

# 프로젝트명 : tirilee 코딩 테스트

Django 프레임 워크로 개발을 진행 하였으며, 앱은 로그인과 회원가입을 담당하는 account 와 실제 앱을 담당하는 board 앱으로 구성하였습니다.

SQLite3 의 로우 쿼리문을 account 앱에서 사용을 하였고, Django에서 제공하는 쿼리셋은 board 앱에서 사용하였습니다. 두가지 쿼리형식을 사용하여 프로젝트를 진행 해 보았습니다.

## 작업 환경

OS : Windows10, macOS 10.15.5 Catalina

언어 : Python 3.7.7

사용 프레임워크 : Django 3.0.6

DB : SQLite3

에디터 : VSCODE 1.45.1

사용 모듈 :

asgiref	3.2.7
Django	3.0.6
django-appconf	1.0.4
django-imagekit	4.0.2
pilkit	2.0
Pillow	7.1.2
pip	20.1.1
pytz	2020.1
setuptools	41.2.0
six	1.15.0
sqlparse	0.3.1

# DB 설계 및 스토리보드 설계

테이블명	member															
	board															
		member														
		index	email	password	nickname	phone	introduce									
		int	varchar	varchar	text	text	text									
		primary key	not null	not null	not null	not null										
		not null														
		auto increment														

board																
		index	product	price	category	picture	nickname									
		int	text	number	text	text	text									
		primary key	not null	not null	not null		not null									
		not null														
		auto increment														

**중복 확인** (if 문 사용. True일 경우 다시 재입력)

```
SELECT * FROM member WHERE
email='email' OR
nickname='nickname' OR phone='phone';
```

**가입**

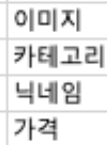
```
INSERT INTO member VALUES(
'email', 'password', 'nickname', 'phone');
```

연락처는 입력시 정규표현식으로 숫자만 입력 받을 것!  
비밀번호 입력시 정규표현식 사용 숫자, 영문, 특수문자 포함 12자

**로그인** (if문 사용. True일 경우 로그인, else 이메일이나 패스워드 불일치)

```
SELECT email, password FROM member
WHERE
email='email' AND
```

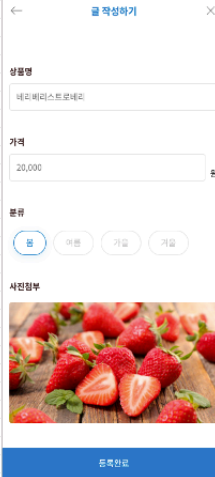
세션을 통하여 로그인 유지 구현



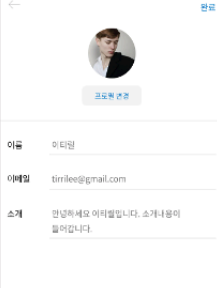
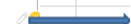


```
SELECT * FROM board WHERE product='검색명';
```

```
SELECT * FROM board WHERE category='선택카테고리'
```



이미지 파일 업로드



```
UPDATE INTO member VALUES(nickname='이름', email='이메일', introduce='소개');
```

자세한 내용은 Repositories 내 'DB및 화면 설계.xlsx' 를 참조 하시면 될 것 같습니다.

# 화면 구성 및 내용

## 전체 구조

- account
  - > \_\_pycache\_\_
  - > migrations
  - > static
  - > templates
  - \_\_init\_\_.py
  - admin.py
  - apps.py
  - forms.py
  - models.py
  - tests.py
  - urls.py
  - views.py
- board
  - > \_\_pycache\_\_
  - > migrations
  - > static
  - > templates / board
  - \_\_init\_\_.py
  - admin.py
  - apps.py
  - forms.py
  - models.py
  - tests.py
  - urls.py
  - views.py
- media / board / images
  - 스크린샷\_2020-05-24\_오후\_8.06.17.jpg
  - 스크린샷\_2020-05-26\_오후\_8.37.49.jpg
- myproject
  - > \_\_pycache\_\_
  - \_\_init\_\_.py
  - asgi.py
  - settings.py
  - urls.py
  - wsgi.py
- venv
- db.sqlite3
- DB및 화면 설계.xlsx
- manage.py
- requirements.tx
- requirements.txt

## 로그인 및 회원가입



☐ 로그인 상태 유지하기

[아이디 찾기](#) | [비밀번호 찾기](#)

회원가입 없이 카카오톡 계정만으로 바로 이용이 가능합니다.  
로그인 시, 이용약관 및 개인정보처리방침 동의로 간주됩니다.

최초 접속 화면 `http://127.0.0.1:8000/` 접속시.

아이디 찾기와 비밀번호 찾기는 구현하지 않았습니다.

## 가입 정보를 입력해주세요 :)

이메일

이메일을 입력해주세요.

비밀번호

닉네임

티릴리에서 사용할 닉네임

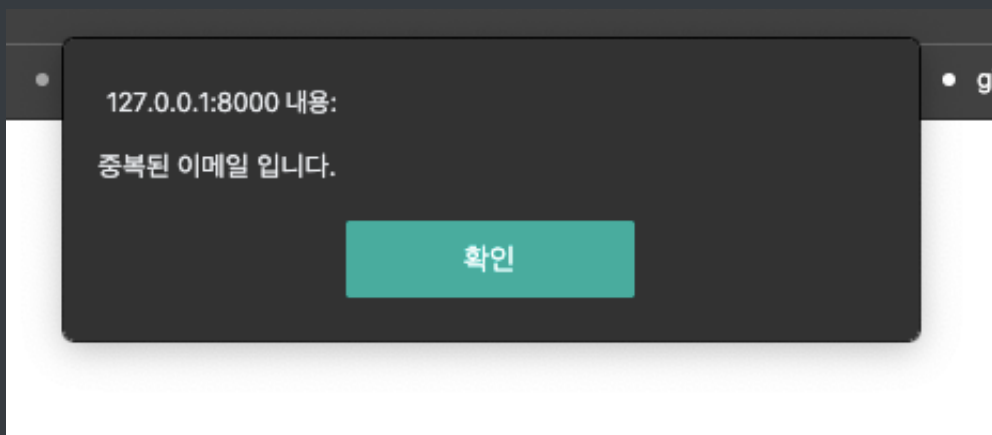
연락처

"-"제외, 숫자만 입력해주시

취소

가입완료

회원 가입 버튼을 눌러 회원 가입 페이지로 이동 시. 마지막 가입완료를 눌렀을 시에 중복회원을 검증하게 됩니다.



중복 감지시 다시 회원가입 페이지로 돌아오며, 중복이 되지 않을 시에 회원 가입이 완료 됩니다.

## 게시판 페이지



# 티릴리 과일농장

[구경하기](#)**복숭아****asd**

가을

30000

★★★★☆

**사과****asd**

여름

50000

★★★★☆

**사과****asd**

겨울

30000

★★★★☆

**사과****asd**

여름

30000

**밀감****asd**

겨울

40000

메인 페이지.

상단은 네비게이션 바로 구성 하였고, 로그인 값이 세션으로 저장되어 상단에서 로그인 한 닉네임을 가져와 출력 해줍니다.

과일명을 입력해주세요.

검색

계절 선택 ▾

700 x 400

복숭아

asd

가을

30000

★★★★☆

700 x 400

사과

asd

여름

50000

★★★★☆

700 x 400

사과

asd

겨울

30000

★★★★☆

700 x 400

사과

asd

여름

30000

★★★★☆

700 x 400

밀감

asd

겨울

40000

★★★★☆

700 x 400

사과

asd

여름

50000

★★★★☆

700 x 400

사과

asd

여름

30000

★★★★☆

## 검색 페이지

사과라는 과일명과, 여름이라는 카테고리를 선택하여 해당 제품만 검색 되도록 구현 하였습니다.

# 이번 주 베스트 과일후기

[보러가기](#)**복숭아**

asd

가을

30000

★★★★☆

**사과**

asd

여름

50000

★★★★☆

**사과**

asd

겨울

30000

★★★★☆

**사과**

asd

여름

30000

**밀감**

asd

겨울

40000

[글 작성하기](#)

커뮤니티 메인 페이지

하단부의 글 작성하기를 클릭하면 등록 페이지로 이동 됩니다.

# 글 작성하기

상품명

상품명을 입력해주세요.

가격

상품 가격을 입력해주세요

분류

☐봄 ☐여름 ☐가을 ☐겨울

사진첨부

선택된 파일 없음

# 글 작성하기

상품명

파인애플

가격

45000

분류

☐ 봄 ☒ 여름 ☐ 가을 ☐ 겨울

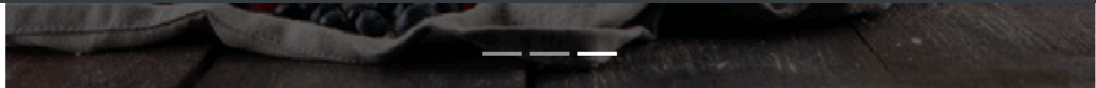
사진첨부

파일 선택

스크린샷 2020-06-01 오전 1.57.16

등록 완료

선택된 파일 없음



복숭아

asd

가을

30000

★★★★☆

사과

asd

여름

50000

★★★★☆

사과

asd

겨울

30000

★★★★☆

사과

asd

여름

30000

★★★★☆

밀감

asd

겨울

40000

★★★★☆

파인애플

광순

여름

45000

★★★★☆

새로 등록한 글과 글쓰기가 로그인 되어있는 사용자명과 함께 등록이 됩니다.



## 마이페이지

이름   광순

이메일   ksyong1234@gmail.com



소개

수정 완료

로그 아웃

### 마이페이지

로그인 되어있는 사용자의 닉네임과 이메일 주소를 가져와 출력하여 보여주고, 소개를 수정 할 수 있게 구현 하였습니다.

로그아웃 클릭 시 세션값이 delete 되고 처음 로그인 화면으로 이동 합니다.

# 프로젝트를 진행 하며 힘들었던 점이나 문제점 및 해결방안

전체적인 화면 디자인이 다소 아쉬웠습니다. 제공 해주신 템플릿에서 아이콘이나 상호 이미지 파일등을 이용하는 것이 한계였습니다.

그리고 세부적인 기능 구현이 다소 아쉬웠습니다. 위와 같이 이미지 파일을 첨부하여 저장할 수 있지만 첨부 된 사진을 불러오는 기능은 구현하지 못하여 다소 아쉽습니다.

그리고 로그인 유지기능과 일반 로그인의 세션 만료 기능을 구현하지 못하였습니다. 장고에서 제공하는 세션 기능의 정보가 부족하였고, 좀더 자세하게 문서를 참조하여 기능을 구현하면 될 것 같습니다.

프로젝트 초반 세션에서 값을 받아오는 중, list형태가 아닌 String 형태로 값이 받아오는 것을 확인하여 다소 어려운 점이 있었습니다. 하지만 json 으로 리스트 형태로 변환하는 방법을 사용하여 이 문제를 해결 하였습니다.

## 발전하기 위한 방향

조금 더 세세한 기능 구현에 고민과 집중하여 이미지 불러오거나 로그인 기능 등 버그를 해결 하는 데 집중하면 좋을 것 같다는 생각을 하였습니다.

## 프로젝트를 하며 느낀점

아무래도 학원의 스케줄을 병행 하다보니 다소 시간이 부족했던 느낌이었습니다. 부족했던 시간때문에 제대로 된 기능 구현이 많지 않아 다소 아쉬웠던 것 같습니다.

하지만 혼자서 어느정도의 규모가 있는 프로젝트를 진행 해 보니 나 자신이 할수 있다 라는 자신감을 가지게 된 것 같습니다. 부족하지만 저의 능력을 많이 성장 시키게 해 준 프로젝트 인 것 같습니다.

본 프로젝트로 인하여 Django 프레임 워크의 이해도도 많이 늘어난 것 같아 결과를 떠나서 정말 하기 잘했다 했던 프로젝트였던 것 같습니다.