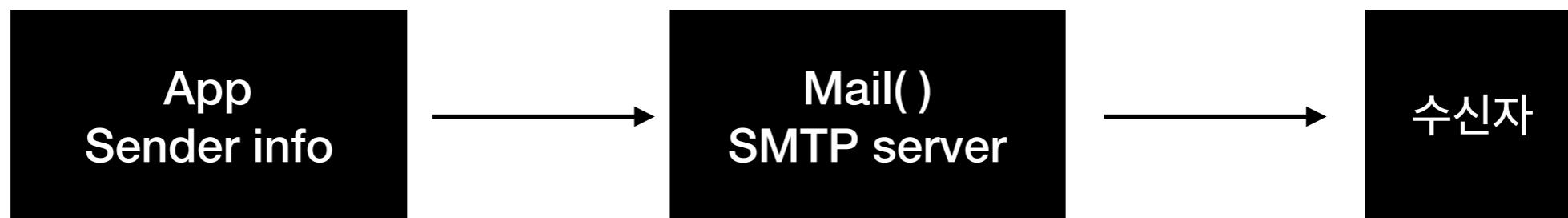
CS



Flask 가지고 놀기_6월 5일

메일 보내기

<https://pythonhosted.org/Flask-Mail/>





Flask 가지고 놀기_6월 5일

```
1  from flask import Flask
2  from flask_mail import Mail, Message
3  # https://pythonhosted.org/Flask-Mail/
4  # mail app을 만드는 2가지 방법
5  # 1. flask app을 먼저 만들고 이를 통해 Mail 객체 생성
6  app = Flask(__name__)
7  app.config.update(
8      DEBUG=True,
9      #EMAIL SETTINGS
10     MAIL_SERVER='smtp.gmail.com',
11     MAIL_PORT=465,
12     MAIL_USE_SSL=True,
13     MAIL_USERNAME = 'doobw@likelion.org',
14     MAIL_PASSWORD = '████████'
15 )
16 mail = Mail(app)
```

```
17
18 # 2. Mail객체를 먼저 만들고, 이후에 만든 flask app을 mail 객체에 설정
19 mail = Mail()
20
21 app = Flask(__name__)
22 mail.init_app(app)
23
```



Flask 가지고 놀기_6월 5일

```
27
28     @app.route('/email')
29     def email_test():
30         msg = Message("Hello",
31                     sender="doorbw@outlook.com",
32                     recipients=["qjadn9@naver.com"])
33         msg.body = "testing"
34         # msg.html = "<b>testing</b>"
35         mail.send(msg)
36     return 'email'
```



Flask 가지고 놀기_6월 5일

```
@app.route("/")
def index():

    msg = Message("Hello",
                  sender="from@example.com",
                  recipients=["to@example.com"])
```

You can set the recipient emails immediately, or individually:

```
msg.recipients = ["you@example.com"]
msg.add_recipient("somebodyelse@example.com")
```

If you have set `MAIL_DEFAULT_SENDER` you don't need to set the message sender explicitly, as it will use this configuration value by default:

```
msg = Message("Hello",
              recipients=["to@example.com"])
```

If the `sender` is a two-element tuple, this will be split into name and address:

```
msg = Message("Hello",
              sender=("Me", "me@example.com"))

assert msg.sender == "Me <me@example.com>"
```

The message can contain a body and/or HTML:

```
msg.body = "testing"
msg.html = "<b>testing</b>"
```

Finally, to send the message, you use the `Mail` instance configured with your Flask application:

```
mail.send(msg)
```



Flask 가지고 놀기_6월 5일





Flask 가지고 놀기_6월 5일

소스	<input checked="" type="radio"/> 팀플린 <input type="radio"/> Github <input type="radio"/> Bitbucket <input type="radio"/> Git/SVN <input type="radio"/> 압축파일																
이름	flask_test																
설명	컨테이너 설명을 입력해주세요.																
소프트웨어 스택 선택	<table border="1"><tr><td colspan="2">Search</td></tr><tr><td></td><td>C/C++</td></tr><tr><td></td><td>Python</td></tr><tr><td></td><td>Django</td></tr><tr><td></td><td>Flask</td></tr><tr><td></td><td>Jupyter Notebook</td></tr><tr><td></td><td>TensorFlow</td></tr></table>	Search			C/C++		Python		Django		Flask		Jupyter Notebook		TensorFlow		
Search																	
	C/C++																
	Python																
	Django																
	Flask																
	Jupyter Notebook																
	TensorFlow																
	<table border="1"><tr><td colspan="2">Flask</td></tr><tr><td>프로젝트 유형</td><td>Flask 프로젝트</td></tr><tr><td>OS</td><td><input checked="" type="checkbox"/> Ubuntu 14.04 LTS</td></tr><tr><td>Python3</td><td><input checked="" type="checkbox"/> 3.6.5</td></tr><tr><td>Python</td><td><input checked="" type="checkbox"/> 2.7.6</td></tr><tr><td>pip3</td><td><input checked="" type="checkbox"/> 9.0.1</td></tr><tr><td>pip</td><td><input checked="" type="checkbox"/> 9.0.1</td></tr><tr><td>Jupyter</td><td><input checked="" type="checkbox"/> 4.3.0</td></tr></table>	Flask		프로젝트 유형	Flask 프로젝트	OS	<input checked="" type="checkbox"/> Ubuntu 14.04 LTS	Python3	<input checked="" type="checkbox"/> 3.6.5	Python	<input checked="" type="checkbox"/> 2.7.6	pip3	<input checked="" type="checkbox"/> 9.0.1	pip	<input checked="" type="checkbox"/> 9.0.1	Jupyter	<input checked="" type="checkbox"/> 4.3.0
Flask																	
프로젝트 유형	Flask 프로젝트																
OS	<input checked="" type="checkbox"/> Ubuntu 14.04 LTS																
Python3	<input checked="" type="checkbox"/> 3.6.5																
Python	<input checked="" type="checkbox"/> 2.7.6																
pip3	<input checked="" type="checkbox"/> 9.0.1																
pip	<input checked="" type="checkbox"/> 9.0.1																
Jupyter	<input checked="" type="checkbox"/> 4.3.0																



Flask 가지고 놀기_6월 5일

The screenshot shows a Goorm IDE interface with the following details:

- IDE Header:** Chrome, 파일, 수정, 보기, 방문 기록, 북마크, 사용자, 도움말.
- Address Bar:** https://ide-run.goorm.io/workspace/flask_test
- Project Explorer:**展示了项目结构，包括`goorm/me/AjG7`、`task_best`和`index.py`。
- Code Editor:** 显示了`index.py`文件的内容：

```
flask insert Flask
import sys
app = Flask(__name__)

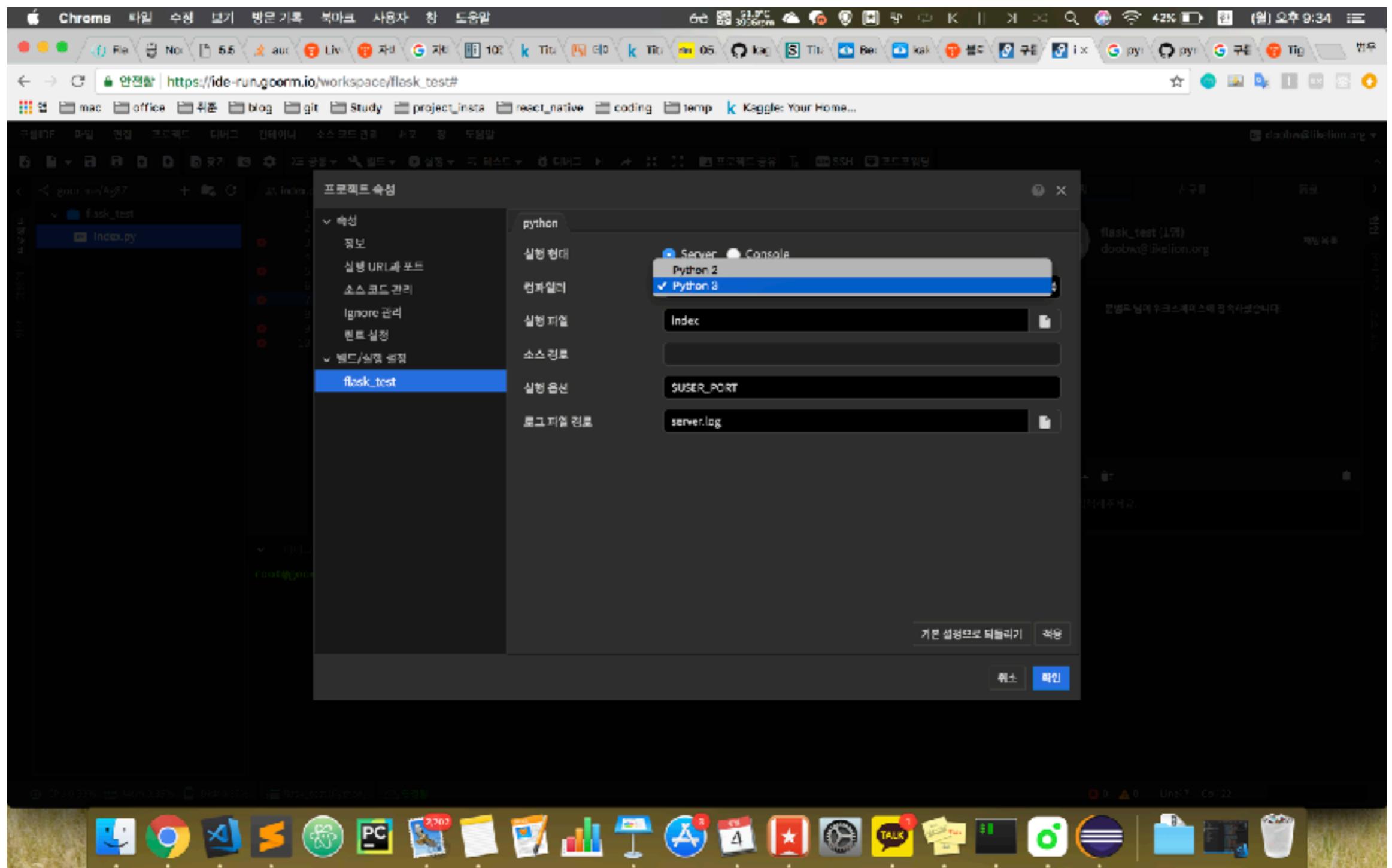
@app.route('/')
def hello():
    return "Hello, goorm!"

if __name__ == "__main__":
    app.run(host='0.0.0.0', port=int(sys.argv[1]))
```

- Terminal:** 显示命令`root@goorm:~/workspace/flask_test\$`。
- Right Panel:** 显示了用户信息（doobw@likation.org）和消息通知。
- Bottom Taskbar:** 包含多个图标，如文件、浏览器、聊天等。



Flask 가지고 놀기_6월 5일





Flask 가지고 놀기_6월 5일

```
from flask import Flask
import sys
app=Flask(__name__)
@app.route("/")
def hello():
    return "Hello World!"
```

```
root@goorm:/workspace/flask_test# pip3 install flask
Collecting flask
```

```
You are using pip version 9.0.3, however version 10.0.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command
root@goorm:/workspace/flask_test# pip3 install pymysql
Collecting pymysql
  Downloading https://files.pythonhosted.org/packages/32/e8/222d9e1c7821f
)
  100% |████████████████████████████████| 1.81kB 0.81kB/s
```



Flask 가지고 놀기_6월 5일

```
계속 하시겠습니까? [Y/n] y
중단.
root@google:~/workspace/flask_test# sudo apt-get install mysql-server
패키지 목록을 읽는 중입니다... 완료
의존성 트리를 만드는 중입니다
상태 정보를 업데이트하는 중입니다
설치할 패키지가 없습니다
```

Mysql 설치

이거때문에 goormIDE 이용...

윈도우에서는 설치시 몇가지 오류가 발생할 수 있다고 함

sudo : 터미널에서 관리자 권한으로 다음 명령어를 실행하겠다.

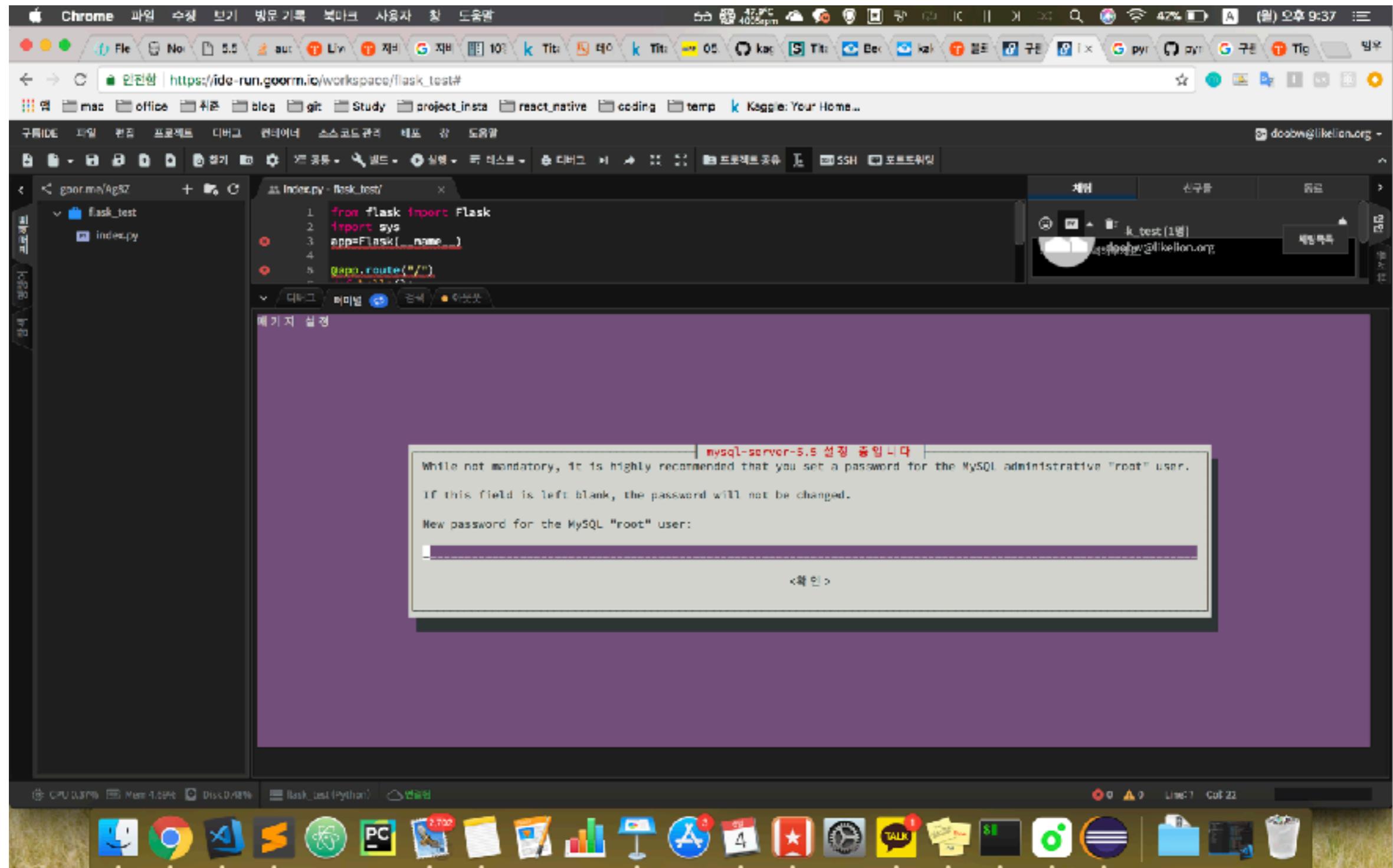
apt-get : ubuntu 계열의 리눅스에서 사용되는 패키지 관리 명령어

install : 설치

mysql-server : mysql서버..



Flask 가지고 놀기_6월 5일





Flask 가지고 놀기_6월 5일

```
Processing triggers for libc-bin: (/lib/x86_64-linux-gnu/libc.so.6 ...)
root@goorm:/workspace/flask_test# service mysql restart
 * Stopping MySQL database server mysqld
   ...done.
 * Starting MySQL database server mysqld
   ...done.
 * Checking for tables which need an upgrade, are corrupt or were
not closed cleanly.
root@goorm:/workspace/flask_test# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 43
Server version: 5.5.60-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```



Flask 가지고 놀기_6월 5일

Create database userDB default character set utf8 collate utf8_general_ci;

Use userDB

Create table user(id int primary key auto_increment, name varchar(32) not null, email varchar(100) not null, verify int not null default 0);

Select * from user;

```
mysql> Create database userDB default character set utf8 collate utf8_general_ci;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> use userDB
Database changed
mysql> Create table user(id int primary key auto_increment, name varchar(32) not null, email varchar(100) not null, verify int not null default 0);
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> select * from userDB;
ERROR 1146 (42S02): Table 'userDB.userDB' doesn't exist
mysql> select * from user;
Empty set (0.00 sec)
```



Flask 가지고 놀기_6월 5일

```
1 # -*- coding: utf-8 -*-
2 from flask import Flask
3 import sys
4 import pymysql
5
6 conn = pymysql.connect(host='localhost', user='root', password='root', db='userDB', charset='utf8', )
7 curs = conn.cursor(pymysql.cursors.DictCursor)
8
9
10 app=Flask(__name__)
11
12 @app.route("/")
13 def hello():
14     return "Hello goorm!"
15
16 @app.route('/mysql_test')
17 def mysql_test():
18     sql = 'insert into user (name,email) values ("문범우","qjadn9@naver.com")'
19     curs.execute(sql)
20     conn.commit()
21     return 'goood'
22
23
24 if __name__ == "__main__":
25     app.run(host='0.0.0.0', port=int(sys.argv[1]))
```