

Journey from 2 to 3

Nejc Saje, Zemanta



TIME



to shoot in video, better weather sealing, a touchscreen, and upgraded autofocus. (There are plenty of other more detailed specs, many of which will read like alphabet soup to casual photographers.)

Bottom line is this: These are among the highest performing cameras available. They're for photographers who for shooting portraits, landscapes, news events, and fast-paced

You May Like



The Online Furniture Retailer That's Shaking Up The Industry

Macy's



Your 401K Isn't Growing As Fast As it Should Be - Here's Why

AARP



10 Cars That Will Cost You Less Than 25k

Allstate



What You Need to Tell Your Kids About Budgeting [Video]

Bank of America

Recommended by  **Outbrain**





Zemanta

Content Area Media Sources Publishers

Click

Incremental

10

20

30

40

50

60

70

80

90

100

110

120

130

140

150

160

170

180

190

200

210

220

230

240

250

260

270

280

290

300

310

320

330

340

350

360

370

380

390

400

410

420

430

440

450

460

470

480

490

500

510

520

530

540

550

560

570

580

590

600

610

620

630

640

650

660

670

680

690

700

710

720

730

740

750

760

770

780

790

800

810

820

830

840

850

860

870

880

890

900

910

920

930

940

950

960

970

980

990

1000

1010

1020

1030

1040

1050

1060

1070

1080

1090

1100

1110

1120

1130

1140

1150

1160

1170

1180

1190

1200

1210

1220

1230

1240

1250

1260

1270

1280

1290

1300

1310

1320

1330

1340

1350

1360

1370

1380

1390

1400

1410

1420

1430

1440

1450

1460

1470

1480

1490

1500

1510

1520

1530

1540

1550

1560

1570

1580

1590

1600

1610

1620

1630

1640

1650

1660

1670

1680

1690

1700

1710

1720

1730

1740

1750

1760

1770

1780

1790

1800

1810

1820

1830

1840

1850

1860

1870

1880

1890

1900

1910

1920

1930

1940

1950

1960

1970

1980

1990

2000

2010

2020

2030

2040

2050

2060

2070

2080

2090

2100

2110

2120

2130

2140

2150

2160

2170

2180

2190

2200

2210

2220

2230

2240

2250

2260

2270

2280

2290

2300

2310

2320

2330

2340

2350

2360

2370

2380

2390

2400

2410

2420

2430

2440

2450

2460

2470

2480

2490

2500

2510

2520

2530

2540

2550

2560

2570

2580

2590

2600

2610

2620

2630

2640

2650

2660

2670

2680

2690

2700

2710

2720

2730

2740

2750

2760

2770

2780

2790

2800

2810

2820

2830

2840

2850

2860

2870

2880

2890

2900

2910

2920

2930

2940

2950

2960

2970

2980

2990

3000

3010

3020

3030

3040

3050

3060

3070

3080

3090

3100

3110

3120

3130

3140

3150

Zemanta dashboard backend

- Django
- Initial commit June 2014
- 3k pull requests
- 16k commits
- 110k Python LOC
- 2,2k unit tests
- 5 active developers

Why upgrade?

- Advanced unpacking
- Keyword only arguments
- Chained exceptions
- Iterators by default
- Asyncio
- Type hinting

```
>>> a, b, *rest = range(10)
```

```
def f(a, b, *args, option=True):  
    ...
```

Why **we** upgraded?

- Advanced unpacking
- Keyword only arguments
- Chained exceptions
- Iterators by default
- Asyncio
- Type hinting

```
>>> a, b, *rest = range(10)
```

```
def f(a, b, *args, option=True):  
    ...
```

Why **we** upgraded?

- Type hinting
- ... and Django 2.0 🐍

Type hinting

```
1  from typing import NewType
2
3  PersonName = NewType('PersonName', str)
4
5
6  def get_person_name(name: str) -> PersonName:
7      return PersonName(name)
8
9
10 class Dog:
11     name: str
12     age: int
13
14     def __init__(self, name: str, age: int):
15         self.name = name
16         self.age = age
17
18     def greet(self, who: PersonName):
19         print("Hi %s, I'm %s and I'm %s years old" % (who, self.name, self.age))
20
21
22 if __name__ == '__main__':
23     dog0 = Dog('Fluffy', 'two years')
24     dog1 = Dog('Fluffy', 2)
25     dog1.greet('Nejc')
26     dog1.greet(get_person_name('Nejc'))
27
28
```

Which approach?

- Python 2 & Python 3
 - libmodernize
 - libfuturize
- Python 3 only
 - official 2to3 tool

2to3

- Automated Python 2 to 3 code translation

```
> 2to3 --write --nobackups
```

Manual fixes needed

- Dictionary ordering was non-deterministic since 3.3 to 3.5, **reverted in 3.6**
 - Problematic if tests compare unsorted JSON - it changed every time
- Bytes and unicode not interchangeable anymore
 - Some libraries are expecting bytes instead of strings in certain places
 - Some files should be treated as bytes now, especially images
 - `HttpResponse.content` is now bytes
- `assertItemsEqual` removed
 - Replaced by `assertCountEqual`

Manual fixes needed

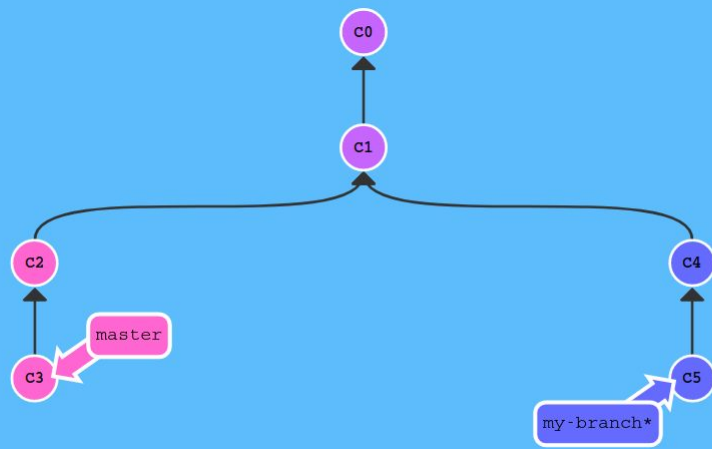
- No `message` attribute on `Exception` class anymore
 - Should use `str(exc)` instead
- Stricter comparisons
 - no more `max(1.2, None)` - results in `TypeError`
- `exceptions` module no longer exists
 - moved to builtin
- Import changes
 - Implicit relative imports not allowed anymore
 - If you have modules *a* and *b* inside *mypkg*, *a* cannot do `from b import BClass`.
 - either `from mypkg.b import BClass`
 - or `from .b import BClass`

What needs to be done then?

1. Automated conversion
2. Manual fixes

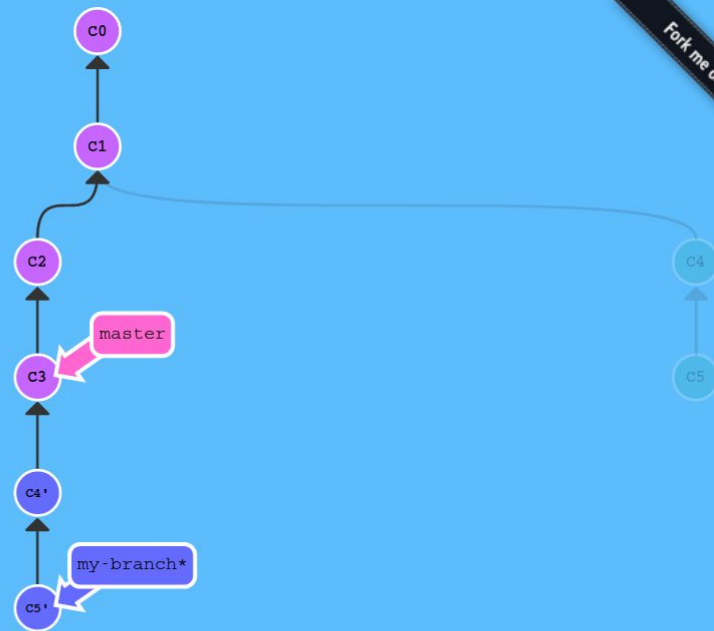
Stop the world?

```
Learn Git Branching
$ git branch my-branch
$ git commit
$ git commit
$ git checkout my-branch
$ git commit
$ git commit
```



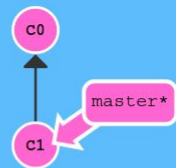
Fork me on GitHub


```
Learn Git Branching
git branch my-branch
git commit
git commit
git checkout my-branch
git commit
git commit
git rebase master
```



Fork me on GitHub

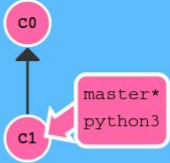
Our migration workflow



Fork me on GitHub

Learn Git Branching

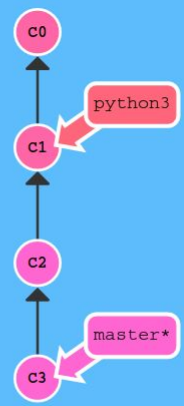
```
$ git branch python3
```



Fork me on GitHub

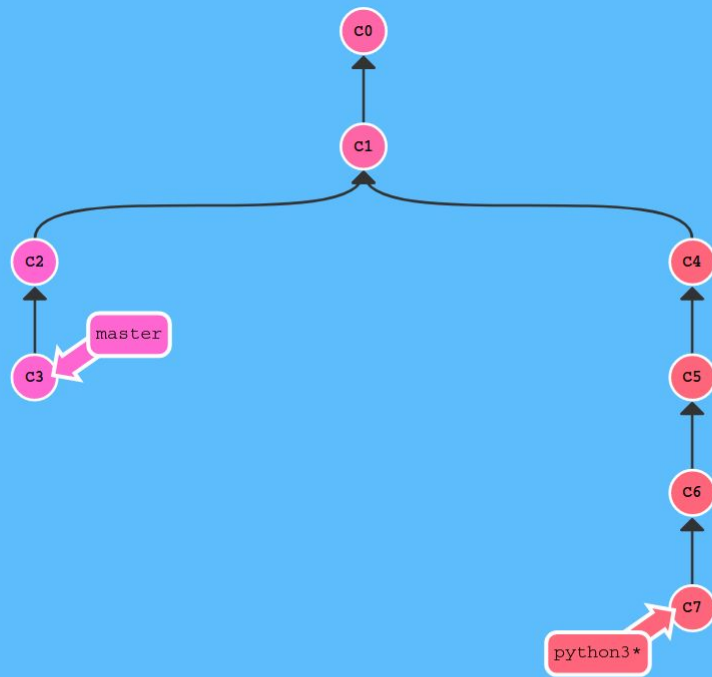
Learn Git Branching

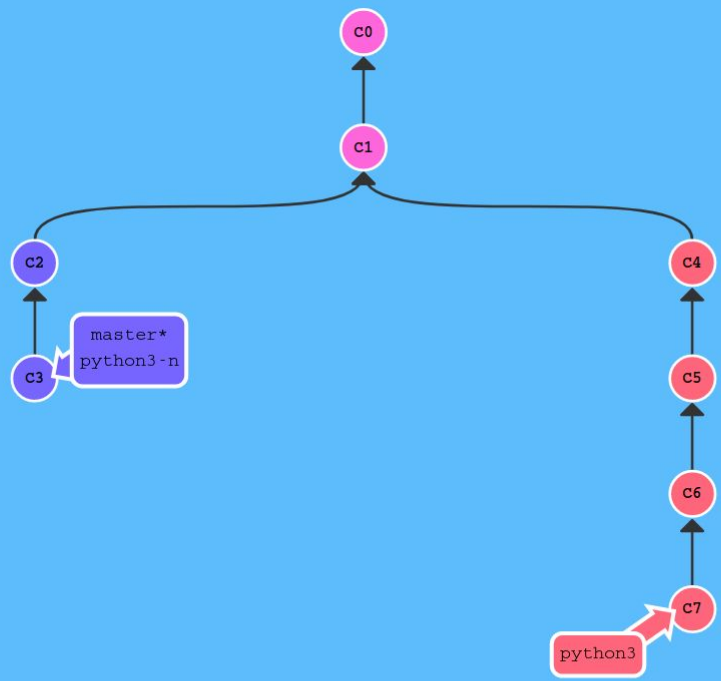
```
$ git branch python3  
$ git commit  
$ git commit
```



Fork me on GitHub

```
git branch python3
git commit
git commit
git checkout python3
git commit
git commit
git commit
git commit
```





Learn Git Branching

```
git branch python3
git commit
git commit
git checkout python3
git commit
git commit
git commit
git commit
git checkout master
git branch python3-new
```

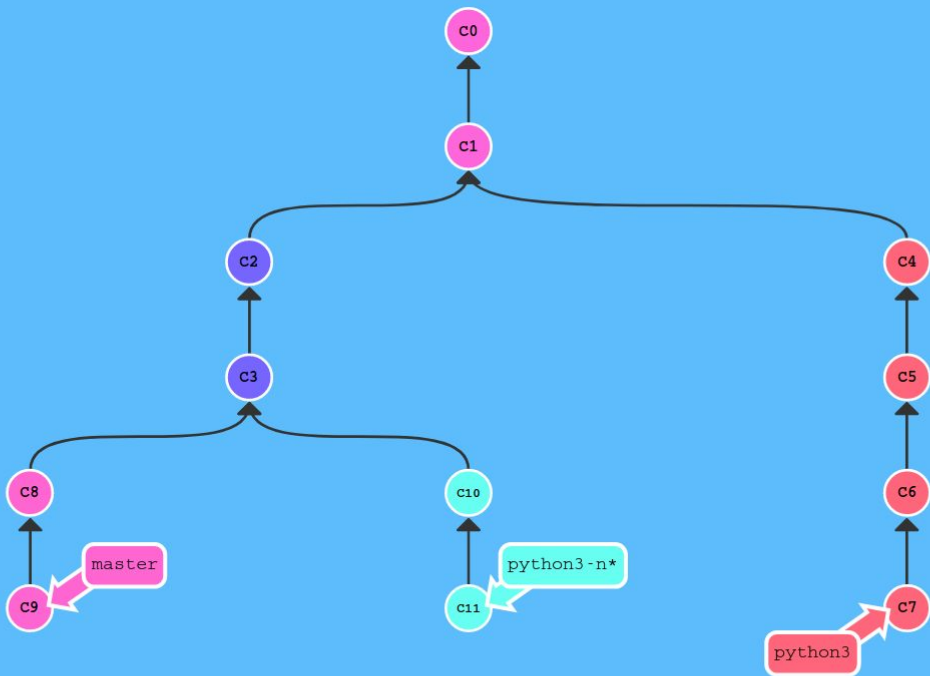
⚠ Sorry, we need to keep branch names short for the visuals. Your branch name was truncated to 9 characters, resulting in "python3-n"

Learn Git Branching

```
$ git branch python3
$ git commit
$ git commit
$ git checkout python3
$ git commit
$ git commit
$ git commit
$ git commit
$ git checkout master
$ git branch python3-new
```

⚠ Sorry, we need to keep branch names short for the visuals. Your branch name was truncated to 9 characters, resulting in "python3-n"

```
$ git commit
$ git commit
$ git checkout python3-n
$ git commit
$ git commit
```

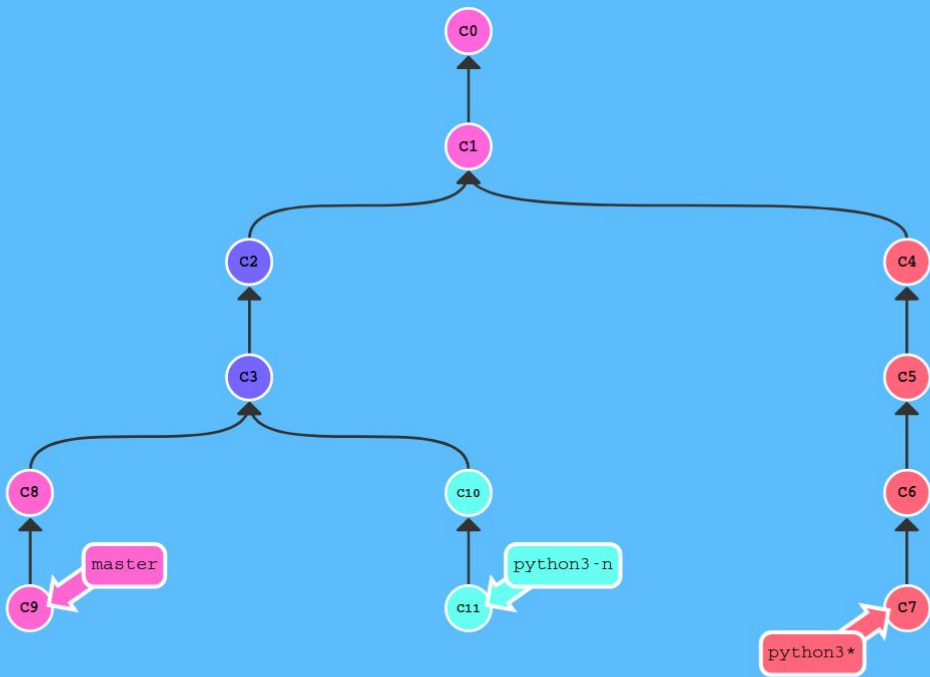


Learn Git Branching

```
git branch python3
git commit
git commit
git checkout python3
git commit
git commit
git commit
git commit
git checkout master
git branch python3-new
```

⚠ Sorry, we need to keep branch names short for the visuals. Your branch name was truncated to 9 characters, resulting in "python3-n"

```
git commit
git commit
git checkout python3-n
git commit
git commit
git checkout python3
```



```
git branch python3
git commit
git commit
git checkout python3
git commit
git commit
git commit
git checkout master
git branch python3-new
```

❗ Sorry, we need to keep branch
short for the visuals. Your branch
name was truncated to 9 characters
resulting in "python3-n"

```
git commit
git commit
git checkout python3-n
git commit
git commit
git checkout python3
git rebase -i python3-n
```

Interactive Rebase

Rebasing 4 Commits

(Drag and drop to re-order. Toggle the "pick" button to omit or re-add a commit)

C4

Pick

C5

Pick

C6

Pick

C7

Pick

Cancel

Confirm

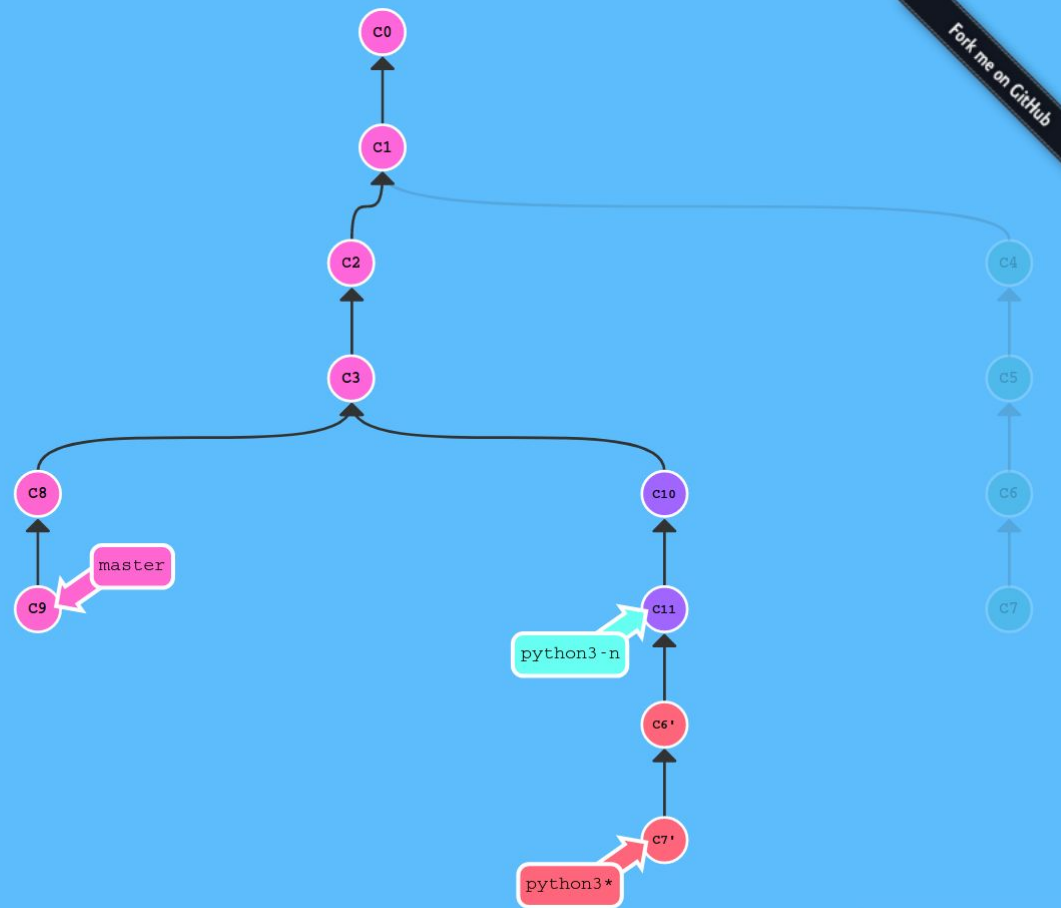
C4

C5

C6

C7

```
Learn Git Branching
git branch python3
git commit
git commit
git checkout python3
git commit
git commit
git commit
git commit
git checkout master
git branch python3-new
! Sorry, we need to keep branch names
short for the visuals. Your branch
name was truncated to 9 characters,
resulting in "python3-n"
git commit
git commit
git checkout python3-n
git commit
git commit
git checkout python3
git rebase -i python3-n
```



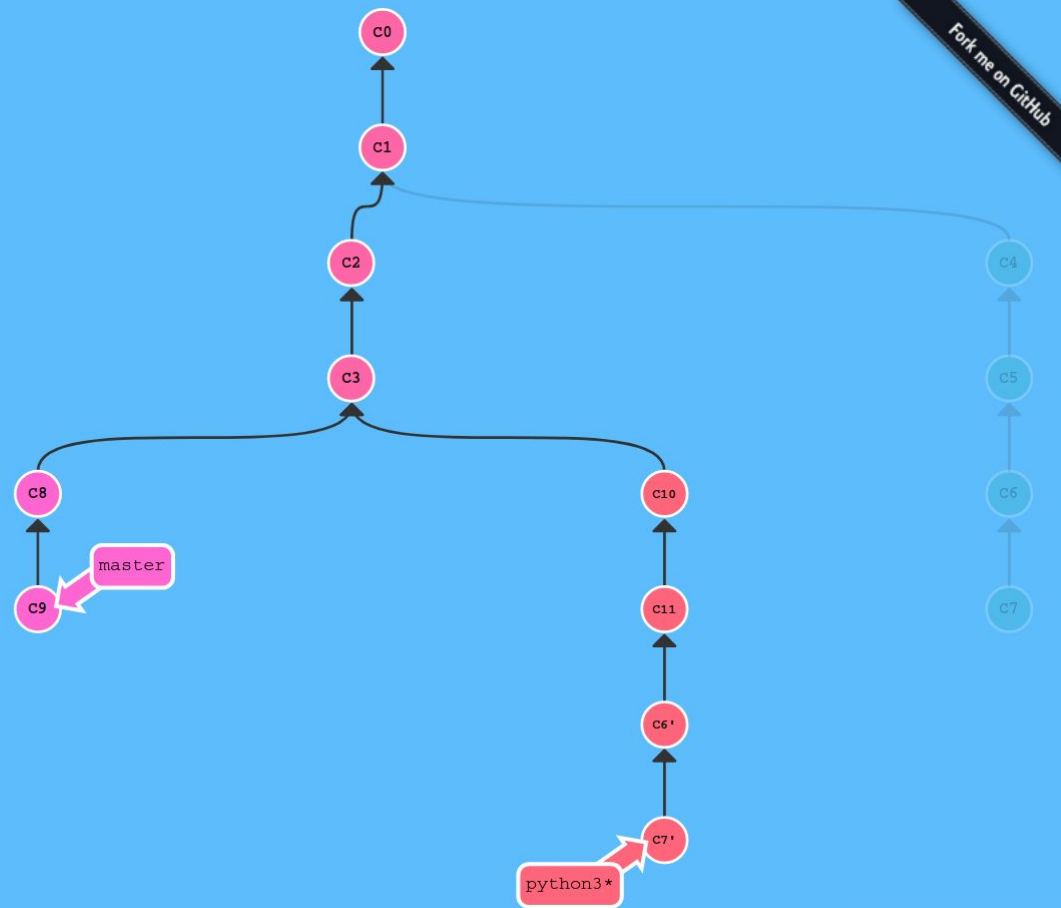
Fork me on GitHub

Learn Git Branching

```
git branch python3
git commit
git commit
git checkout python3
git commit
git commit
git commit
git commit
git checkout master
git branch python3-new
```

⚠ Sorry, we need to keep branch names short for the visuals. Your branch name was truncated to 9 characters, resulting in "python3-n"

```
git commit
git commit
git checkout python3-n
git commit
git commit
git checkout python3
git rebase -i python3-n
git branch -D python3-n
```



Finally...

```
Ran 1983 tests in 174.237s
```

```
FAILED (failures=101, errors=336, skipped=1)
```

```
Ran 2038 tests in 154.589s
```

```
FAILED (failures=4, skipped=1)
```

```
Ran 2038 tests in 152.821s
```

```
OK (skipped=1)
```

Weak spots

- reading/writing files
- encryption/decryption
- calling external services
- credentials

Questions?

Thank you!