FOUR FUNCTIONS IN THE STRUCTURE OF TECHNICAL DOCUMENTATION (AND WHY THEY MATTER)
WHAT NOBODY TELLS YOU ABOUT DOCUMENTATION

DANIELE PROCIDA

FOUR FUNCTIONS IN THE STRUCTURE OF TECHNICAL DOCUMENTATION (AND WHY THEY MATTER)
DANIELE PROCIDA

- daniele.procida@divio.com
- EvilDMP (IRC, GitHub, Twitter)
- ... or just talk to me!
Autopilot for your Django websites
djangoproject.com

“The web framework for perfectionists with deadlines”

django
Django core developer, Django Software Foundation Vice President
Python in Africa

- 2015: PyCon Namibia
- 2016: PyCon Zimbabwe
- 2017: PyCon Nigeria
- 2018: PyCon Ghana
- 2019: PyCon Africa
TRYING TO KEEP UP WITH SUCCESS

Documentation able to respond to...

- new users!
- lots of new users!
- new functionality!
- new demands!
- new use-cases!
- more of everything!
PLEASD TO REPORT: THAT PROBLEM IS NOW SOLVED
THERE ISN’T ONE THING CALLED “DOCUMENTATION”...
TUTORIALS

lessons that take the reader by the hand through a series of steps to complete a project
WHAT MATTERS

- learning by doing
- getting started
- inspiring confidence
- repeatability
- immediate sense of achievement
- concreteness, not abstraction
- minimum necessary explanation
- no distractions
HOW-TO GUIDES

Guides that take the reader through the steps required to solve a common problem.
HOW-TO GUIDES

WHAT MATTERS

▸ a series of steps
▸ a focus on the goal
▸ addressing a specific question
▸ no unnecessary explanation
▸ a little flexibility
▸ practical usability
▸ good naming

Crab with wakame salad and wasabi mayonnaise

Try this simple to prepare salad of white crabmeat served with cucumber, seaweed and spicy mayonnaise.

By Rick Stein
From Spring Kitchen with Tom Kerridge

Ingredients

40g/1½oz dried wakame seaweed
½ cucumber, peeled
225g/8oz fresh white crabmeat
1 tbsp bonito flakes

For the wakame dressing

½ tsp dashi granules
40ml/1½fl oz warm water
8ml rice wine vinegar
REFERENCE

technical descriptions of the machinery and its operation
What Matters

▸ structure
▸ consistency
▸ description
▸ accuracy
EXPLANATION

discussions that clarify and illuminate a particular topic
WHAT MATTERS

▸ giving context
▸ explaining why
▸ multiple examples, alternative approaches
▸ making connections
▸ no instruction or technical description
<table>
<thead>
<tr>
<th>Category</th>
<th>Orientation</th>
</tr>
</thead>
<tbody>
<tr>
<td>TUTORIALS</td>
<td>Learning-oriented</td>
</tr>
<tr>
<td>HOW-TO GUIDES</td>
<td>Problem-oriented</td>
</tr>
<tr>
<td>UNDERSTANDING-Oriented</td>
<td>Understanding-oriented</td>
</tr>
<tr>
<td>EXPLANATION</td>
<td>Information-oriented</td>
</tr>
<tr>
<td>REFERENCE</td>
<td></td>
</tr>
</tbody>
</table>
HOW-TO GUIDES
Most useful when we’re working
REFERENCE
TUTORIALS
Most useful when we’re studying
EXPLANATION
TUTORIALS
LEARNING-ORIENTED
Most useful when we’re studying
UNDERSTANDING-ORIENTED
EXPLANATION

HOW-TO GUIDES
PROBLEM-ORIENTED
Most useful when we’re working
INFORMATION-ORIENTED
REFERENCE
Django documentation

Everything you need to know about Django.

How the documentation is organized

Django has a lot of documentation. A high-level overview of how it’s organized will help you know where to look for certain things:

- **Tutorials** take you by the hand through a series of steps to create a Web application. Start here if you’re new to Django or Web application development. Also look at the “First steps” below.

- **Topic guides** discuss key topics and concepts at a fairly high level and provide useful background information and explanation.

- **Reference guides** contain technical reference for APIs and other aspects of Django’s machinery. They describe how it works and how to use it but assume that you have a basic understanding of key concepts.

- **How-to guides** are recipes. They guide you through the steps involved in addressing key problems and use-cases. They are more advanced than tutorials and assume some knowledge of how Django works.
Overview

django CMS is a modern web publishing platform built with Django, the web application framework “for perfectionists with deadlines”.

django CMS offers out-of-the-box support for the common features you’d expect from a CMS, but can also be easily customised and extended by developers to create a site that is tailored to their precise needs.

Tutorials - start here

For the new django CMS developer, from installation to creating your own addon applications.

Key topics

Explanation and analysis of some key concepts in django CMS.

How-to guides

Practical step-by-step guides for the more experienced developer, covering several important topics.

Reference

Technical reference material, for classes, methods, APIs, commands.
Welcome to Nashpy’s documentation!

This is a Python library used for the computation of equilibria in 2 player strategic form games.

This is a library with simple dependencies (it only requires numpy and scipy) so that it is pip installable: if you want to do sophisticated equilibria computation you should use gambit.

- Tutorial: building and finding the equilibrium for a simple game
  - Introduction to game theory
  - Installing Nashpy
  - Creating a game
  - Calculating the utility of a pair of strategies
  - Computing Nash equilibria
- How to
  - Install Nashpy
  - Create a game
  - Calculate utilities
  - Solve with support enumeration
  - Solve with vertex enumeration
  - Find equilibria for large games
- Reference
  - Support enumeration
  - Vertex enumeration
  - Discussion
  - Bibliography
  - Source files
- Discussion
  - John Nash
  - How does Nashpy relate to Gambit
  - Other Python Game theory libraries
ANOTHER EXAMPLE
TUTORIALS

LEARNING-ORIENTED LESSONS
HOW-TO GUIDES

PROBLEM-ORIENTED STEPS
SEABOARD WORLD AIRLINES 747F
NORMAL OPERATING CHECKLIST

BEFORE TAXI
START ARM SWITCH ....................... OFF
ELECTRICAL POWER ................... SET
APU BLEED ............................. CLOSE
HYDRAULICS .......................... AUTO/NORMAL/OQTY CKD
SEAT BELTS & SHLD HARNESS ........ ON
GEAR & NOSE STEER PINS .............. REMOVED/CKD
GROUND EQUIPMENT ................. DISCONNECT/CLEAR

18,000 FT OR/TRANSITION LEVEL CHECK
ALTIMETERS .......................... SET/X-CKD
Landing Data ......................... SET
Landing & Logo Lights ............. ON/10,000'

APPROACH
FLAPS .................................... GREEN LIGHT/DETENT
ADF/VOR SWITCHES ................. SET
Radio Alt ................. MDA/DH SET
NACELLE ANTI-ICE ................... SET
FUEL SYS ................... MAIN BOOST ON/HEAT OFF
NO SMOKE ..................... ON

BEFORE LANDING
Landing Gear ............... DOWN-GREEN LIGHT
Auto Brake ................. SET/LT OUT
Speed Brake .......... ARM
FLAPS .................................... SET
Flag Scan OM-500' .......... CALL OUT

BEFORE TAKE OFF
Landing & Strobe Lights ............ ON
Transponder ................. ON
Auto Brake ................. ARM
Body Gear ................. DISARM

BEFORE STARTING
INS ................................... 3 CKD/ALIGN
O₂ & Interphone ................. ON 100% CKD/BOOM
STATIC SOURCE SEL ............... NORMAL
A-SKID ................................. ON
Body Gear Steering ............... ARM
Auto Brake ..................... LDG-OFF
COMPASS CONTROLLERS ........... SLAVED
Emergency Lights ............... ARMD
Seat Belt, No Smoke ...... OFF
Alt Flaps ....................... OFF
Stall Warning .................... TEST/NORMAL
Mach A/S ......................... TEST
Nacelle & Wing Anti-Ice ...... OFF
Probe Heat .................... PITOTS ONLY
Window Hea........................ TEST
EXTERIOR LIGHTS .............. SET
Radio Ins Switch ................. RADIO
Nav Radios/Auto Flt Panel .. RADIO
Ground Prox ....................... TEST
FLT Mode Annunciators .......... SET
FLT INSTR/FLT DIR/ALT'S ..... CKD/TEST/SET
Radio Alt ..................... TEST
Reserve Brake ................... CKD/CLOSED
Ldg Gear ..................... DOWN/GREEN
Speed Brake .................... FWD DETENT
Throttles/Start Levers .......... CLOSED/CUTOFF
Park Brake ...................... SET/PRESS CKD
SELCAL/Radar & Transponder .... SET/STBY
Electrical Panel .......... SET
Oil Quantity ..................... NORMAL
Fuel Qty/Gross Wt .............. LBS/SET
Wing Flap ....................... 00-00

CLIMB
Landing Gear ............... UP & OFF
FLAPS .............................. UP-LIGHTS OUT
PROBE HEAT ..................... ON
NO SMOKE ..................... OFF
Ignition ....................... SET
Fuel Sched ..................... SET
Air Cond ....................... ARM

AFTER LANDING
Body Gear Steering .......... ARMED
Speed Brake .................... DOWN/DETENT
FLAPS .............................. UP/LTS OUT
Ldg Lts & Strobe Lights ...... SET
IGNITION ....................... OFF
Radar & Transponder .......... OFF
Stabilizer Trim ................. 5 SET
Brake Temp & Hydraulics ...... CKD
Anti-Skid Ground Mode .......... SET
Test Upper Deck & Cargo Heat .. OFF
Fire Warning .................... TEST
APU ....................... START

PARKING
Parking Brake ................. SET
APU or External Power .......... CONNECTED
Start Levers .................... OFF
REFERENCE

INFORMATION-ORIENTED TECHNICAL DESCRIPTION
EXPLANATION
UNDERSTANDING-ORIENTED DISCUSSION
begins to fork here
Practical steps
Most useful when we’re studying

---

Theoretical knowledge
Most useful when we’re working

begins to fork here
ALL ABOUT ME

DANIELE PROCIDA

daniele.procida@divio.com

EvilDMP (IRC, GitHub, Twitter)

... or just talk to me!
DOCUMENT THE GAME OF CHESS
<table>
<thead>
<tr>
<th>Introduction</th>
<th>How to...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation</td>
<td>Reference</td>
</tr>
</tbody>
</table>

**Explanation**

**Reference**
**Introduction - your first game**

1. Set up the board
2. Take each piece through its moves
3. Capture a piece
4. Check the King
5. Checkmate - you win!

**How to...**

- Open the game
- Respond to common openings
- Control the centre of the board
- Use a chess clock

**Explanation**

- The history of chess
- Middle-game strategies
- End-game strategies
- Numerical and positional advantage

**Reference**

- The rules of chess
- Competition rules
- Standard competition formats
LET'S DOCUMENT OUR SOFTWARE

Our API allows us to interrogate and manage all the data associated with a conference - people, presentations, tickets, schedule, etc.

Your job: write the documentation.
<table>
<thead>
<tr>
<th>Introduction</th>
<th>How to...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explanation</td>
<td>Reference</td>
</tr>
<tr>
<td>Introduction</td>
<td>How to...</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------------</td>
</tr>
<tr>
<td>1. Install the client and demo server</td>
<td>‣ Embed the client in an application</td>
</tr>
<tr>
<td>2. Authenticate</td>
<td>‣ Authenticate using LDAP</td>
</tr>
<tr>
<td>3. Read data</td>
<td>‣ Lock the database for complex writes</td>
</tr>
<tr>
<td>4. Construct a complex query</td>
<td>‣ Use query batching</td>
</tr>
<tr>
<td>5. Write data</td>
<td>‣ Use an alternative client</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Explanation</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>‣ When not to use the API</td>
<td>‣ Client commands and options</td>
</tr>
<tr>
<td>‣ Batching vs caching</td>
<td>‣ Data schema format</td>
</tr>
<tr>
<td>‣ Designing complex queries</td>
<td>‣ API query language</td>
</tr>
<tr>
<td>‣ Performance-optimisation strategies</td>
<td>‣ The authentication system</td>
</tr>
<tr>
<td>‣ Working with large databases</td>
<td>‣ Error codes</td>
</tr>
</tbody>
</table>
TALK TO ME

DANIELE PROCIDA

- Divio
- Django
- dockerised Django deployment
- documentation
- daniele.procida@divio.com
- EvilDMP (IRC, GitHub, Twitter)