Welcome to

Software Development Practice II

Basic web server

Send file to --> Client (web browser)

start from terminal (command line)

python3 -m http.server

Common Gateway Interface (CGI)

Client ---> Server ---> CGI ---> Your Program

Client <--- Server <--- CGI <--- Your Program

Web Framework

• A web framework or web application framework is a software framework that is designed to support the development of web applications https://en.wikipedia.org/wiki/Web_framework

• To try Django https://www.djangoproject.com

Pronounced: JANG-oh (The "D" is silent)

Django Web Framework

Encourages rapid development and clean, pragmatic design

 Built by experienced developers, it takes care of much of the hassle of Web development

so you can focus on writing your app

without needing to reinvent the wheel

Create a new project

```
# commandline/terminal
django-admin startproject hello
```

new project will be in directory hello

Project Files

```
# commandline/terminal
cd hello
ls
hello manage.py
```

```
1s hello # Another hello directory (inside hello) # containing setting files
```

```
__init__.py settings.py urls.py wsgi.py
```

Start a server in a new terminal

```
# Open a new terminal cd hello python3 manage.py runserver ...

Django version 2.0.1, using settings 'hello.settings'

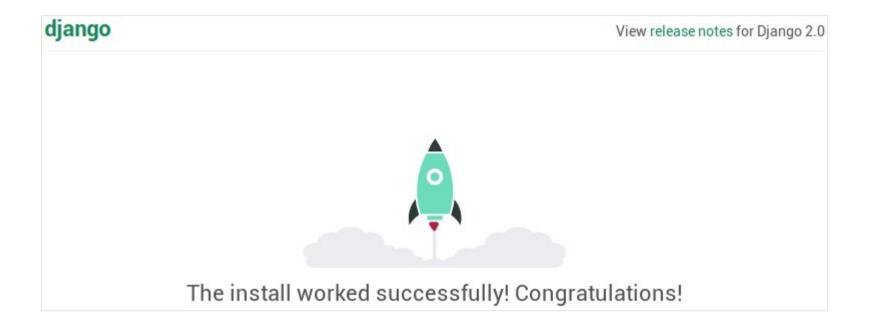
Starting development server at http://127.0.0.1:8000/
```

Ouit the server with CONTROL-C.

Need manage.py to runserver
correct directory?

Open a browser

URL http://127.0.0.1:8000



Create a new app

(many apps in a project?)

python manage.py startapp demo

```
# demo directory containing several files
admin.py apps.py __init__.py migrations
models.py tests.py views.py
```

Create a function

in demo/views.py

```
# Create your views here.

from django.http import HttpResponse

def index(request):
    response = "Hello from index (demo/views.py)"
    return HttpResponse(response)
```

Config URL for the function

(from project)

```
# edit project URL config
# hello/urls.py
from django.urls import include, path
urlpatterns = [
     path('demo/', include('demo.urls')),
# Project hello/urls.py include app demo/urls.py
# Project URL start with 'demo/' will to app URL
# Project URL link to app URL
# try http://127.0.0.1:8000/demo/
```

URL http://127.0.0.1:8000/demo

```
ImportError: No module named 'demo.urls'
# Project hello/urls.py include app demo/urls.py
# But demo/urls.py does not exist
```

Config URL for the function

(from app)

```
# create demo/urls.py
from django.urls import path
from . import views
urlpatterns = [
   path('hi/', views.index, name='index'),
# Project hello/urls.py include app demo/urls.py
# Project URL start with 'demo/' will to app URL
# Project URL start with 'hi/' call the function
```

URL http://127.0.0.1:8000/demo/hi

may need to restart server

Try HTML tag

```
def index(request):
    response = "<b>Hello</b> from index ..."
    return HttpResponse(response)
```

Config URL for more functions

(from app)

```
# create demo/urls.py
from django.urls import path
from . import views
urlpatterns = [
    path('hi/', views.index, name='index'),
    path('time/', views.showtime, name='time'),
# must create showtime()
```

Showtime()

```
# Simple way to debug
#(get a updated response?)
from django.http import HttpResponse
from datetime import datetime
def index(request):
    r = "<b>Hello</b> from index (demo/views.py)"
    return HttpResponse(r)
def showtime(r):
   t = datetime.now()
    r ="current time: " + t.strftime("%A, %d. %B %Y %I:%M:%S")
    return HttpResponse(r)
```

Change Timezone

```
# edit hello/settings.py
# change TIME_ZONE as below
TIME_ZONE = 'Asia/Bangkok'
```



Exercise

- Create more functions in views
 - to make sure you understand the basic

• Follow Django tutorials

https://docs.djangoproject.com/en/2.0/intro/tutorial01/

• Prepare slides to present next time

• Creating a project

• Creating the Polls app (Projects vs. apps)

• Write your first view

• Database setup

(not needed, use sqlite)

• Creating models

• Playing with the API

• Introducing the Django Admin

• Writing more views

• Use the template system

• Namespacing URL names

• Write a simple form

• Use generic views