

## PYQ + REVISION

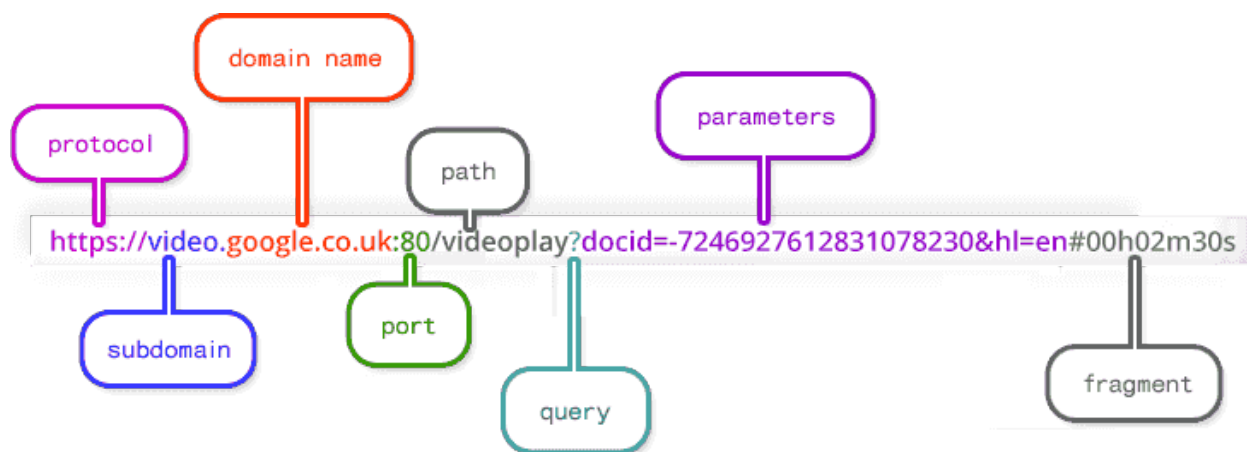
### M-V-C paradigm

- Model:
  - Core data to be stored for the application
  - Databases; indexing for easy searching, manipulation;
- View:
  - User-facing side of application
  - Interfaces for finding information, manipulating
- Controller:
  - "Business logic" - how to manipulate data

User  
uses **controller** ->  
to manipulate **model** ->  
that updates **view** ->  
that user sees

Hardly new: origins in Smalltalk language: 1979

### URL BREAKDOWN



## PYQ + REVISION

### CURL REQUEST

♥ these comics?

buy a collection!

★ wizardzines.com ★

curl



\$ curl wizardzines.com

# curl

-H

is for header

good for POST requests:

-H "Content-Type: application/json"

allow compression:

-H "Accept-Encoding: gzip"

JULIA EVANS  
@b0rk

--data  
to POST data!

--data '{"name": "julia"}'

--data @filename.json

↑  
this way is WAY better  
if you have a lot of data  
to send!

-i

show response headers

-V, -vv

verbose: show request headers etc

-I

show only response headers  
(HEAD request)

-k

insecure: don't verify SSL  
certificates

-X POST

send a POST request instead  
of a GET (-X PUT etc works too)

--cert, --key

use a client certificate

★ copy as curl ★

Have something in your  
browser you want to download  
from the command line?

In Firefox / Chrome:

Developer Tools  
→ Network tab  
→ right click on the thing  
→ copy as curl

### Basic Structure

curl [options] [URL] (localhost:5000/login[defined route])

# PYQ + REVISION

## Key Components

1. **URL:** The endpoint you're making the request to
2. **HTTP Method:**
  - **-X:** Specifies the HTTP method (e.g., `GET`, `POST`, `PUT`, `DELETE`).
  - **Example:** `-X POST`
3. **Headers:**
  - **-H:** Adds a header to the request.
  - **Example:** `-H "Content-Type: application/json"`
4. **Data:**
  - **-d:** Sends data in a request body (useful for `POST` or `PUT`).
  - **Example:** `-d '{"key": "value"}'`

## Example Request

```
curl -X POST -H "Content-Type: application/json" -d '{"name": "John"}'  
https://api.example.com/users
```

This example sends a POST request to `https://api.example.com/users` with a JSON body `{"name": "John"}` and sets the `Content-Type` header to `application/json`.

Sep 23 : 640653697642

Sep 23 : 640653697654

## PYQ + REVISION

Sep 23 : 640653697633

|

**MIME TYPE** : A MIME type (Multipurpose Internet Mail Extensions type) is a standardized way to indicate the nature and format of a file or data being transmitted over the internet. MIME types are used in various internet protocols like HTTP to describe the content of web pages, email attachments, and other forms of media.

## HTTP STATUS CODES

Sept : 640653697635

200 , 201 , 404 , 400 , 405

200—when it sends the response successfully

201—let's say i want to create a new user and i am sending a post request server creates a new user and sends this 201 code

404—url not found server can't find the url that user is visiting

405—Method not Allowed

400—Bad request

Refer to the IMAGE BELOW

## Methods

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods>

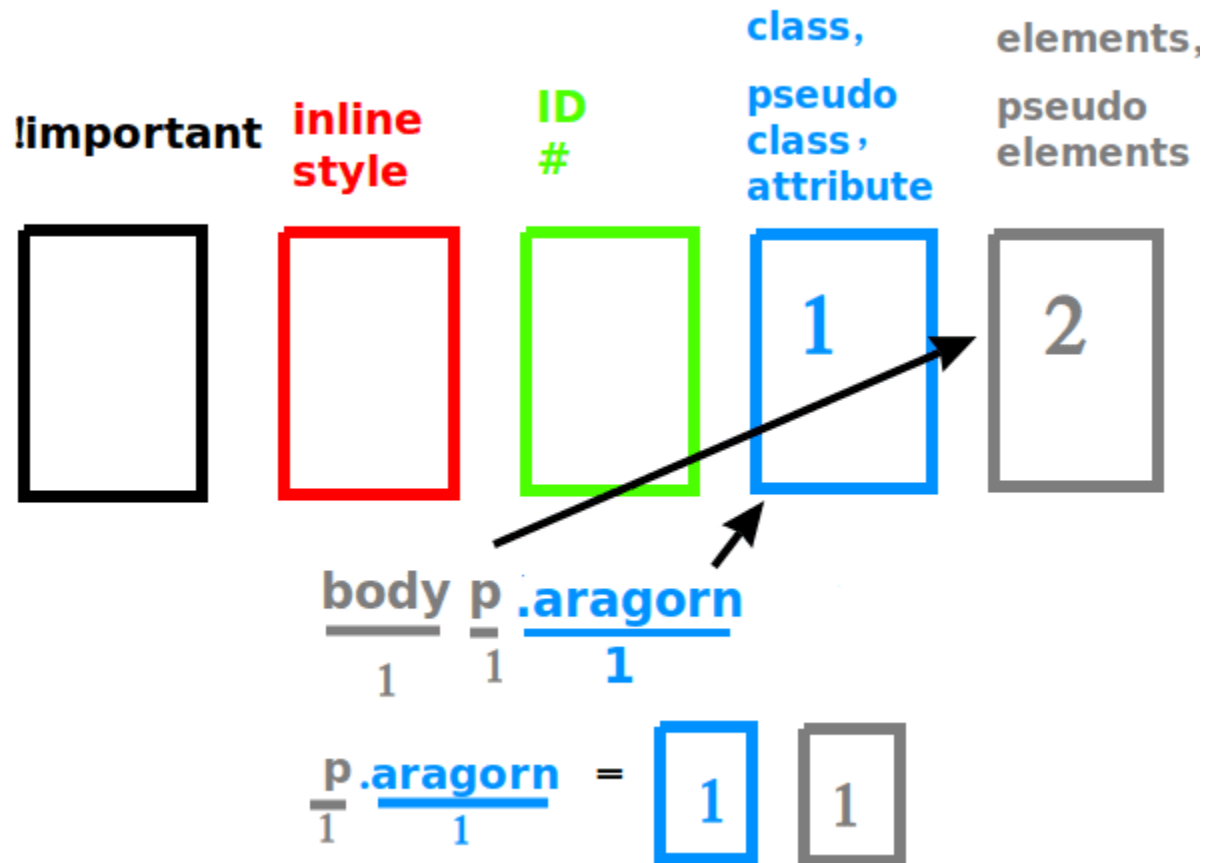
## PYQ + REVISION

## PYQ + REVISION

<b>1XX</b> Informational Requests	100 Continue 101 Switching Protocols 102 Processing
<b>2XX</b> Successful Requests	200 OK 201 Created 202 Accepted 203 Non-Authoritative Information 204 No Content 205 Reset Content 206 Partial Content 207 Multi-Status 208 Already Reported
<b>3XX</b> Redirects	300 Multiple Choices 301 Moved Permanently 302 Found 303 See Other 304 Not Modified 305 Use Proxy 307 Temporary Redirect 308 Permanent Redirect
<b>4XX</b> Client Errors	400 Bad Request 401 Unauthorized 402 Payment Required 403 Forbidden 404 Not Found 405 Method Not Allowed 407 Proxy Authentication Required 408 Request Timeout 409 Conflict 410 Gone 412 Precondition Failed 416 Request Range Not Satisfiable 417 Expectation Failed 422 Unprocessable Entity 423 Locked 424 Failed Dependency 426 Upgrade Required 429 Too Many Requests 431 Request Header Fields Too Large 451 Unavailable for Legal Reasons
<b>5XX</b> Server Errors	500 Internal Server Error 501 Not Implemented 502 Bad Gateway 503 Service Unavailable 504 Gateway Timeout 505 HTTP Version Not Supported 506 Variant Also Negotiates 507 Insufficient Storage 508 Loop Detected 510 Not Extended 511 Network Authentication Required

## PYQ + REVISION

### HTML & CSS



Left to Right Descending  
Left has highest Priority

1. Read the code thoroughly don't rush
2. !important is the most important
3. UL & OL
4. SPAN is a big SCAM—(Let's go to Onenotes[display:block])

## **PYQ + REVISION**

**Sep 23 : 640653697628**

**Sep 23 : 640653697624**

**Sep 23 : 640653697650**

**Sep 23 : 640653697630**

**Jan 24 : 640653697630**

**Jan 24 : 640653816150**

**Accessibility section week 3**

## **JINJA & STRING**

**One-note is best!!!!**

**Sep 23 : 640653697632**

**Sep 23 : 640653697644**

**Jan 24 : 640653816143**

**Jan 24 : 640653816131**

**Jan 24 : 640653816125**

**Jan 24 : 640653816149**

**Template Inheritance**

**Jan 24 : 640653816139**



## PYQ + REVISION

### SYS.ARGV (Command Line Arguments)

```
C:\Users\Vanshi>cd Desktop

C:\Users\Vanshi\Desktop>python com.py 1 2 3 4
This is the name of the program: com.py
Argument List: ['com.py', '1', '2', '3', '4']

C:\Users\Vanshi\Desktop>_
```

1. “{{SPACE}}” is considered as a separator.
2. Python will not be considered as an argument
3. Easiest questions

Sep 23 : 640653697631

Sep 23 : 640653697629

Jan 24 : 640653816133

## FLASK ROUTES

1. Routes
2. How we read parameters from the routes
3. Parser
4. parameters (?) & how do we read them
5. Error handler

## **PYQ + REVISION**

Sep 23 : 640653697645  
Sep 23 : 640653697653  
Sep 23 : 640653697637  
Sep 23 : 640653697638  
Sep 23 : 640653697652  
Jan 24 : 640653816126  
Jan 24 : 640653816128  
Jan 24 : 640653816137  
Jan 24 : 640653816127

## **Sql-alchemy**

1. Why do we use it
2. ORM
3. Add remove objects (session)
4. Relationships
5. Creating and filtering objects

Sep 23 : 640653697643  
Sep 23 : 640653697639  
Sep 23 : 640653697656  
Sep 23 : 640653697648  
Jan 24 : 640653816132  
Jan 24 : 640653816155  
Jan 24 : 640653816129

## **PYQ + REVISION**

### **REST-ful API**

1. **How do we use it**
2. **Add resources**
3. **Methods as functions**

**Jan 24 : 640653816144**

## PYQ + REVISION