

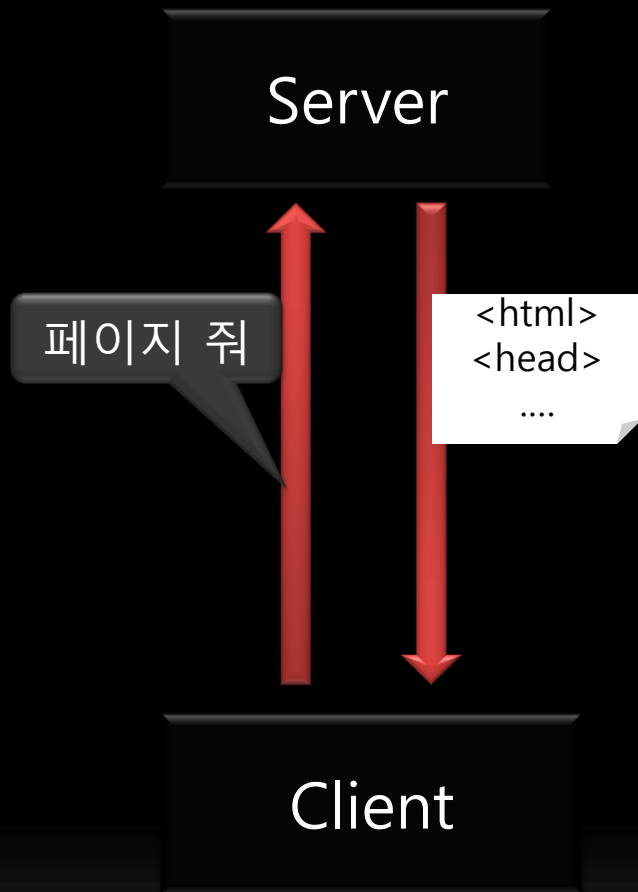
`09 Summer SPARCS Seminar

Introduction to Django

SPARCS `08

서우석(pipoket)

▶ What is Web?



Request Page

Give Page

...on what Protocol?

▶ HTTP Hyper Text Transfer Protocol

Protocol for HyperText

HTTP/1.0 HTTP/1.1

▶ HTTP Hyper Text Transfer Protocol

Header

- HTTP/1.0 200 OK
- Content-Type: text/html

HTML File

- `<html><head><title>...`

▶ Serving Webpage

요청을 기다린다

요청을 받는다

해당 페이지를 돌려준다

Following HTTP

▶ Serving Webpage



Apache는...

HTTP 요청을 받고

해당하는 HTML문서 포착

HTTP 프로토콜에 맞게 포장

요청한 클라이언트에 전달

▶ Serving Webpage



내용이 계속 변한다

코드를 계속 수정?

1분 단위로 업데이트?

1초 단위로 업데이트?

HOW?

▶ CGI Common Gateway Interface

```
1 #!/usr/local/bin/python
2 import cgi; cgi.enable()
3
4 print "Content-Type: text/html#\n"
5
6 html_header = '''
7 <HTML>
8 <HEAD></HEAD>
9 <BODY> '''
10
11 html_footer = '''
12 </BODY>
13 </HTML> '''
14
15 print html_header
16
17 print "<H2>%s</H2>" % 'Title'
18 print "contents : <br>"
19 print "<b>%s</b><br>" % "bold-typed text"
20
21 for k in range(10):
22     print k, '<br>'
23
24 print html_footer
```

HTML

한번 만들면
바꾸기 힘들다

그렇다면

그때그때

HTML을 생성하자!

▶ CGI Common Gateway Interface

단점

1. 지저분하다 *<html> <head>...*
2. 신경쓸 것이 많다 *Content-Type...*
3. 디버깅이 힘들다 *500 Internal Server Error*
4. 보안이 취약할지도? *고의로 Exception을...*
5. 추가적인 기능 구현이 힘들다
 - Database와의 연동?
 - 다른 모듈들과의 연동?

▶ Django

HTML 한번 만들면 바꾸기 힘들다

그럼, 분리하자!

HTML과 자료 처리와 자료 저장

► Django

Django Framework

Model
(DB)

Controller
(Code)

View
(html)

▶ How to Install

- On Debian-Package System

```
# apt-get install python-django
```

```
# easy_install django
```

- Or you can compile and install it!

▶ Django Workflow

"Project"

"Applications"

Make own "View", "Model"

Connect "View" with "urls.py"

Check with "Dev. Server"

► Start Django Project

```
$ django-admin.py startproject "ProjectName"
```

```
[02:45:13] pipoket@pipoket:~/project$ django-admin.py startproject spseminar  
[02:45:19] pipoket@pipoket:~/project$ cd spseminar/  
[02:45:23] pipoket@pipoket:~/project/spseminar$ ls  
__init__.py  manage.py  settings.py  urls.py  
[02:45:24] pipoket@pipoket:~/project/spseminar$ █
```

► Start Django Application

```
$ django-admin.py startapp "ApplicationName"
```

```
[02:46:22] pipoket@pipoket:~/project/spseminar$ django-admin.py startapp helloworld  
[02:46:29] pipoket@pipoket:~/project/spseminar$ ls  
helloworld  __init__.py  manage.py  settings.py  urls.py  
[02:46:31] pipoket@pipoket:~/project/spseminar$ cd helloworld/  
[02:46:33] pipoket@pipoket:~/project/spseminar/helloworld$ ls  
__init__.py  models.py  views.py  
[02:46:33] pipoket@pipoket:~/project/spseminar/helloworld$ █
```

► Make View

```
[02:47:25] pipoket@pipoket:~/project/spseminar/helloworld$ vim views.py
```

```
1 # Create your views here.  
2 import datetime  
3  
4 from django.http import HttpResponse  
5  
6 def helloworld(request):  
7     date = datetime.datetime.now()  
8     string = "Hello World!\n%s" % date  
9     return HttpResponse(string)
```

▶ Connect URL with View

```
[02:48:57] pipoket@pipoket:~/project/spseminar$ vim urls.py
```

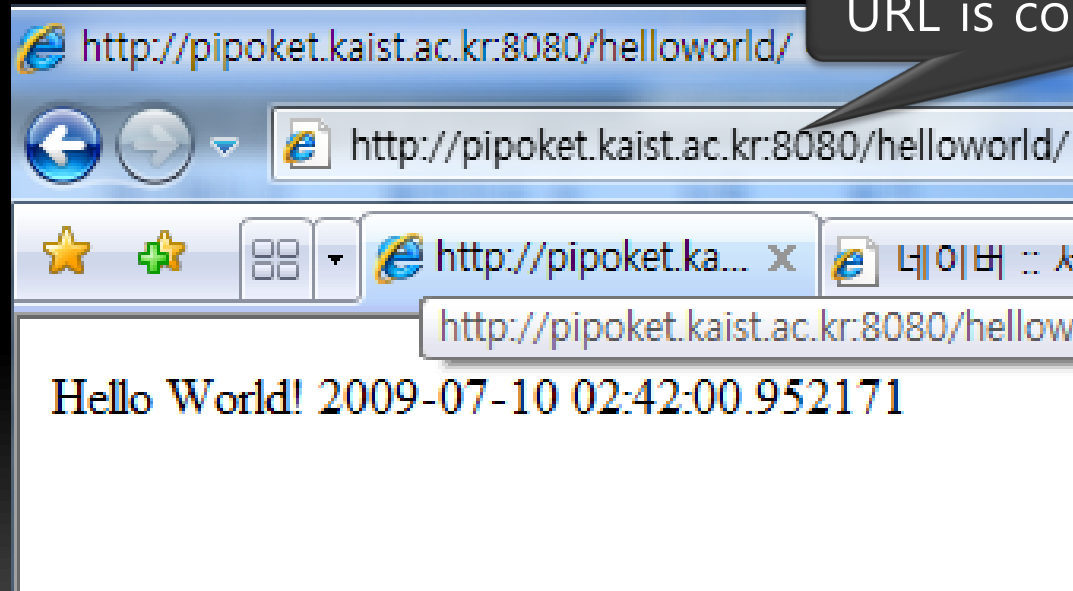
Import view

```
1 from django.conf.urls.defaults import *
2 from spseminar.helloworld.views import helloworld
3
4 # Uncomment the next two lines to enable the admin:
5 # from django.contrib import admin
6 # admin.autodiscover()
7
8 urlpatterns = patterns('',
9     # Example:
10    # (r'^spseminar/', include('spseminar.foo.urls')),
11    (r'^helloworld/', helloworld),
12
13    # Uncomment the admin/doc line below and add 'django.contrib.admindocs'
14    # to INSTALLED_APPS to enable admin documentation:
15    # (r'^admin/doc/', include('django.contrib.admindocs.urls')),
16
17    # Uncomment the next line to enable the admin:
18    # (r'^admin/(.*)', admin.site.root),
```

Connect view with URL

▶ Start Dev. Server

```
$ cd $ROOT_PATH_TO_PROJET  
$ python manage.py runserver 0.0.0.0:8080
```



► Conclusion

Simple Concept of HTTP

Why Django is necessary

How to Install Django

Basic Dev. Flow in Django

▶ Homework

- Download the file

http://pipoket.kaist.ac.kr/sp_seminar/django_hw1.tar.gz

- Extract the downloaded file to any path you want
(Refer the Python101, Python102 pdf about extraction)
- You have to make a simple web application does...
 - Get the input from the form
 - Save the input to the file
 - Show the entry of the file

▶ Homework

- Form and getting input is already implemented
- What you have to do is...
 - Implement the "TODO" part on
 - "input" and "show" functions
 - They are in "spseminar/studentinfo/views.py"
 - Read the comment carefully (hey, it`s in Korean 😊)
 - Implement the function as you wish
- You should need Dev. Server.

USE THE PORT RANGE BETWEEN 20000-30000

If some port is already taken, use the other port

▶ Useful Information

- Vim에 색깔도 없고..
Indentation도 자동으로 안돼요 ㅏㅏㅏ..

```
$ cp /home/pipoket/.vimrc ~/
```

```
$ cp -r /home/pipoket/.vim ~/
```

- .vimrc 파일은 vim이 실행될 때 마다,
색깔, indent등에 관련된 설정을 적용해 줌!
- 만들기 귀찮으니깐 그냥 가져다 쓰세요~