

Prompt: given docs-community.rtfid.io, make a contribution

I have put a [FIX] next to any part of the notes where I've suggested a change to the docs or docs process. You can do a ctrl-F for it, should be 14 potential fixes (15 results when including this paragraph). Bolded ones that I feel confident about and that I could submit a PR for, other things I'd want approval/discussion on or require knowledge I don't have.

Also, a general suggestion: this wasn't as hard as I expected. I would do this again and submit more docs PRs if there was a social element, like a monthly (quarterly?) docs sprint where I got to hang out & catch up with community members, but I don't know that I'd make time to keep contributing on any kind of regular basis otherwise.

Notes

initial reading of docs-community.rtfid.io (I wandered into the devguide too)

- [FIX] minor and a matter of taste, but there's a lot of clicking around in onboarding; if possible, a smaller number of pages with judicious use of anchors and ToCs might be nice
- **[FIX]** somewhat confusing whether the issue to search for issue trackers is [docs-community/issues](#) (linked first at [docs-community.readthedocs.io \[...\]](#) [contributing.html](#)) or [cpython/labels/docs](#) (linked slightly later [docs-community.readthedocs.io\[...\]/community-guide.html#how-can-i-help](#)) - might make sense to indicate at the first link that these issues are meta-issues around docs
- **[FIX]** minor, but in [this section about the diataxis framework](#) it's confusing what "explanations" are. on a meta level it feels like you're trying to fill out the diataxis framework by having a section on explanation and point out where it is in your code, but on first read (before clicking the link to learn about diataxis) I was trying to figure out where on the HOWTO page "explanation" was. I might edit that bullet point to say, "There is no section dedicated to explanations but these can be found throughout Python's documentation, for example the Unicode HOWTO contains ample explanation throughout."

finding and investigating an issue

- searched the issues, found this one from a week ago: "[Inappropriateness in unittest documentation with actual behavior](#)"

- seems somewhat straightforward to fix (remove the info about the mock behavior) but I have some additional questions:
 - you *can* get this behavior if you use the unsafe flag, so maybe it should still be documented? but it seems weird to include a fairly long piece of the docs for an edge case induced by a flag you have to actively pass in.
 - it's also default behavior for 3.5 and older but 3.5 is no longer supported; presumably that means it shouldn't be in the docs at all but it's not clear how the docs handle material that's changed from version to version
 - *I will attempt to look these questions up (how to flag changes between versions, and how to document edge cases)*
 - [FIX] I keep confusing docs-community.readthedocs.io with <https://devguide.python.org>, is there some way to more easily disambiguate? even a different color background might help
 - [FIX] the combination of the docs being part of a broader dev guide means it's hard to search just the docs on a site-wide search. but the fact that things are mostly broken up into their own page means it's hard to do a control+f. 🙄 - maybe put things into single pages more?
 - okay, searching is not working, I will skim the ToC to see if anything looks promising.
 - I don't think it's there? I'm going to ask in Discord.
 - Wait maybe I can figure this out for myself by going to the docs and finding an example. I think I need to use the release notes to find something that's changed, then look at its related docs.
 - okay, so I've found some little footnote-y things like:
 - "Added in version 3.5.2."
 - "Changed in version 3.10: NewType is now a class rather than a function. As a result, there is some additional runtime cost when calling NewType over a regular function."
 - "Changed in version 3.7: [Generic](#) no longer has a custom metaclass."
 - So this suggests that you *should* mark changes as far back as 3.5 but it's still not clear. Okay, going back to discord to ask.
 - I asked in Discord, everyone was very nice (although I had a little hesitation just posted my question to #general when often discords have a special #questions or #getting-help channel) but there doesn't seem to be guidance. Maybe there should be a lil guide? [FIX]
 - Hugo in Discord noted that there's [markup](#) for these little changes which should probably be in any lil guide, but I still need to know when to use them
 - So that was the question about versions, I'm still trying to figure out how the flag impacts how things should be documented. Going back to the issue

- It's actually kind of confusing what this "autospeccing" section is. It appears to be the only section on the entire page that doesn't begin with a reference to a method. Is it missing a method display, or does it not have one?
 - Should it reference this?
 - <https://github.com/python/cpython/blob/bbb49888a752869ae93423c42039a3a8dfab34d4/Lib/unittest/mock.py#L2707>
 - Meta-question: is it better to ask in Discord or submit PR and get feedback? Gonna ask the meta-question in Discord.
- It honestly feels like more of a tutorial than reference material. The length and depth of it makes me nervous about removing material when I don't understand it all, but also...why is it so different from its surrounding docs?
 - I am going to submit a PR anyway because my goal today is to test the contribution process, not actually fix anything, but if I was really trying to fix it I'd probably stop here (or ask for help in Discord, I guess)
 - In retrospect I should have looked in the issue tracker for docs related to a piece of python I've used before, instead of diving deep into whatever catches my eye lol
- Okay, my plan will be:
 - get set up to contribute to docs
 - make the following change:
 - add reference to create_autospec
 - keep the verbose description, but remove the reference to the issue with misspelled asserts
 - add a version changed for 3.5 and a little detail there about what used to happen
 - submit PR

setting up docs dev environment/building docs

- [FIX] now it's time to finally go back to [the instructions to build the docs](#). it's the first thing after the intro and I understand why it's there but I don't love the placement because I didn't actually need to build anything until now, like an hour into the process
- lots of clicking around:
 - steps:
 - clicked "get Cpython repository clone", lead me to <https://devguide.python.org/getting-started/setup-building/#checkout>
 - then clicked "fork the cpython repository on github"
 - <https://devguide.python.org/getting-started/git-boot-camp/#fork-cpython>

- which finally had a link to github to click fork
 - [FIX] kind of an annoying amount of steps? could be all on one page, even if it means duplicating instructions here.
 - otoh I had no problem because I'm an experienced FOSS contributor who has forked things on github a thousand times, and maybe you don't *want* people who are unfamiliar with github?
 - wait actually the instructions *are* all under the second link?
 - idk my point is this section could maybe be simplified.
- omg this conference wifi, I have been trying to clone the cpython repo for nearly an hour 😭
- now I'm getting stuck on configuring remotes
 - git remote set-url --push upstream git@github.com:<username>/cpython.git
 - gives me the error error: No such remote 'upstream'
 - I think this command didn't work?
 - git config --local branch.main.remote upstream
 - Or maybe it did work but in a way I don't understand? I'm so confused.
 - WAIT I FIGURED IT OUT
 - **[FIX]** 'adding a remote' was under 'clone a forked repository' section instead of the 'configure remotes' section just below it, so I missed it
- I started compiling and building cpython but then stopped, I think I'm supposed to go back to the first page???
 - <https://devguide.python.org/getting-started/setup-building/#compile-and-build>
 - [FIX] I understand the point of not repeating yourself but all the clicking around really is confusing
- wow "make htmllive" is really cool
- ok I think I've built the docs and can now start making changes

actually making a change

- copying my plan from above:
 - add reference to create_autospec
 - keep the verbose description, but remove the reference to the issue with misspelled asserts
 - add a version changed for 3.5 and a little detail there about what used to happen
- **[FIX]** might benefit from a bit of instruction on how to find docs files in the repo from the url in the browser - it's pretty straightforward if you've worked with python docs before but a sentence or two could help.
- oooooooh the function 'create_autospec' exists on this page, it's just elsewhere, <https://docs.python.org/3/library/unittest.mock.html#create-autospec>
 - and for inexplicable-to-me reasons the further explanation has been deferred to somewhere else
 - so I think I'm going to leave it where it is so that the diff of my other changes shows up properly but in the PR I'll flag that it could be merged into

create_autospec; moved next to create_autospec; or moved to a separate how-to guide

- the precommit hook is causing problems for me, gonna try and uninstall it
 - uninstalled the pre-commit hook and made PR anyway
 - okay, that worked, but the linting is failing and there's no way to lint my PR directly, AFAICT - I *have* to get the pre-commit hook working, so let me try again
 - Guess I'm gonna google "RuntimeError: failed to find interpreter for Builtin discover of python_spec='python3.12'" to get my docs fixed merged 😞
 - ooohh I think it's because my default working Python version is not 3.12. it's python 3.10. did I miss being told to specifically create a 3.12 environment?
 - I did not, I was just told "make venv"
 - Melanie looked up the correct command to ensure python3.12:
make venv PYTHON=python3.12
 - [FIX] if we're going to tell people to install the pre-commit hook by default and the pre-commit hook requires python 3.12 we should probably tell people to make a venv with 3.12
 - it seems to require python3.12 but make venv didn't help me by specifying python3.12
 - "language_version: python3.12"
 - while I was debugging the pre-commit hook issue I posted the failing linting test to Discord and they helped me read the output better, the fact that the linting issue was trailing whitespace was explained a bit above where I was looking (the specific error message)
 - [FIX] In a project I co-maintain we have some screenshots showing a few common test failures in order to help people interpret them - maybe useful here? IDK, might be overexplaining
 - why is it so hard to create a virtual environment with python3.12 :(
 - maybe you can't specify the python version within a virtual env?
 - ohhhhh it was pyenv I was missing
 - oh cool now when I run the pre-commit hook I get [this error](#)
 - I'm giving up on the pre-commit hook again, but the docs should *definitely* have a bit of info like "hey if you are running into X or Y error that's an error with the pre-commit hook and you can disable it by typing "pre-commit uninstall" and just figure out the issues in the PR
 - okay I missed adding a file so I need a commit that is something like "the same as last time but actually adding the file".
 - I asked Ned what to do (squash commit and force push? make a commit with a quick name like I would with most projects? do my best to describe the silly commit I need to make?) and he said definitely **not** force push, instead do your best to sound professional as you explain you forgot to add a file, and they'll squash later, so that should be in the docs somewhere [FIX]

pre-commit error 2

```
File "/home/shauna/.pyenv/versions/3.12.3/lib/python3.12/importlib/__init__.py", line 90,  
in import_module  
    return _bootstrap._gcd_import(name[level:], package, level)  
        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^  
  
File "<frozen importlib._bootstrap>", line 1387, in _gcd_import  
File "<frozen importlib._bootstrap>", line 1360, in _find_and_load  
File "<frozen importlib._bootstrap>", line 1331, in _find_and_load_unlocked  
File "<frozen importlib._bootstrap>", line 935, in _load_unlocked  
File "<frozen importlib._bootstrap_external>", line 995, in exec_module  
File "<frozen importlib._bootstrap>", line 488, in _call_with_frames_removed  
File  
"/tmp/pip-build-env-ldtnc3_i/normal/lib/python3.12/site-packages/wheel/bdist_wheel.py", line 27,  
in <module>  
    from .macosx_libfile import calculate_macosx_platform_tag  
File  
"/tmp/pip-build-env-ldtnc3_i/normal/lib/python3.12/site-packages/wheel/macosx_libfile.py", line  
43, in <module>  
    import ctypes  
File "/home/shauna/.pyenv/versions/3.12.3/lib/python3.12/ctypes/__init__.py", line 8, in  
<module>  
    from _ctypes import Union, Structure, Array  
ModuleNotFoundError: No module named '_ctypes'  
[end of output]
```