# Exercise 9: Creating the jokes App

25 to 40 minutes

The website is going to allow visitors to read other users' jokes and write, edit, and delete their own jokes.

So, you will need to create a `jokes` app.

## ❖ E9.1. Scaffolding

With `djangojokes.com` open in the terminal, run the following command to create the **jokes** app:

```
(.venv) …/projects/djangojokes.com> python manage.py startapp jokes
```

The `djangojokes.com` folder should now look like this:

- `djangojokes.com`
  - `.venv`
  - `djangojokes`
  - **`jokes`**
  - `pages`
  - `templates`
  - `.gitignore`
  - `db.sqlite3`
  - `LICENSE`
  - `manage.py`
  - `README.md`

## ❖ E9.2. Installed Apps

Add the new **jokes** app to `INSTALLED_APPS` in `djangojokes/settings.py`:

```
INSTALLED_APPS = [
    …

    # Local apps
    'jokes.apps.JokesConfig',
    'pages.apps.PagesConfig',
]
```

### Tip: All Else Equal, Alphabetize

In some cases, the order in which code runs makes a difference, but when it doesn't, try to organize similar lines of code in alphabetical order. It makes the code easier to navigate. It's with this in mind that we add `'jokes.apps.JokesConfig'` before `'pages.apps.PagesConfig'`.

## ❖ E9.3. URLs

Next, you need to create a URLConf for the jokes app and let the main URLConf know to hand off some paths to it.

1. Within the `jokes` folder, create a `urls.py` file with the following content:

   ### Exercise Code 9.1: jokes/urls.py

   ```
   from django.urls import path

   app_name = 'jokes'
   urlpatterns = []
   ```

   This URLConf has a namespace of "jokes," which is also the name of the app. The `urlpatterns` list is currently empty, but you will be adding paths to it shortly.

2. Now, you need to let the main URLConf know that it should hand off some paths to the jokes URLConf. Open `djangojokes/urls.py` and modify it as follows:

### Exercise Code 9.2: djangojokes/urls.py

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('jokes/', include('jokes.urls')),
    path('', include('pages.urls')),
]
```

Now, all paths that begin with “jokes/” will get handed off to the jokes URLConf.

# 2.3. Models

Models contains the data fields and behaviors for an app.

## ❖ 2.3.1. The Database

The data for the data fields is stored in a database. By default, that is a SQLite database; however, for production projects, you should use something more robust. You will use SQLite for now, but later you will switch to PostgreSQL, which is the most common database used with Django.

Open `djangojokes/settings.py` and look for the `DATABASES` constant:

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}
```

This is where the database is set. Remember that `BASE_DIR` is set at the top of the settings file. Let’s use the shell to see where exactly that `db.sqlite3` file is saved:

```
(.venv) …/projects/djangojokes.com> python manage.py shell
>>> from djangojokes.settings import *
>>> DATABASES['default']['NAME']
WindowsPath('C:/Webucator/Django/projects/djangojokes.com/db.sqlite3')
```

This shows that `db.sqlite3` is right in the `djangojokes.com` folder:

```
djangojokes.com
    db.sqlite3
```

It was created the first time you ran `python manage.py runserver`. SQLite files are not stored in a readable format, so don't bother opening the file in your editor. Just know that this file is holding the data for the project.

Make sure to exit the shell:

```
>>> exit()
(.venv) …/projects/djangojokes.com>
```

## Git Commit

Commit your code to Git.