CS Directory

Daniel Koohmarey & Jason Carter

Goals

- Our Compsci Directory Database consolidates useful information on the web to allow students to explore courses like never before.

- Enables useful course related queries

Database Size

Table	Records
Courses	362
Assistants	278
Tutors	59
Professors	47
Buildings	17

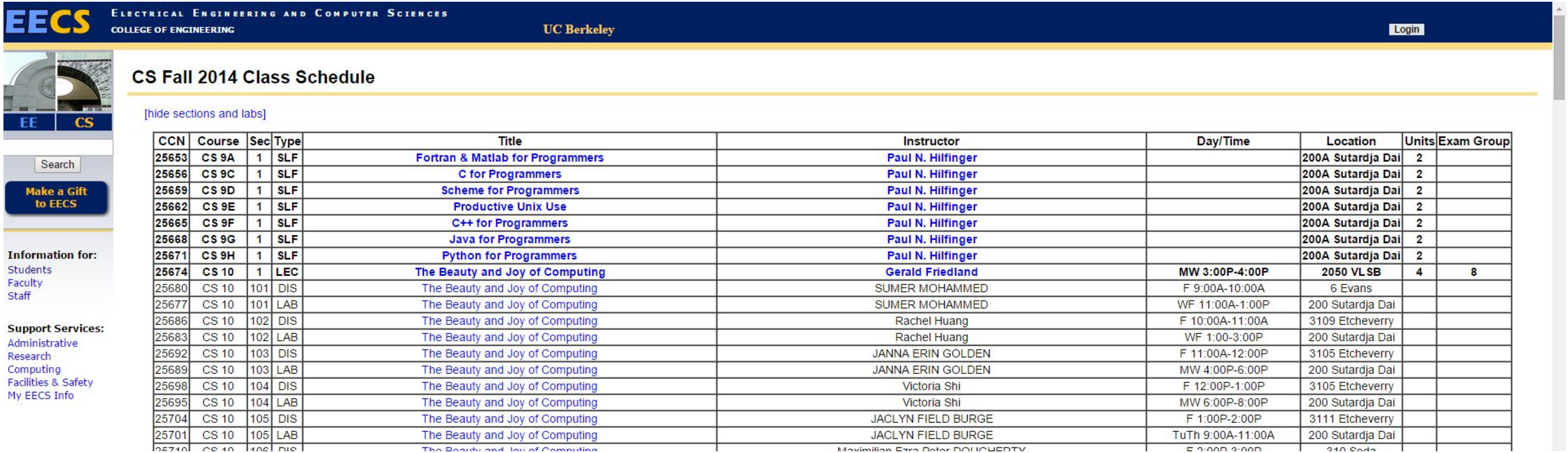
Tables

1. **Tutors** – person_name, course_codes, day, time, building_name, and room_no.
2. **Professors** – professor_name, CCN, rating, salary, position, and homepage.
3. **Courses** – CCN, course_code, course_title, section_no, description, time, room_no, building_name, course_type, units, and exam.
4. **Assistants** – person_name, CCN, and officer.
5. **Buildings** – building_name, history, built, and namesake.

Automate Everything

Online data for:

Courses, Professors, Buildings, Assistants



CS Fall 2014 Class Schedule

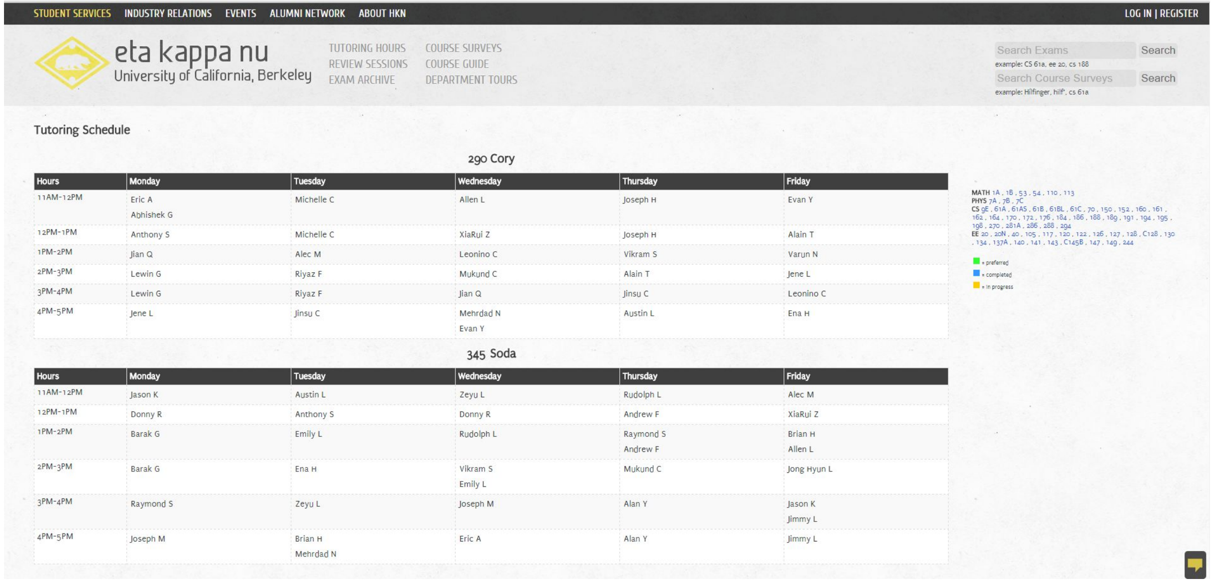
[hide sections and labs]

CCN	Course	Sec	Type	Title	Instructor	Day/Time	Location	Units	Exam Group
25653	CS 9A	1	SLF	Fortran & Matlab for Programmers	Paul N. Hilfinger		200A Sutardja Dai	2	
25656	CS 9C	1	SLF	C for Programmers	Paul N. Hilfinger		200A Sutardja Dai	2	
25659	CS 9D	1	SLF	Scheme for Programmers	Paul N. Hilfinger		200A Sutardja Dai	2	
25662	CS 9E	1	SLF	Productive Unix Use	Paul N. Hilfinger		200A Sutardja Dai	2	
25665	CS 9F	1	SLF	C++ for Programmers	Paul N. Hilfinger		200A Sutardja Dai	2	
25668	CS 9G	1	SLF	Java for Programmers	Paul N. Hilfinger		200A Sutardja Dai	2	
25671	CS 9H	1	SLF	Python for Programmers	Paul N. Hilfinger		200A Sutardja Dai	2	
25674	CS 10	1	LEC	The Beauty and Joy of Computing	Gerald Friedland	MW 3:00P-4:00P	2050 VLSB	4	8
25680	CS 10	101	DIS	The Beauty and Joy of Computing	SUMER MOHAMMED	F 9:00A-10:00A	6 Evans		
25677	CS 10	101	LAB	The Beauty and Joy of Computing	SUMER MOHAMMED	WF 11:00A-1:00P	200 Sutardja Dai		
25686	CS 10	102	DIS	The Beauty and Joy of Computing	Rachel Huang	F 10:00A-11:00A	3109 Etcheverry		
25683	CS 10	102	LAB	The Beauty and Joy of Computing	Rachel Huang	WF 1:00-3:00P	200 Sutardja Dai		
25692	CS 10	103	DIS	The Beauty and Joy of Computing	JANNA ERIN GOLDEN	F 11:00A-12:00P	3105 Etcheverry		
25689	CS 10	103	LAB	The Beauty and Joy of Computing	JANNA ERIN GOLDEN	MW 4:00P-6:00P	200 Sutardja Dai		
25698	CS 10	104	DIS	The Beauty and Joy of Computing	Victoria Shi	F 12:00P-1:00P	3105 Etcheverry		
25695	CS 10	104	LAB	The Beauty and Joy of Computing	Victoria Shi	MW 6:00P-8:00P	200 Sutardja Dai		
25704	CS 10	105	DIS	The Beauty and Joy of Computing	JACLYN FIELD BURGE	F 1:00P-2:00P	3111 Etcheverry		
25701	CS 10	105	LAB	The Beauty and Joy of Computing	JACLYN FIELD BURGE	TuTh 9:00A-11:00A	200 Sutardja Dai		
25710	CS 10	106	DIS	The Beauty and Joy of Computing	Maximilian Ezra Peter DOUGHERTY	F 2:00P-3:00P	310 Soda		

Automate Everything

Online data for:

- Tutors



Tutoring Schedule

290 Cory

Hours	Monday	Tuesday	Wednesday	Thursday	Friday
11AM-12PM	Eric A Abhishek G	Michelle C	Allen L	Joseph H	Evan Y
12PM-1PM	Anthony S	Michelle C	XiaRui Z	Joseph H	Alain T
1PM-2PM	Jian Q	Alec M	Leonino C	Vikram S	Varun N
2PM-3PM	Lewin G	Riyaz F	Mukund C	Alain T	Jene L
3PM-4PM	Lewin G	Riyaz F	Jian Q	Jinsu C	Leonino C
4PM-5PM	Jene L	Jinsu C	Mehrdad N Evan Y	Austin L	Ena H

345 Soda

Hours	Monday	Tuesday	Wednesday	Thursday	Friday
11AM-12PM	Jason K	Austin L	Zeyu L	Rudolph L	Alec M
12PM-1PM	Donny R	Anthony S	Donny R	Andrew F	XiaRui Z
1PM-2PM	Barak G	Emily L	Rudolph L	Raymond S Andrew F	Brian H Allen L
2PM-3PM	Barak G	Ena H	Vikram S Emily L	Mukund C	Jong Hyun L
3PM-4PM	Raymond S	Zeyu L	Joseph M	Alan Y	Jason K Jimmy L
4PM-5PM	Joseph M	Brian H Mehrdad N	Eric A	Alan Y	Jimmy L

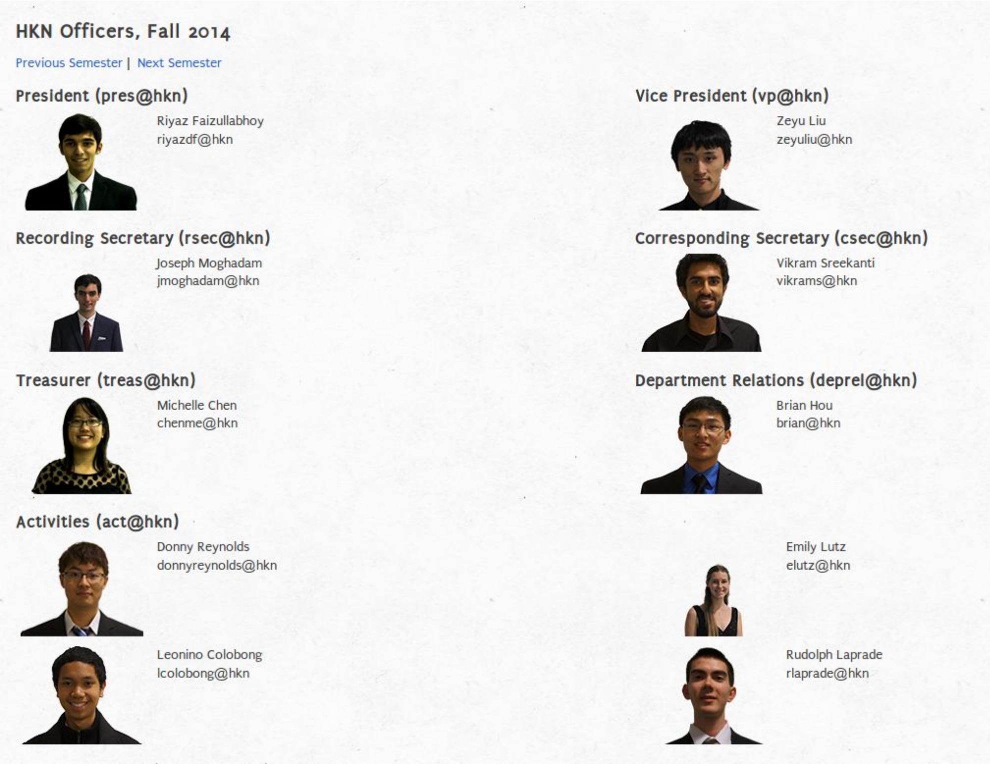
MATH 1A , 1B , 53 , 54 , 110 , 113
PHYS 7A , 7B , 7C
CS 9E , 61A , 61AS , 61B , 61BL , 61C , 70 , 150 , 152 , 160 , 161 , 162 , 164 , 170 , 172 , 176 , 184 , 186 , 188 , 189 , 191 , 194 , 195 , 198 , 270 , 281A , 286 , 288 , 294
EE 20 , 20N , 40 , 105 , 117 , 120 , 122 , 126 , 127 , 128 , C128 , 130 , 134 , 137A , 140 , 141 , 143 , C145B , 147 , 149 , 244

🟩 = preferred
🟦 = completed
🟨 = in progress

Automate and manual corrections

Online data for:

HKN officers



HKN Officers, Fall 2014

Previous Semester | Next Semester

President (pres@hkn)

Riyaz Faizullabhoy
riyazdf@hkn

Vice President (vp@hkn)

Zeyu Liu
zeyuliu@hkn

Recording Secretary (rsec@hkn)

Joseph Moghadam
jmoghadam@hkn

Corresponding Secretary (csec@hkn)

Vikram Sreekanti
vikrams@hkn

Treasurer (treas@hkn)

Michelle Chen
chenme@hkn

Department Relations (deprel@hkn)

Brian Hou
brian@hkn

Activities (act@hkn)

Donny Reynolds
donnyreynolds@hkn

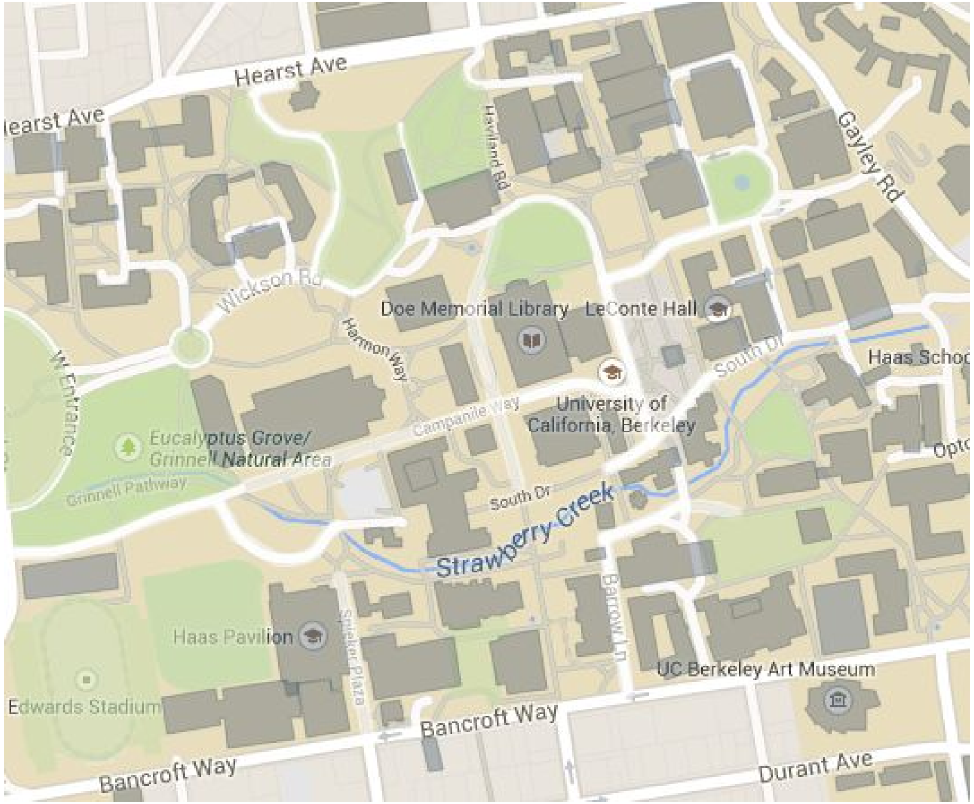
Emily Lutz
elutz@hkn

Leonino Colobong
lcolobong@hkn

Rudolph Laprade
rlaprade@hkn

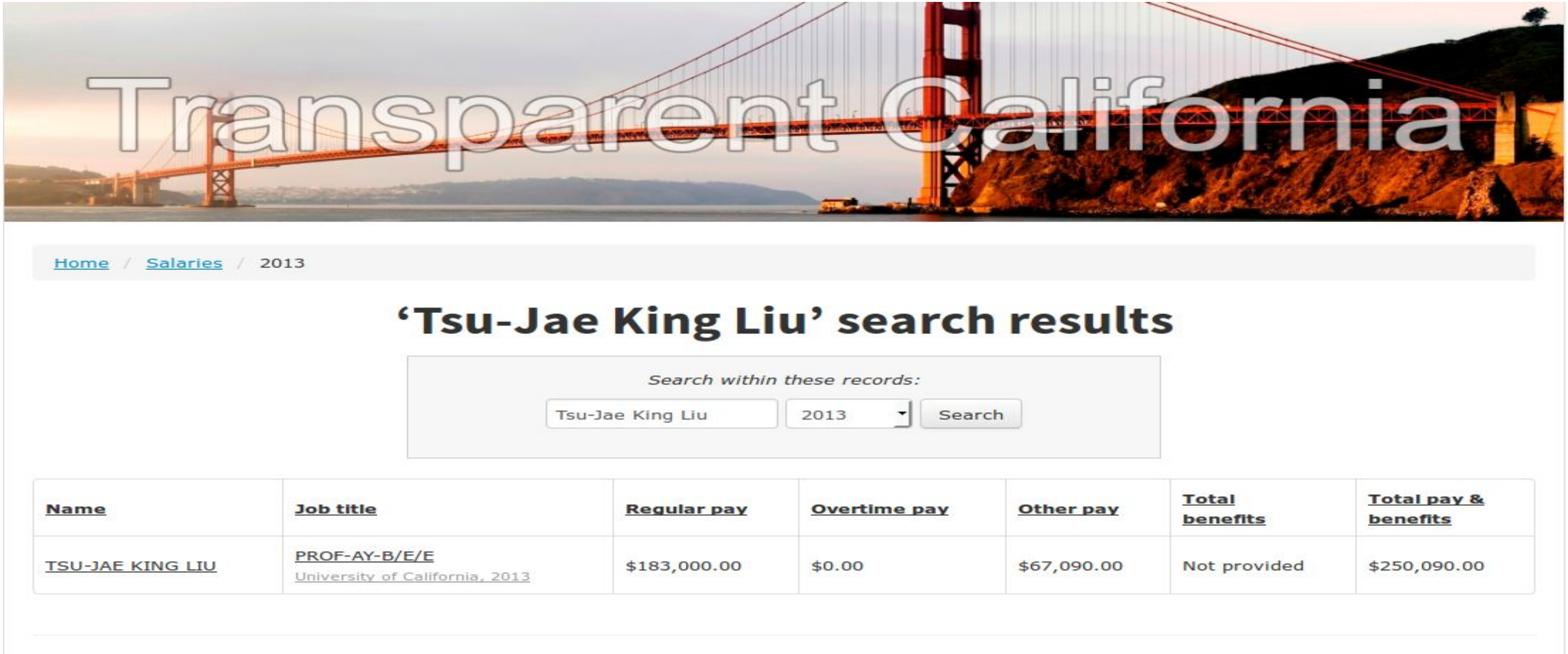
Manual data retrieval

Online data for:
Berkeley
buildings
(Google Maps)



Manual data retrieval

Online data for: Professors



Home / Salaries / 2013

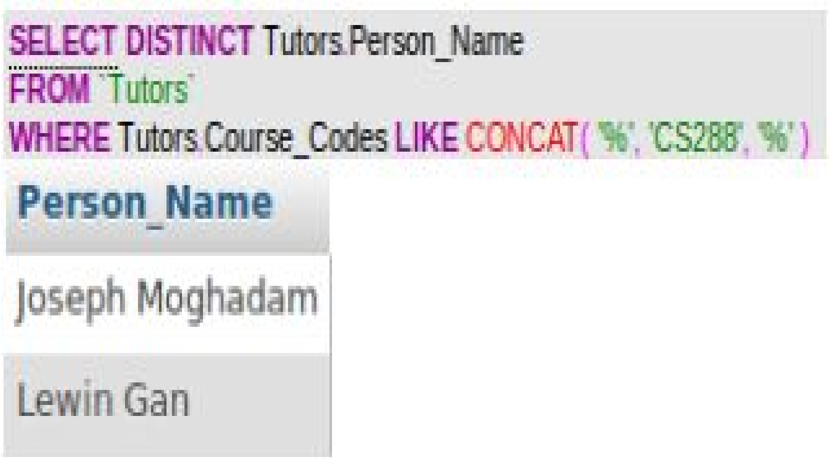
‘Tsu-Jae King Liu’ search results

Search within these records:

Name	Job title	Regular pay	Overtime pay	Other pay	Total benefits	Total pay & benefits
TSU-JAE KING LIU	PROF-AY-B/E/E University of California, 2013	$183,000.00	$0.00	$67,090.00	Not provided	$250,090.00

Use Cases:

Who can tutor CS288?

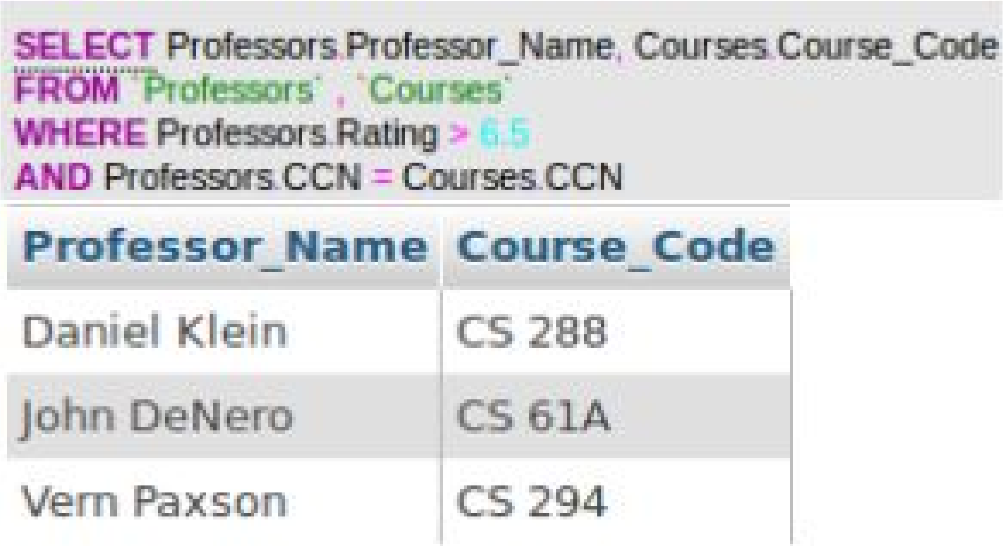


```
SELECT DISTINCT Tutors.Person_Name
FROM `Tutors`
WHERE Tutors.Course_Codes LIKE CONCAT( '%', 'CS288', '%' )
```

Person_Name
Joseph Moghadam
Lewin Gan

Use Cases:

Which courses are being taught by professors with at least a 6.5 rating?

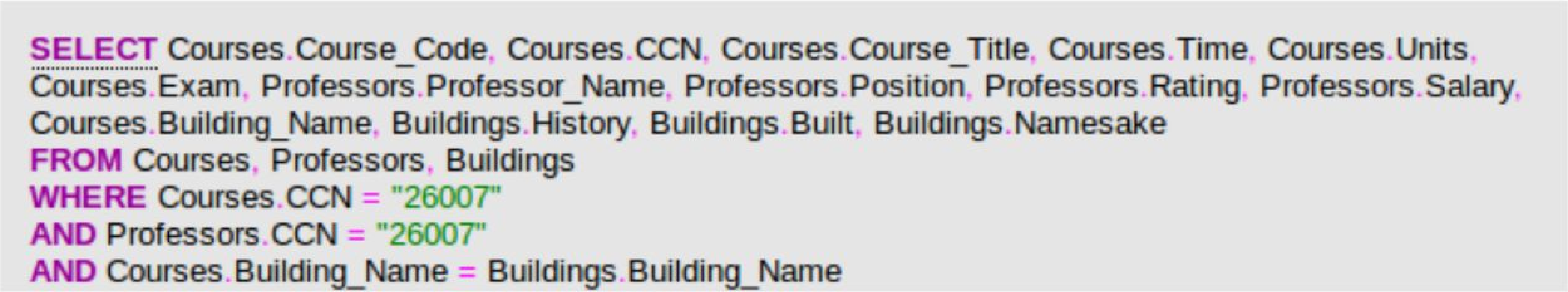


```
SELECT Professors.Professor_Name, Courses.Course_Code
FROM `Professors` , `Courses`
WHERE Professors.Rating > 6.5
AND Professors.CCN = Courses.CCN
```

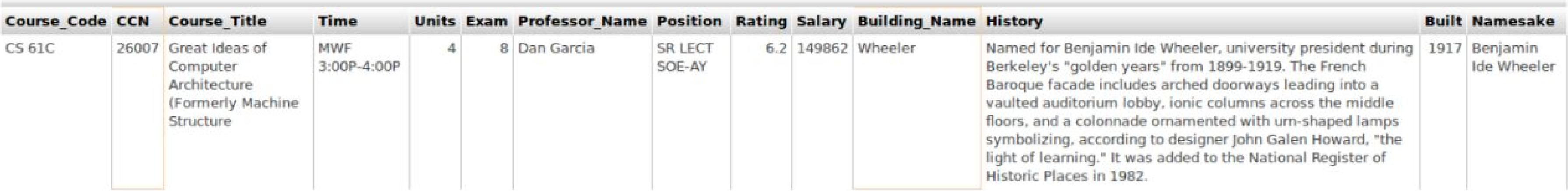
Professor_Name	Course_Code
Daniel Klein	CS 288
John DeNero	CS 61A
Vern Paxson	CS 294

Use Cases:

Can I get more detailed information on CS61C?



```
SELECT Courses.Course_Code, Courses.CCN, Courses.Course_Title, Courses.Time, Courses.Units,
Courses.Exam, Professors.Professor_Name, Professors.Position, Professors.Rating, Professors.Salary,
Courses.Building_Name, Buildings.History, Buildings.Built, Buildings.Namesake
FROM Courses, Professors, Buildings
WHERE Courses.CCN = "26007"
AND Professors.CCN = "26007"
AND Courses.Building_Name = Buildings.Building_Name
```



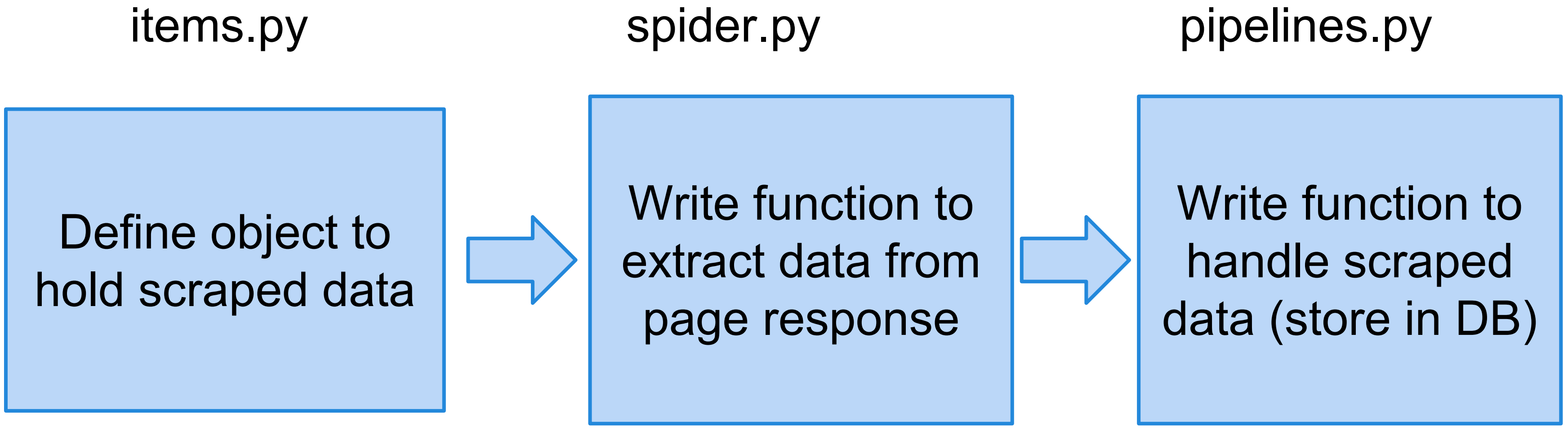
Course_Code	CCN	Course_Title	Time	Units	Exam	Professor_Name	Position	Rating	Salary	Building_Name	History	Built	Namesake
CS 61C	26007	Great Ideas of Computer Architecture (Formerly Machine Structure	MWF 3:00P-4:00P	4	8	Dan Garcia	SR LECT SOE-AY	6.2	149862	Wheeler	Named for Benjamin Ide Wheeler, university president during Berkeley's "golden years" from 1899-1919. The French Baroque facade includes arched doorways leading into a vaulted auditorium lobby, ionic columns across the middle floors, and a colonnade ornamented with urn-shaped lamps symbolizing, according to designer John Galen Howard, "the light of learning." It was added to the National Register of Historic Places in 1982.	1917	Benjamin Ide Wheeler

Scrape Data

Python scraping/crawling framework:

Scrapy Overview

Key Framework files:



items.py

Define object to hold scraped data

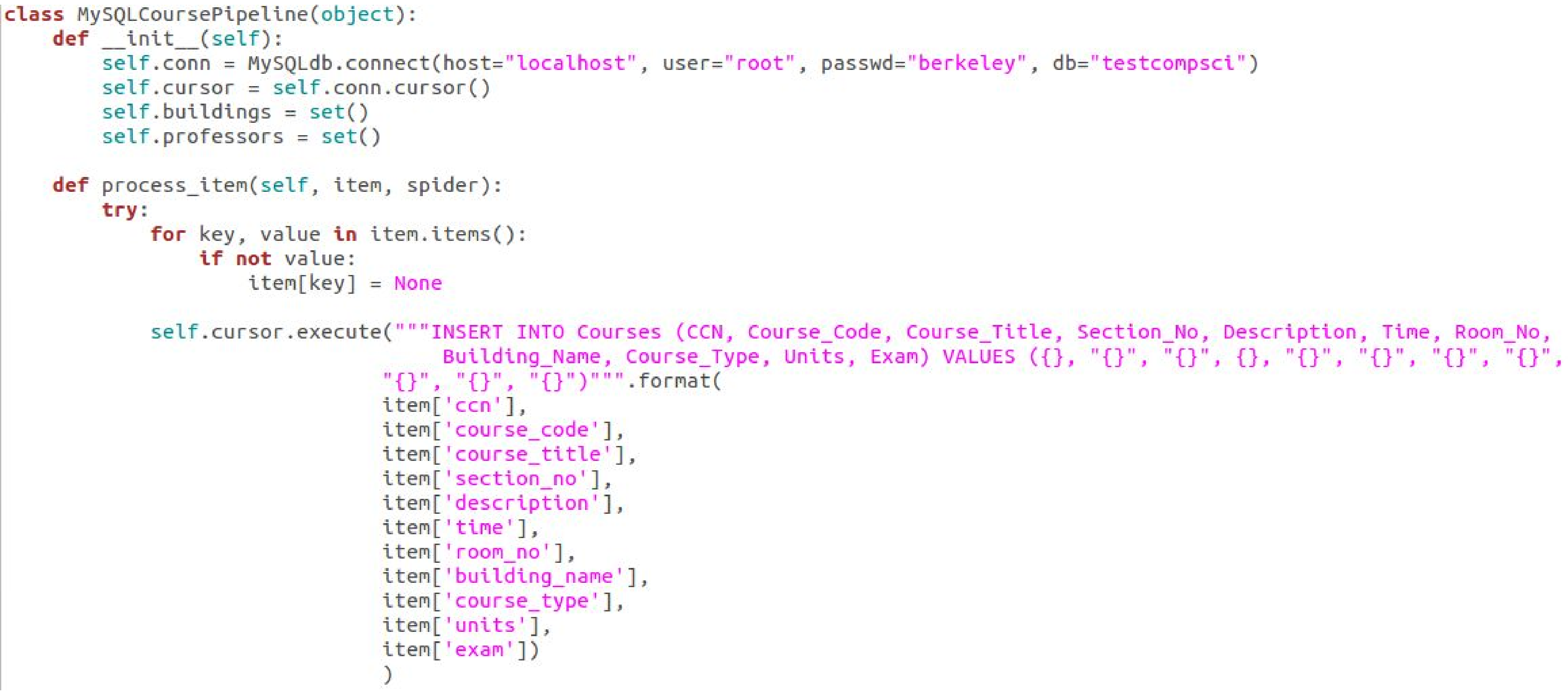
spider.py

Write function to extract data from page response

pipelines.py

Write function to handle scraped data (store in DB)

Save to the database



```
class MySQLCoursePipeline(object):
    def __init__(self):
        self.conn = MySQLdb.connect(host="localhost", user="root", passwd="berkeley", db="testcompsci")
        self.cursor = self.conn.cursor()
        self.buildings = set()
        self.professors = set()

    def process_item(self, item, spider):
        try:
            for key, value in item.items():
                if not value:
                    item[key] = None

            self.cursor.execute("""INSERT INTO Courses (CCN, Course_Code, Course_Title, Section_No, Description, Time, Room_No,
                                    Building_Name, Course_Type, Units, Exam) VALUES ({}, "{}", "{}", {}, "{}", "{}", "{}", "{}",
                               "{}", "{}", "{}")""".format(
                               item['ccn'],
                               item['course_code'],
                               item['course_title'],
                               item['section_no'],
                               item['description'],
                               item['time'],
                               item['room_no'],
                               item['building_name'],
                               item['course_type'],
                               item['units'],
                               item['exam'])
                               )
```

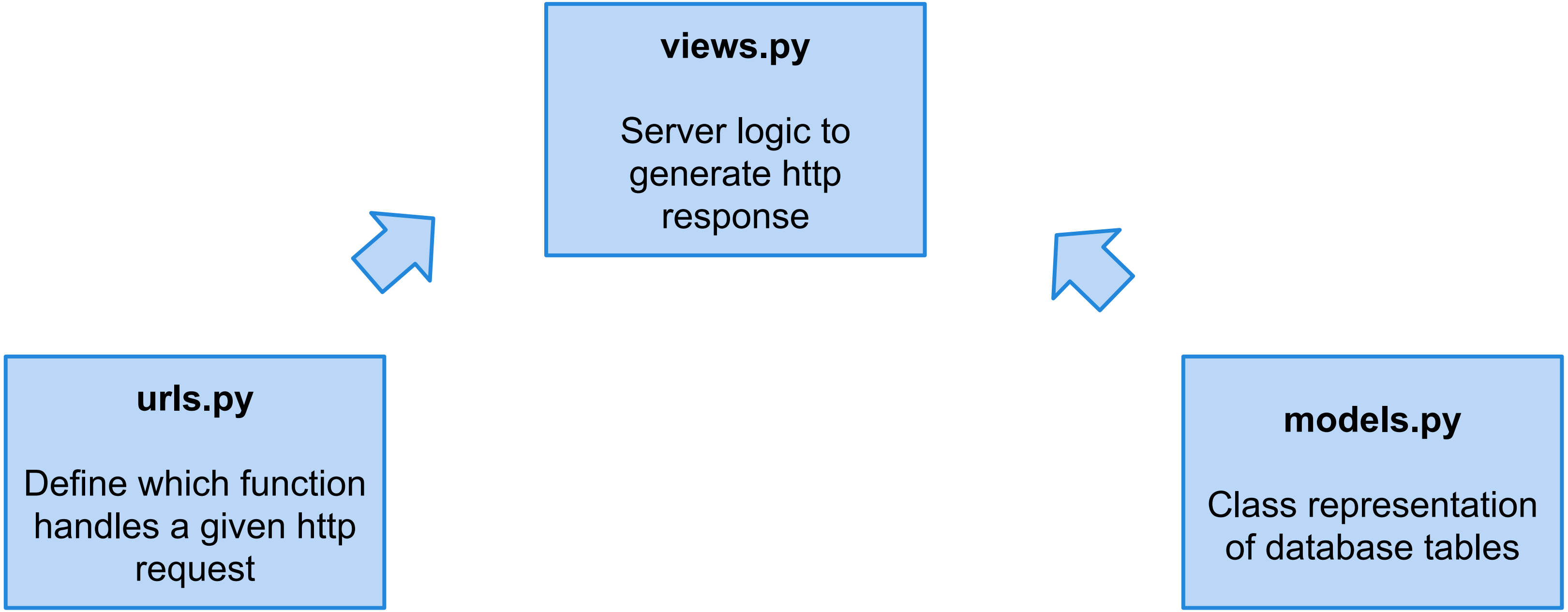
The User Interface

django
python

jQuery
write less, do more.

Django Overview

Model-View Controller Architecture



views.py

Server logic to generate http response

urls.py

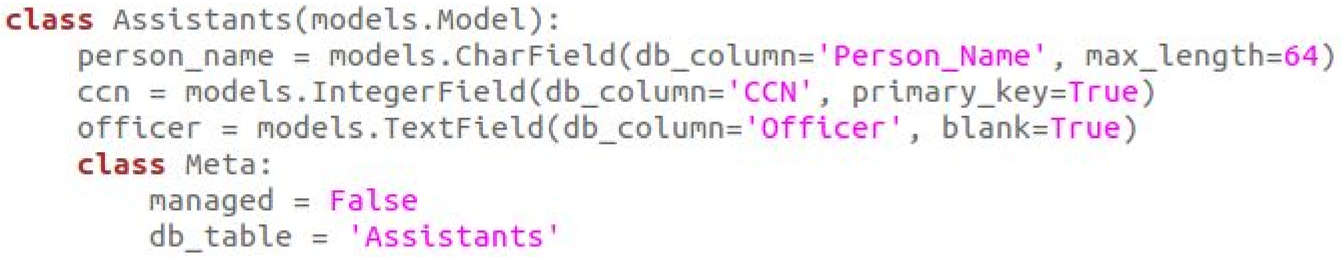
Define which function handles a given http request

models.py

Class representation of database tables

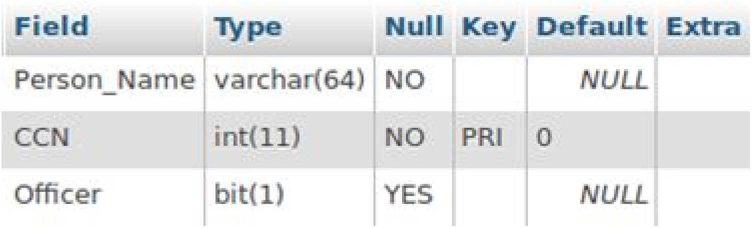
Django Models: DB tables in python

python manage.py inspect db



```
class Assistants(models.Model):
    person_name = models.CharField(db_column='Person_Name', max_length=64)
    ccn = models.IntegerField(db_column='CCN', primary_key=True)
    officer = models.TextField(db_column='Officer', blank=True)
    class Meta:
        managed = False
        db_table = 'Assistants'
```

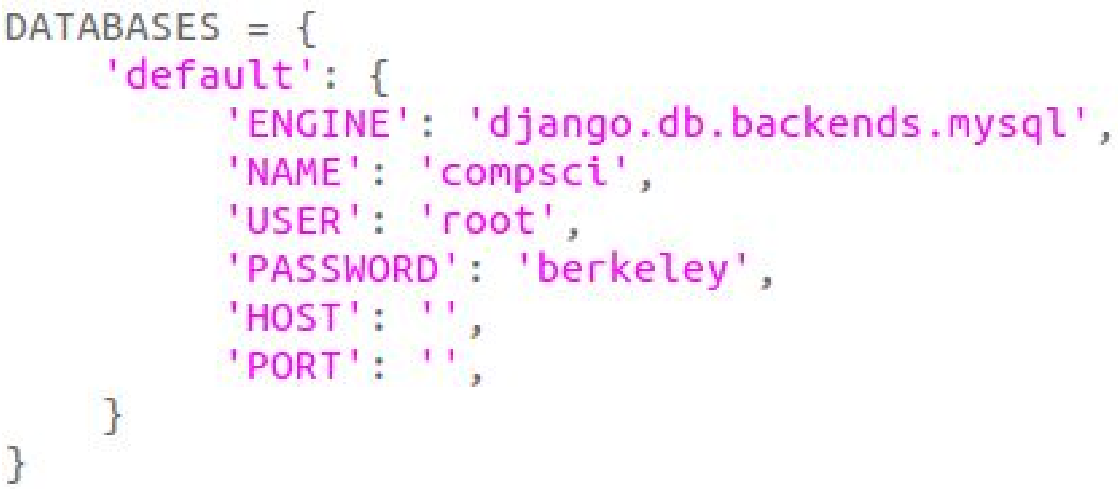
Assistants Table Description:



Field	Type	Null	Key	Default	Extra
Person_Name	varchar(64)	NO		*NULL*	
CCN	int(11)	NO	PRI	0	
Officer	bit(1)	YES		*NULL*	

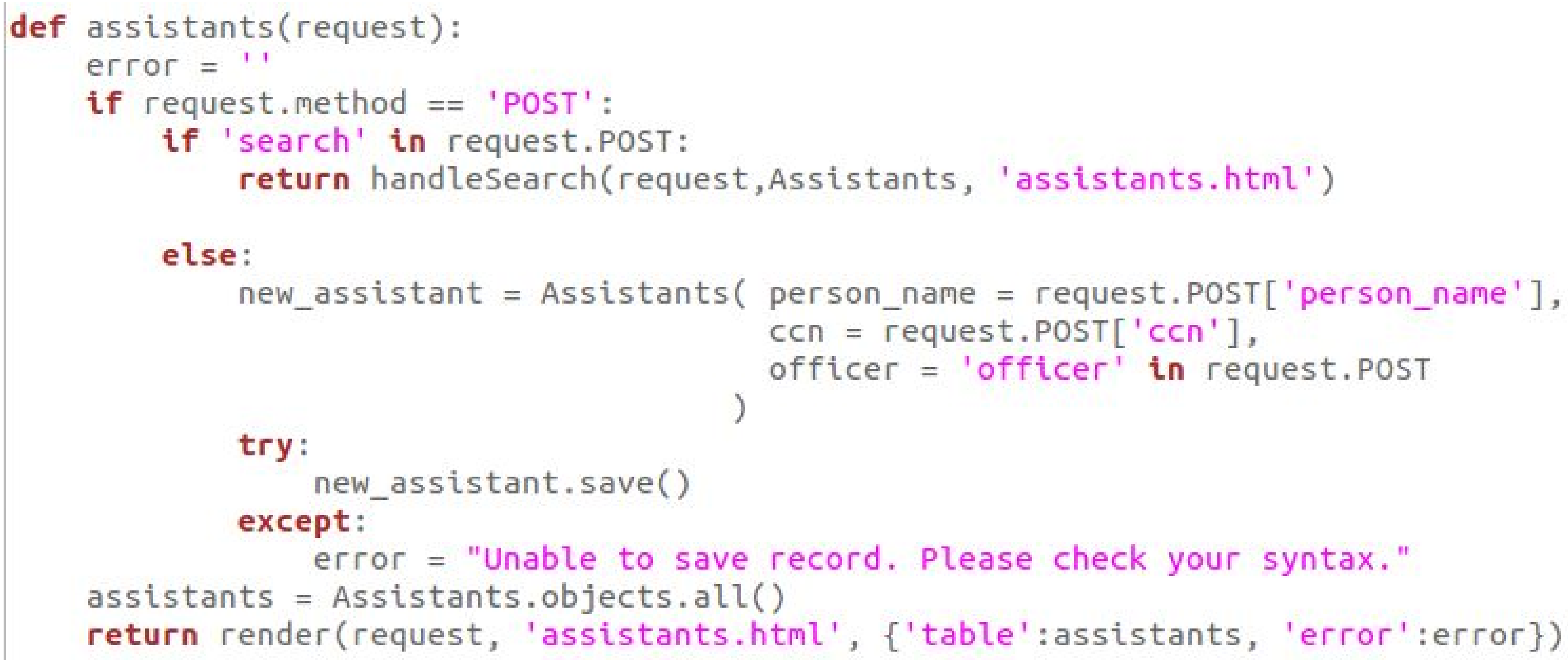
Django - connecting the database

settings.py



```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'compsci',
        'USER': 'root',
        'PASSWORD': 'berkeley',
        'HOST': '',
        'PORT': '',
    }
}
```

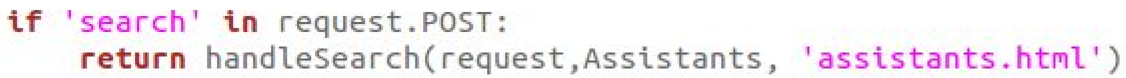
Django Views: Handling Responses



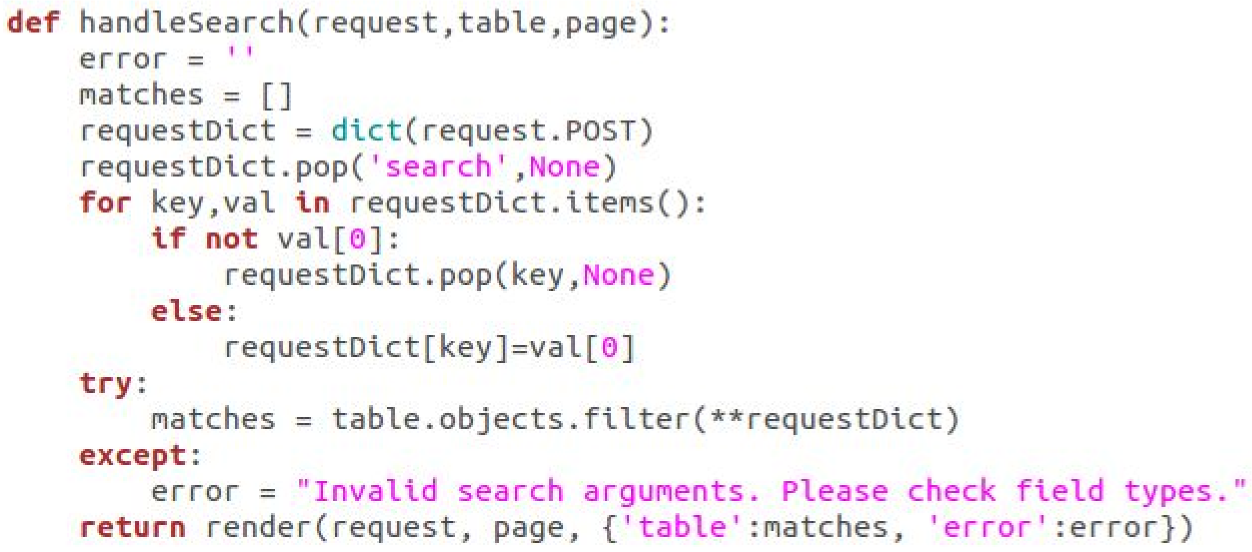
```
def assistants(request):
    error = ''
    if request.method == 'POST':
        if 'search' in request.POST:
            return handleSearch(request,Assistants, 'assistants.html')

        else:
            new_assistant = Assistants( person_name = request.POST['person_name'],
                                        ccn = request.POST['ccn'],
                                        officer = 'officer' in request.POST
                                      )
            try:
                new_assistant.save()
            except:
                error = "Unable to save record. Please check your syntax."
    assistants = Assistants.objects.all()
    return render(request, 'assistants.html', {'table':assistants, 'error':error})
```

Django Views: Handling Responses

Searching our tables (models)



```
if 'search' in request.POST:
    return handleSearch(request,Assistants, 'assistants.html')
```



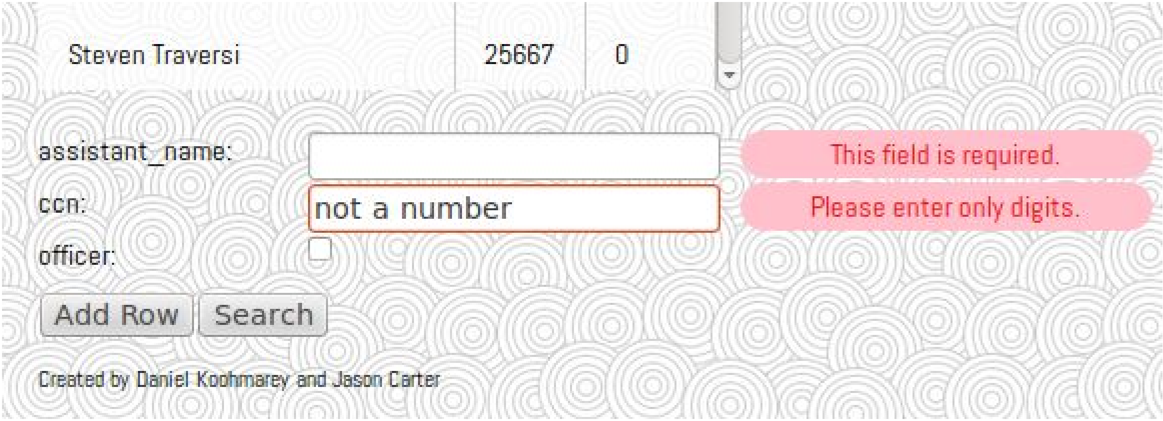
```
def handleSearch(request,table,page):
    error = ''
    matches = []
    requestDict = dict(request.POST)
    requestDict.pop('search',None)
    for key,val in requestDict.items():
        if not val[0]:
            requestDict.pop(key,None)
        else:
            requestDict[key]=val[0]
    try:
        matches = table.objects.filter(**requestDict)
    except:
        error = "Invalid search arguments. Please check field types."
    return render(request, page, {'table':matches, 'error':error})
```

User Interface: Form Validation

.validate(✓)

jQuery **Validation** Plugin

Form validation with jQuery



Steven Traversi	25667	0

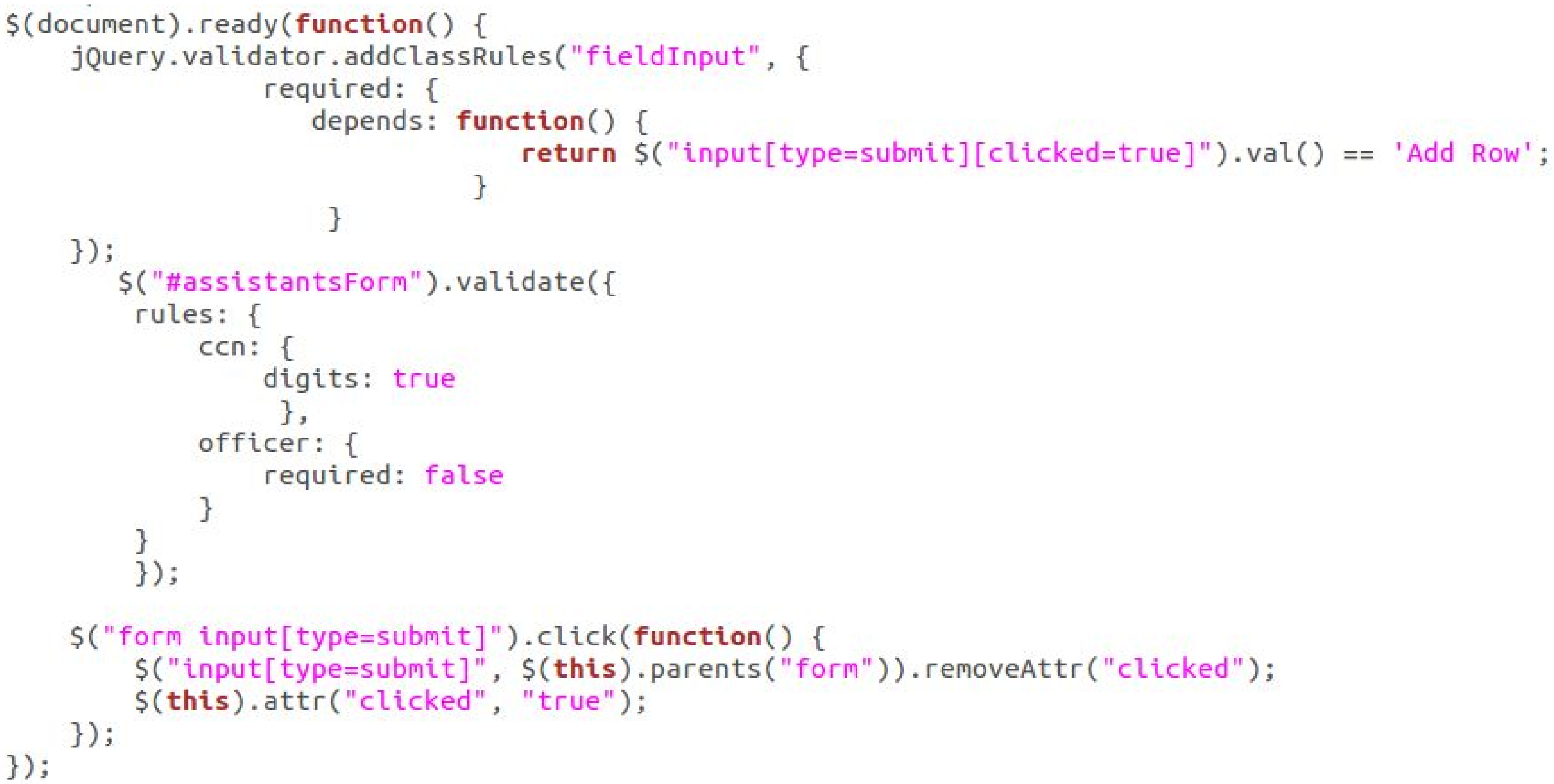
assistant_name: This field is required.

ccn: not a number Please enter only digits.

officer: □

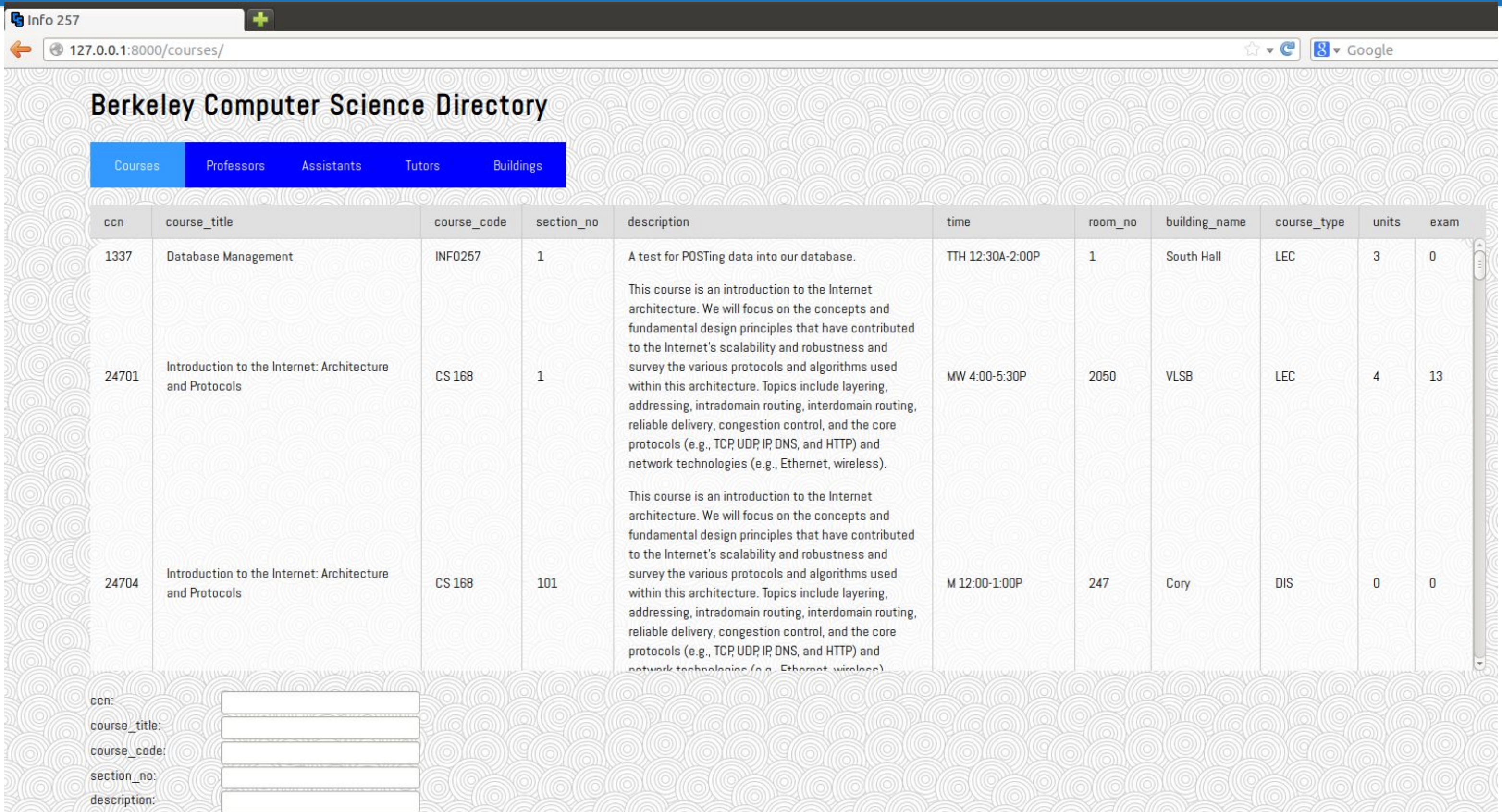
Created by Daniel Koohmarey and Jason Carter

User Interface: Form Validation



```
$(document).ready(function() {
    jQuery.validator.addClassRules("fieldInput", {
                required: {
                   depends: function() {
                                return $("input[type=submit][clicked=true]").val() == 'Add Row';
                             }
                    }
    });
       $("#assistantsForm").validate({
        rules: {
            ccn: {
                digits: true
                 },
            officer: {
                required: false
            }
        }
        });

    $("form input[type=submit]").click(function() {
        $("input[type=submit]", $(this).parents("form")).removeAttr("clicked");
        $(this).attr("clicked", "true");
    });
});
```

Live Demo



Berkeley Computer Science Directory

ccn	course_title	course_code	section_no	description	time	room_no	building_name	course_type	units	exam
1337	Database Management	INFO257	1	A test for POSTing data into our database.	TTH 12:30A-2:00P	1	South Hall	LEC	3	0
24701	Introduction to the Internet: Architecture and Protocols	CS 168	1	This course is an introduction to the Internet architecture. We will focus on the concepts and fundamental design principles that have contributed to the Internet's scalability and robustness and survey the various protocols and algorithms used within this architecture. Topics include layering, addressing, intradomain routing, interdomain routing, reliable delivery, congestion control, and the core protocols (e.g., TCP, UDP, IP, DNS, and HTTP) and network technologies (e.g., Ethernet, wireless).	MW 4:00-5:30P	2050	VLSB	LEC	4	13
24704	Introduction to the Internet: Architecture and Protocols	CS 168	101	This course is an introduction to the Internet architecture. We will focus on the concepts and fundamental design principles that have contributed to the Internet's scalability and robustness and survey the various protocols and algorithms used within this architecture. Topics include layering, addressing, intradomain routing, interdomain routing, reliable delivery, congestion control, and the core protocols (e.g., TCP, UDP, IP, DNS, and HTTP) and network technologies (e.g., Ethernet, wireless)	M 12:00-1:00P	247	Cory	DIS	0	0

ccn:

course_title:

course_code:

section_no:

description: