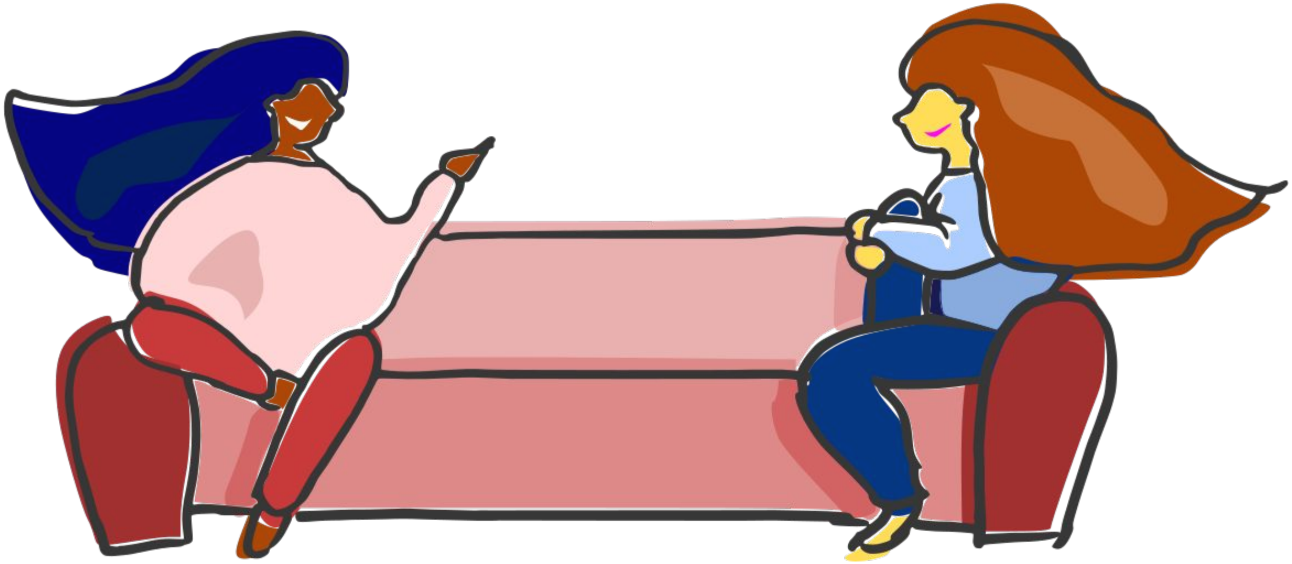
pyladies PARIS

Python 3.9: an incremental release or a breakthrough?

led by Maria Teleńczuk, PhD
Maud Levy



22th June 2021
Online, 7:30 pm

Introduction

Find us:

meetup.com/pyladiesparis

slackin.pyladies.com > #city-paris

https://twitter.com/pyladiesparis

https://github.com/PyLadiesParis

Let’s get to know each other!

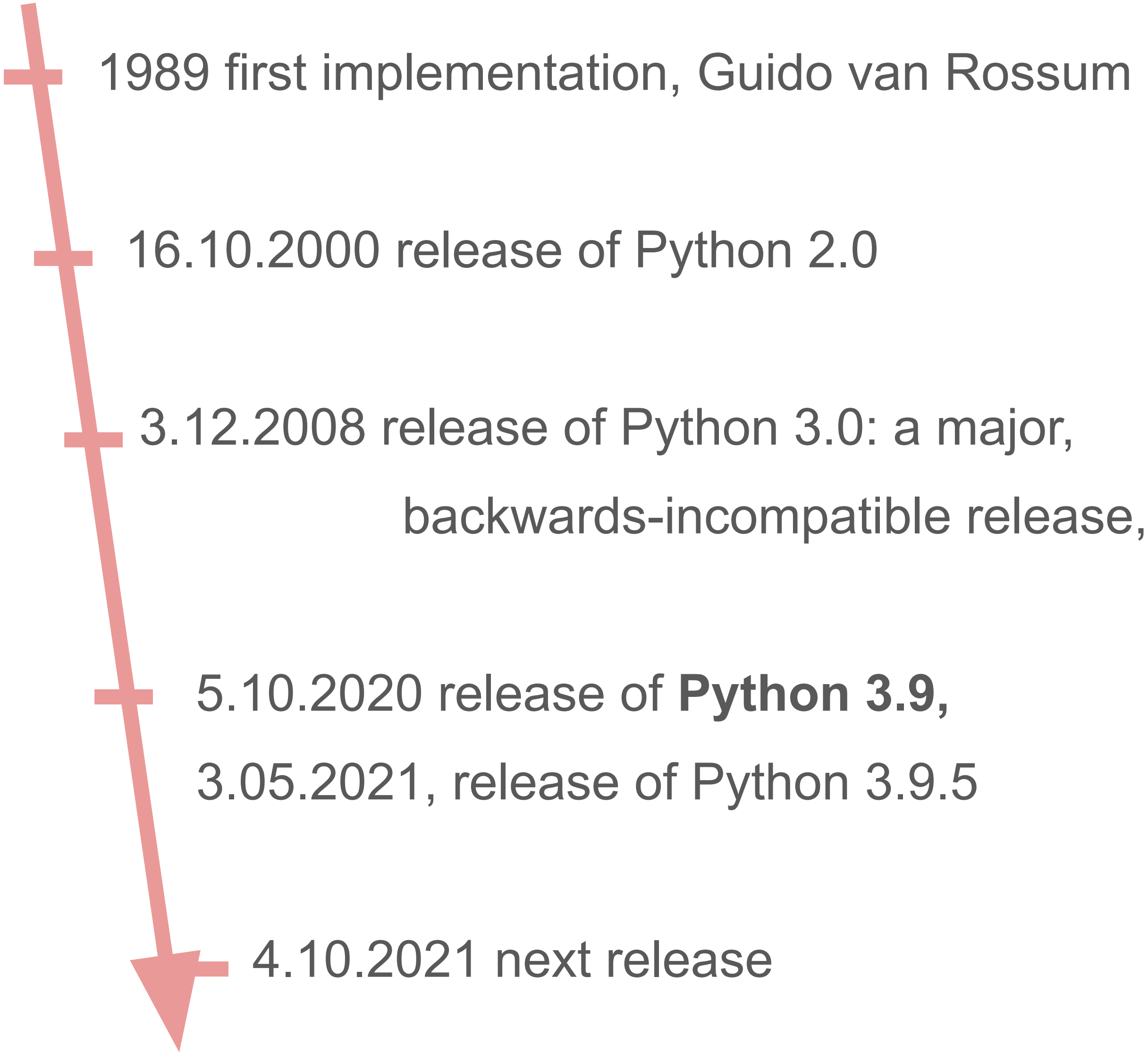
Tell us about yourself

What’s your experience with Python?

What are you looking for in PyLadies?

What languages do you speak?

Quick history of Python



1989 first implementation, Guido van Rossum

16.10.2000 release of Python 2.0

3.12.2008 release of Python 3.0: a major, backwards-incompatible release,

5.10.2020 release of **Python 3.9,**

3.05.2021, release of Python 3.9.5

4.10.2021 next release

Python 3.9, what’s new?

1. Yearly release cycle
2. Faster
3. New Features for strings, and
4. New Features for dictionaries
5. Decorators loose restrictions
6. New python type operations
7. And more...

1. Yearly release cycle

- Before: 18 months cycle
- Now: 12 months
    - Less features per release
    - Faster feedback

2. Faster

Two main performance improvements:

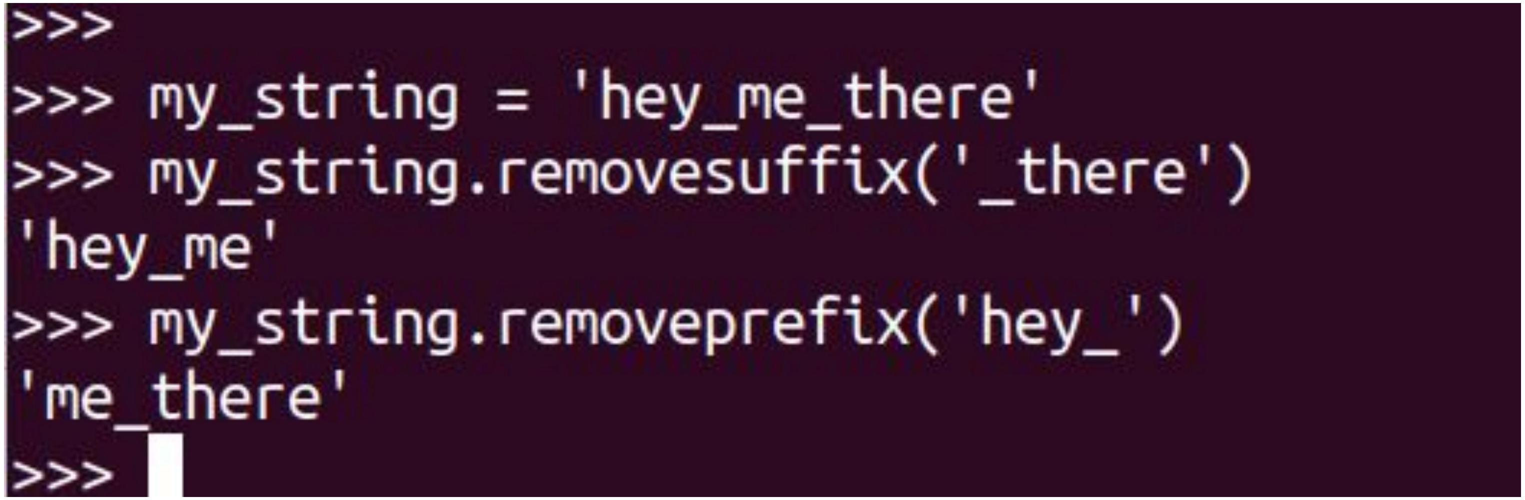
1. More use of vectorcall protocol (introduced in Python 3.8)

2. More efficient parsing of Python source code

3. New Features for strings

New methods to remove prefixes and suffixes:

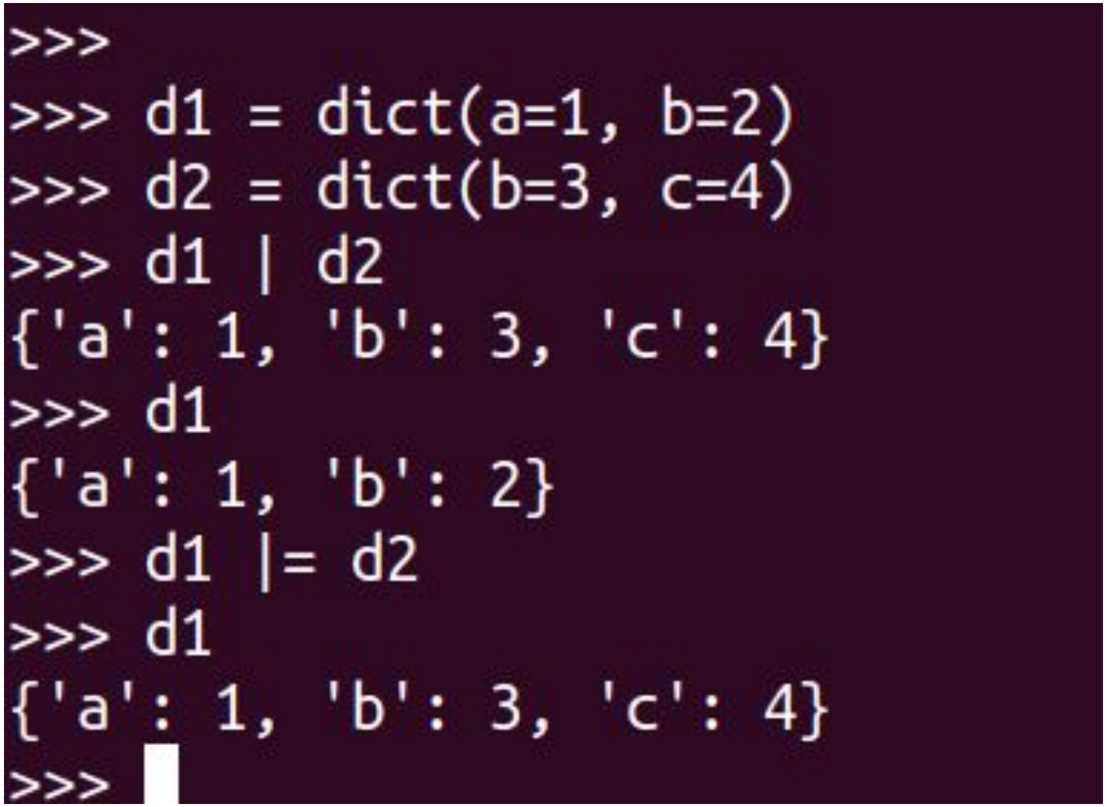
- removeprefix()
- removesuffix()



```
>>>
>>> my_string = 'hey_me_there'
>>> my_string.removesuffix('_there')
'hey_me'
>>> my_string.removeprefix('hey_')
'me_there'
>>>
```

4. New Features for dictionaries

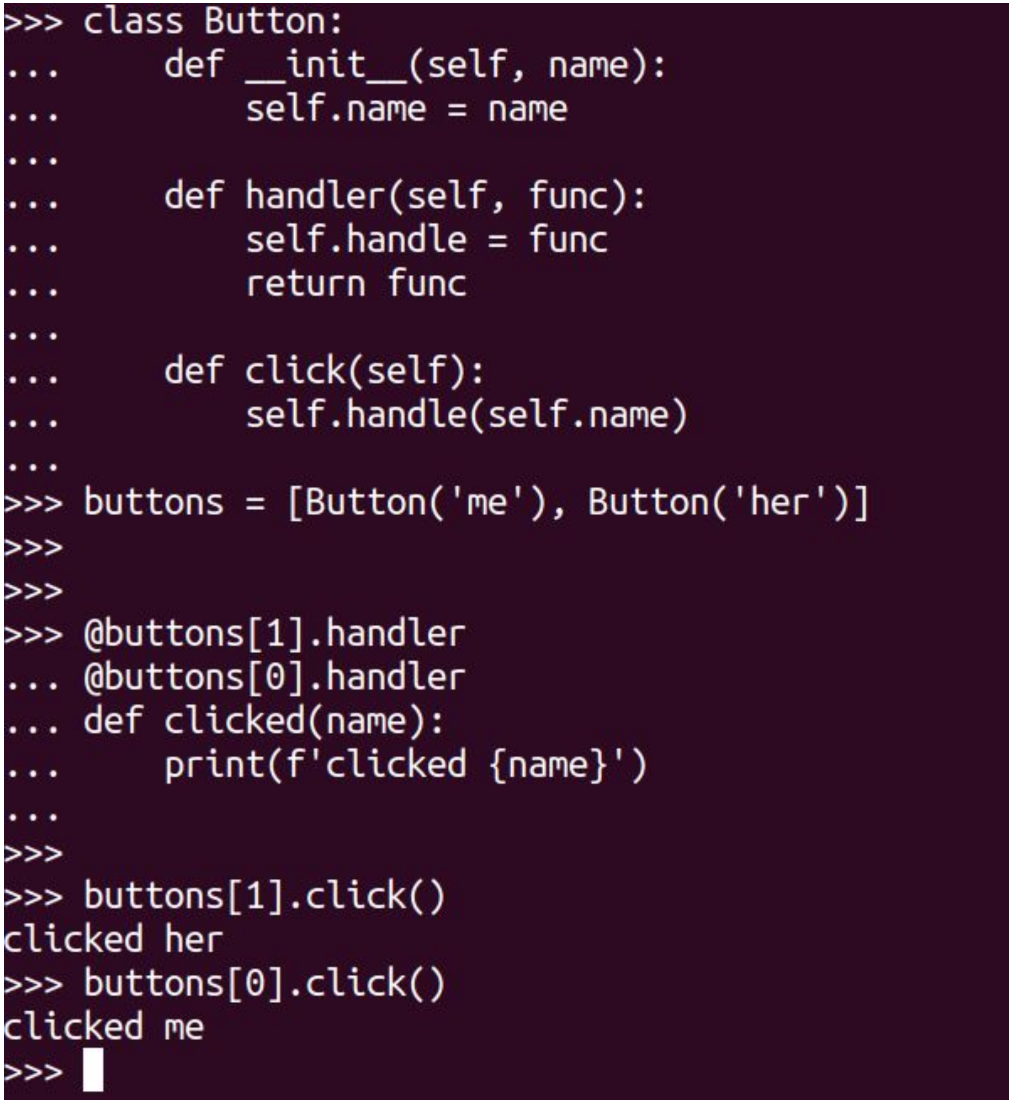
- Union operator |
- Merge operator |=



```
>>>
>>> d1 = dict(a=1, b=2)
>>> d2 = dict(b=3, c=4)
>>> d1 | d2
{'a': 1, 'b': 3, 'c': 4}
>>> d1
{'a': 1, 'b': 2}
>>> d1 |= d2
>>> d1
{'a': 1, 'b': 3, 'c': 4}
>>>
```

5. Decorators lose restrictions

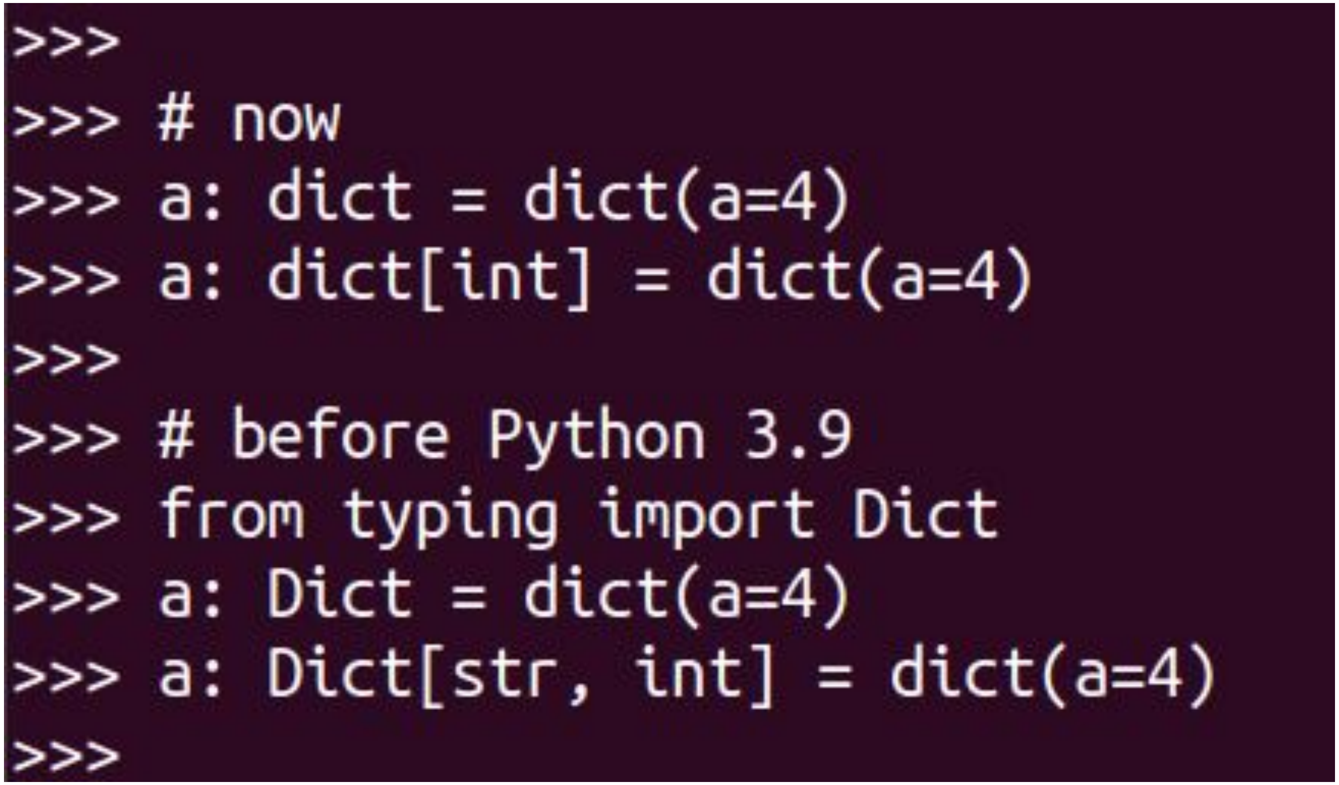
Decorators can now consist of any valid expression



```
>>> class Button:
...     def __init__(self, name):
...         self.name = name
...
...     def handler(self, func):
...         self.handle = func
...         return func
...
...     def click(self):
...         self.handle(self.name)
...
>>> buttons = [Button('me'), Button('her')]
>>>
>>>
>>> @buttons[1].handler
... @buttons[0].handler
... def clicked(name):
...     print(f'clicked {name}')
...
>>>
>>> buttons[1].click()
clicked her
>>> buttons[0].click()
clicked me
>>> █
```

6. New python type operations

Expanded support for type hinting



```
>>>
>>> # now
>>> a: dict = dict(a=4)
>>> a: dict[int] = dict(a=4)
>>>
>>> # before Python 3.9
>>> from typing import Dict
>>> a: Dict = dict(a=4)
>>> a: Dict[str, int] = dict(a=4)
>>>
```

7. Other features you know about/ like?

- Do you know other Python 3.9 features?
- Do you think Python 3.9 is a breakthrough?
- Is there a change in Python which you would like to have and you would consider a breakthrough?

Future of Relaxed Events

- Are there any topics which you would like to discuss?
- Would you like to lead a discussion sometime in the future?
- Would you like us to discuss other Python related topics at the end?
- Or play some games?

Future of Pyladies Paris

1. What kind of events mostly interest you? Face-to-face meetings? What kind?

2. Would you like to be a speaker of a talk in the future?

- Talk proposal form: https://kutt.it/ppspeaking
- Relaxed discussions suggestions: https://kutt.it/ppdiscussions

3. What can we improve?

   Share your feedback: https://kutt.it/ppfeedback

Thanks and hope to see you in 2 weeks