

View Functions with Custom Error Pages



Abdul Rehman

MACHINE LEARNING ENGINEER

@aPythonist www.pythonist.org

Overview



View Functions

Focused on Response

Custom Error Pages



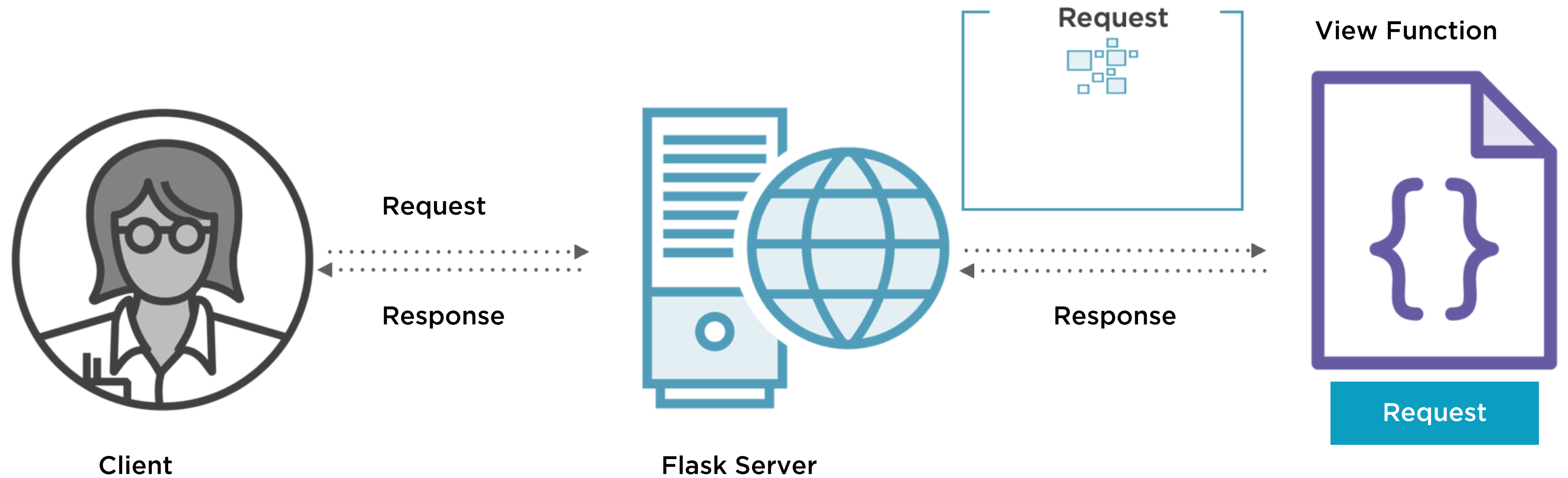
Let's get started!

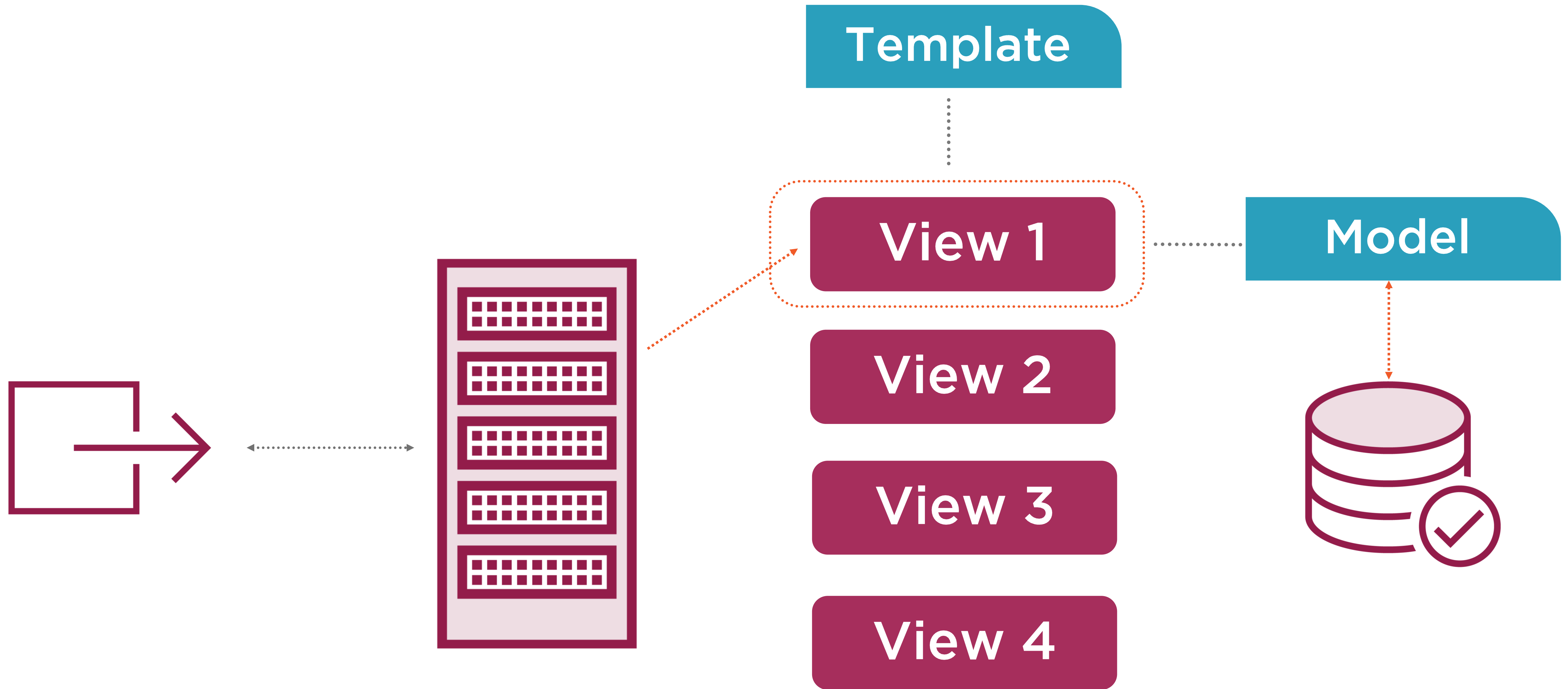


View Functions

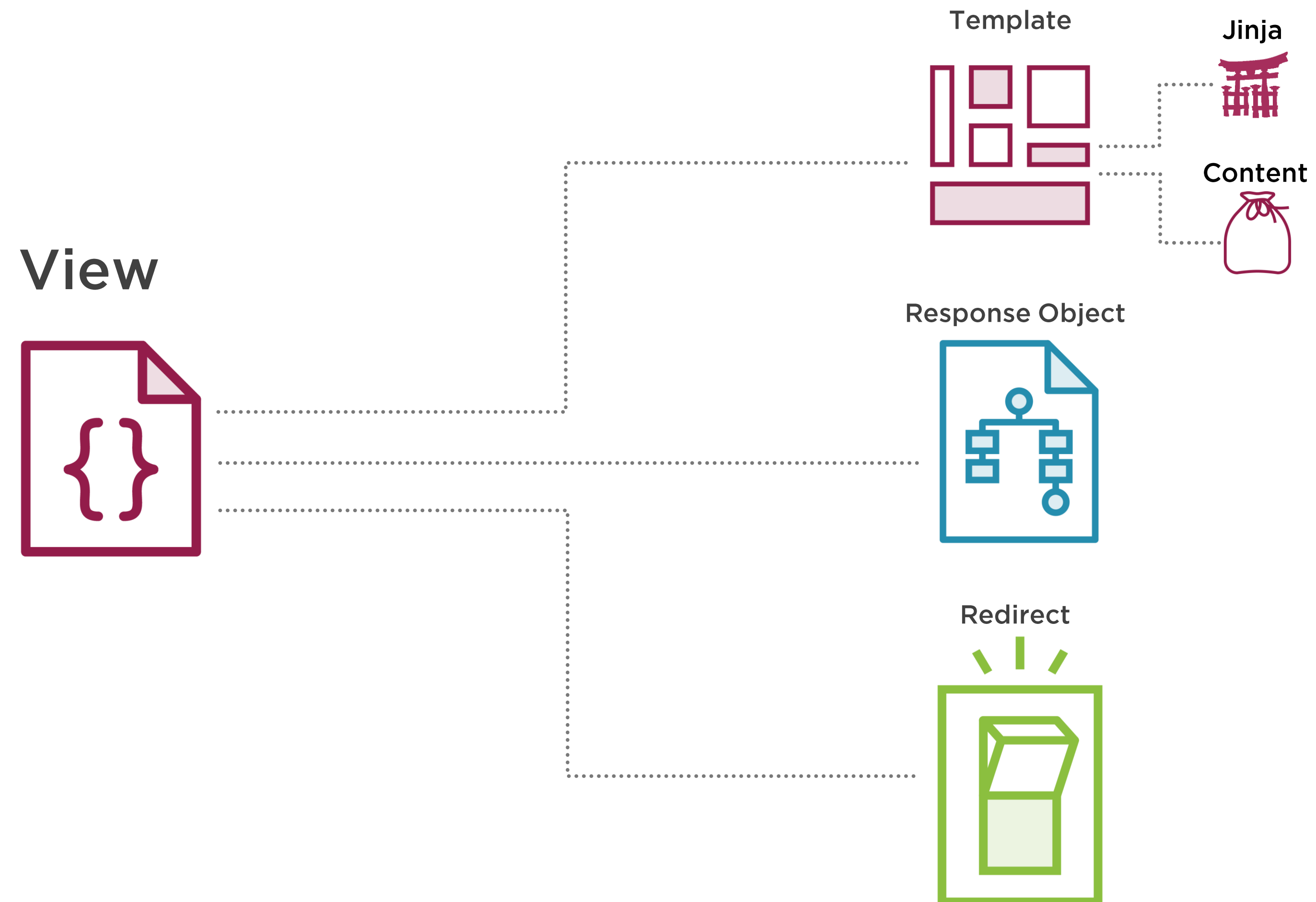


Request-response Cycle





Role of View Functions



Demo



Implementation of Template Response



Response Object

Have an HTML mimetype by default

Make_response() helps to create response

Subclass Response in order to customize

Make_response()

Convert return value to response_class instance

Types of return value:

String

Bytes

Dict

Tuple

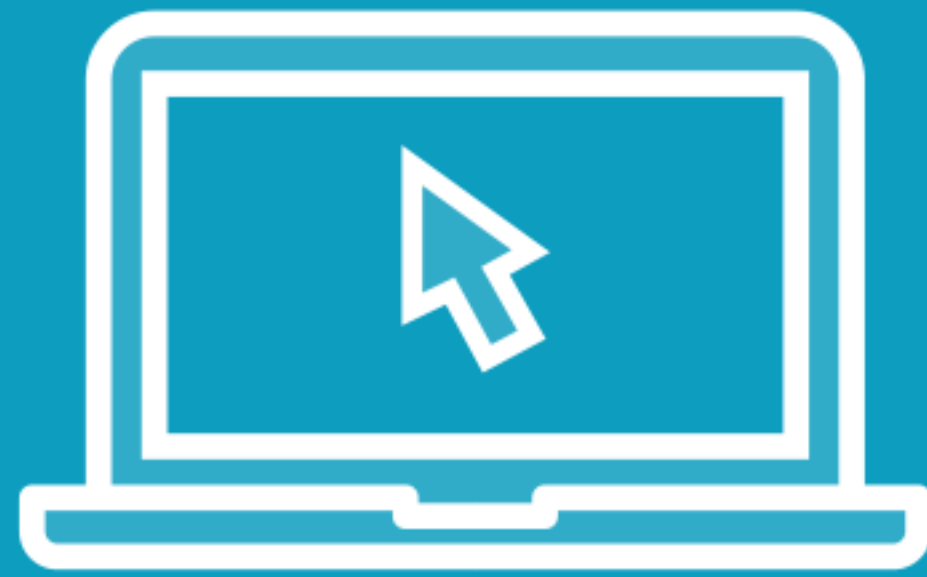
Callable



A view function must return a response. Returning None or the view without a return is not allowed



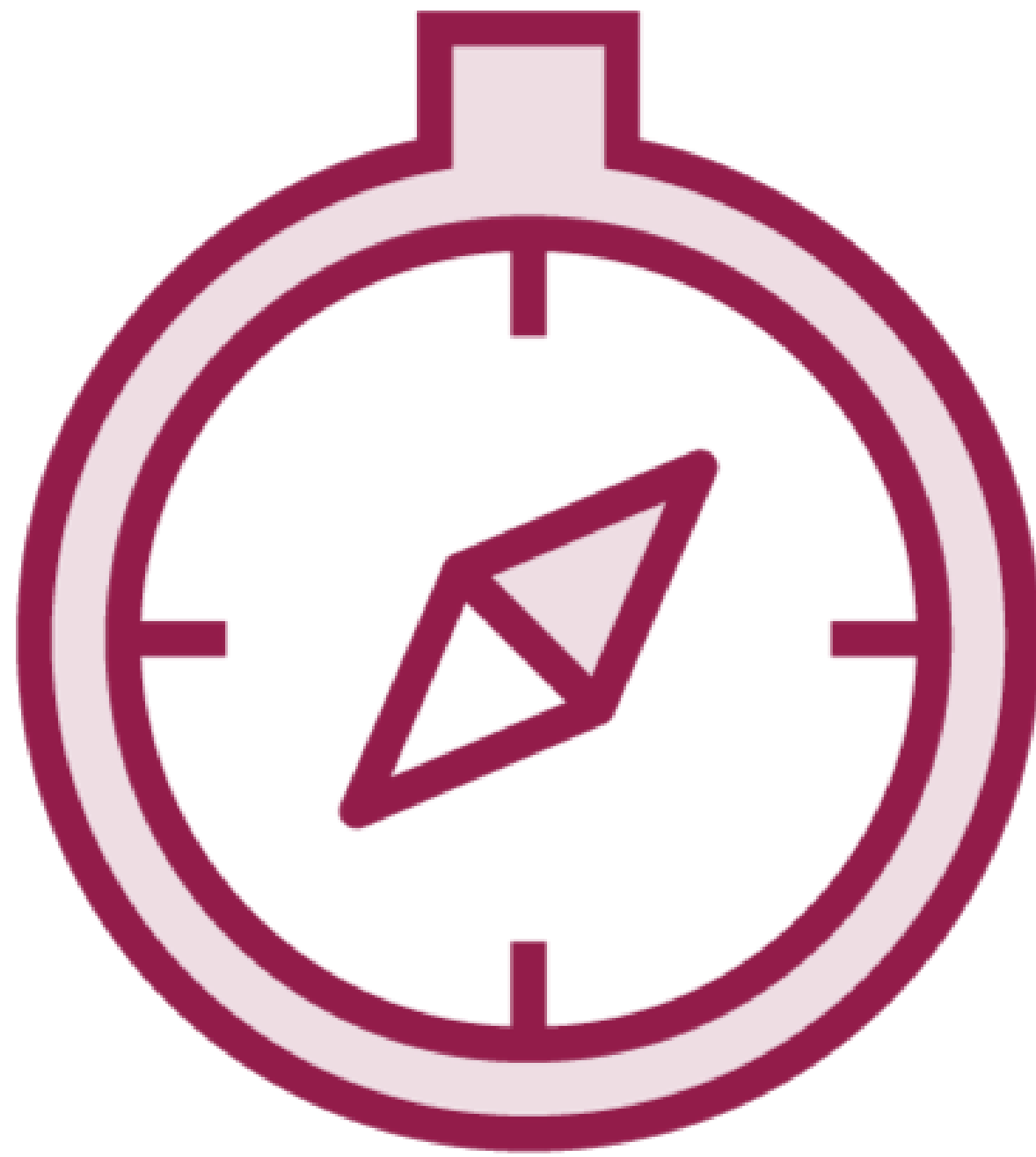
Demo



Implementation of Response Object



Redirect



Redirects the client to the target location

Supported codes: 301, 302, 303, 305, 307 & 308

300 is not real redirect

304 is the answer to a request with **IF-Modified-Since defined**

Redirect function arguments:

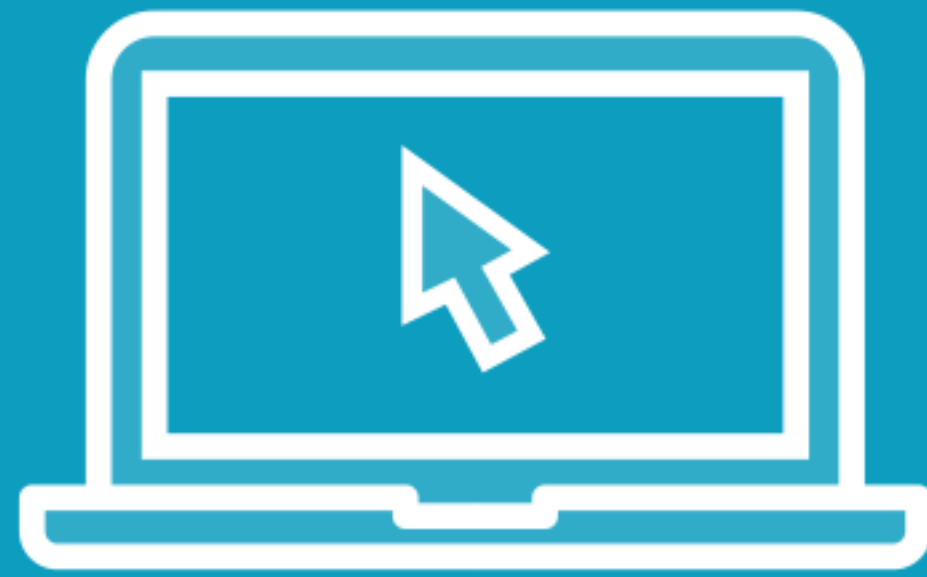
Location: Location the response should redirect to

Code: Redirect HTTP status code – Default is 302

Response: Response class to use for instantiating a response



Demo



Implementation of Redirect



Summary



Ready to implement custom error handling in flask

How to Generate customized responses from view functions



Up Next:

Custom Error Pages in Flask applications



Custom Error Pages



Error Handling

Bad error handling can cause frustration for the end-users of your application

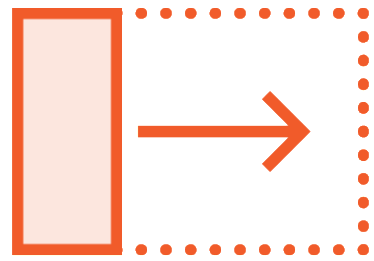
ill-Constructed error messages can make users to stop using your application

Flask comes with abort function

Custom error-handling



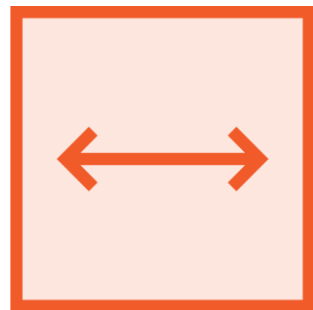
Flask's Abort Function



Takes two arguments: **Status Code & Description**



Look up for exception according to the status code



Wrap the Response object in a proxy WSGI exception and raise it



If we want to design a proper error page and return that out in case of that error in our entire application,
what will we do?



Custom Error Handler

An error handler is a function, just like a view function, but it is called when an error happens and is passed that error. Error is most likely an `HTTPException` except in case of Internal Server Error.

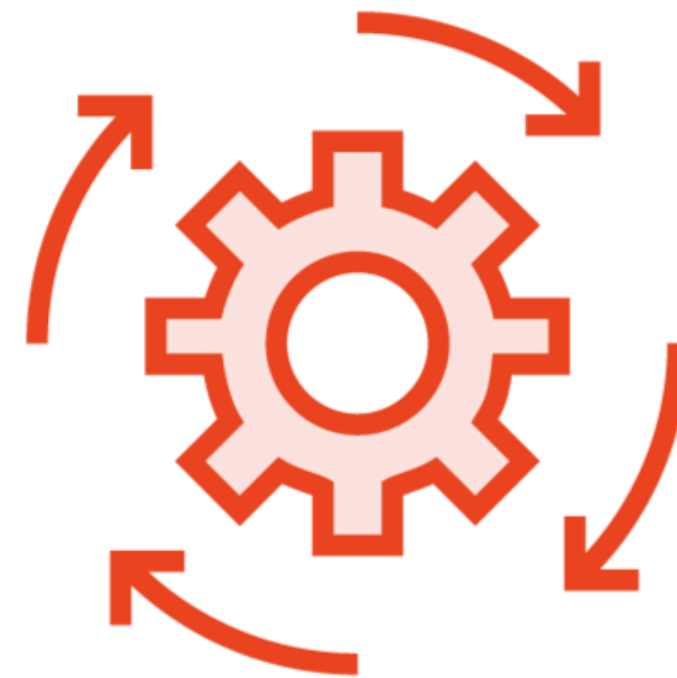


Custom Error-handler



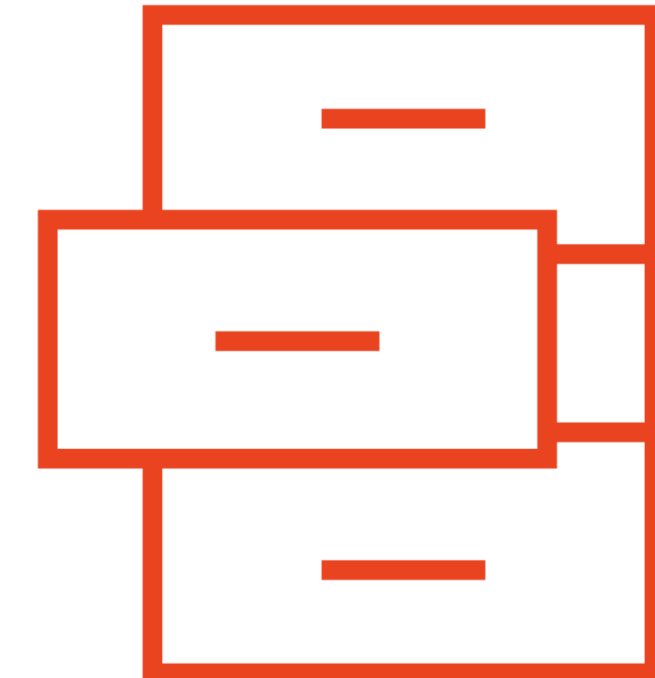
Definition

Registered with `errorhandler()` decorator and error code.



Provide Error Code

Make sure that you provided the HTTP error code.

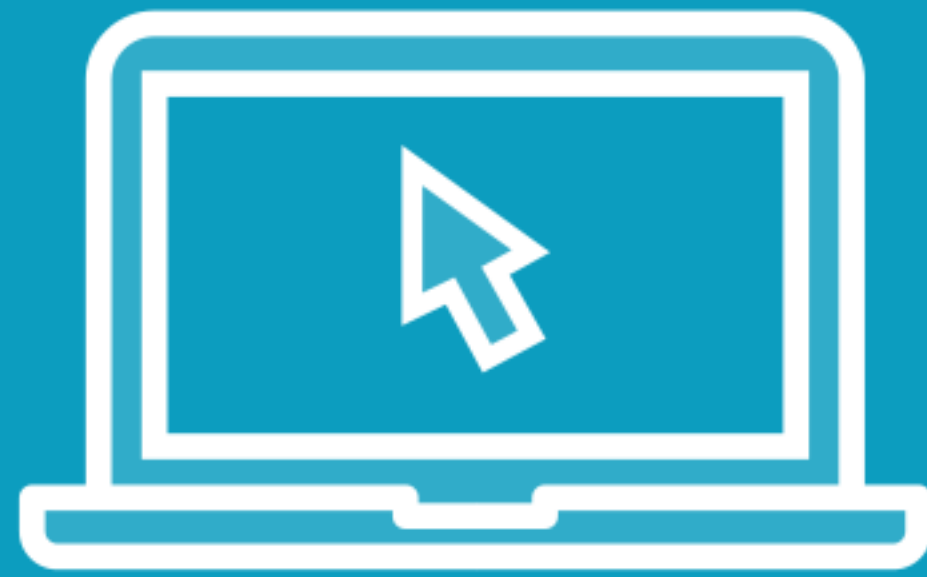


Debug Mode Off

“500 Internal Server Error” will not work If app is working in debug mode.



Demo



Implement errors for

404 Page Not Found

500 Internal Server Error



Up Next:
Implement Logging in Flask applications



“The best error message is the one that never shows up.”

Thomas Fuchs



Summary



Customized View Functions

Customized Error Handling

Up Next:
Logging in Flask

