Python's Guide to the Galaxy

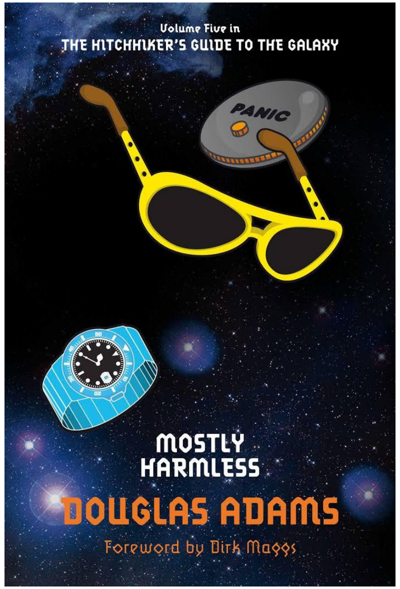
Tom Ron

Swiss Python Summit
February 2016

Tom Ron

- Senior Data Scientist @ Magic Internet
- Geek
- Python Developer
- Mostly Harmless

https://github.com/tomron/python_swiss_2016



Agenda - trilogy in 4 parts

- Data Structures -collections, itertools
- Dates - time, datetime
- Text - string, unicode, re
- And more

Data Structures

Collections

namedtuple()	factory function for creating tuple subclasses with named fields	*New in version 2.6.*
deque	list-like container with fast appends and pops on either end	*New in version 2.4.*
Counter	dict subclass for counting hashable objects	*New in version 2.7.*
OrderedDict	dict subclass that remembers the order entries were added	*New in version 2.7.*
defaultdict	dict subclass that calls a factory function to supply missing values	*New in version 2.5.*

collections

```python
d = {}
d[42] += 1
```

```python
from collections import defaultdict

d = defaultdict(int)
d[42] += 1
```

```python
from collections import Counter

d = Counter()
d[42] += 1
```

```
KeyError: 42
```

```
defaultdict(<type 'int'>, {42: 1})
```

```
Counter({42: 1})
```

collections

```python
d = {1 : 20}
e = {1 : 22}
d + e
```

```
TypeError: unsupported
operand type(s) for +: 'dict'
and 'dict'
```

```python
from collections import
Counter

d = Counter({1 : 20})
e = Counter({1 : 22})
d + e
```

```
Counter({1: 42})
```

iterating

```
books = ["The Hitchhiker's Guide to the Galaxy',
"The Restaurant at the End of the Universe',
"Life, the Universe and Everything',
"So Long, and Thanks for All the Fish',
"Mostly Harmless", "And Another Thing..."]

for index, book in enumerate(books, 1):
    print "\"%s\" is the %s book"%(book, index)
```

```
"The Hitchhiker's Guide to the Galaxy" is the 1 book

"The Restaurant at the End of the Universe" is the 2 book

"Life, the Universe and Everything" is the 3 book
```

iterating

```
publish_years = [1979, 1980, 1982, 1984, 1992, 2009]
```

```
for book, year in zip(books, publish_years):
    print "%s was published in %s"%(book, year)
```

```
The Hitchhiker's Guide to the Galaxy was published in 1979

The Restaurant at the End of the Universe was published in 1980

Life, the Universe and Everything was published in 1982
```

itertools

Infinite iterators	count, cycle, repeat
Iterators terminating on the shortest input sequence	chain, compress, dropwhile, groupby, ifilter, ifilterfalse, islice, imap, startmap, tee, takewhile, izip, iziplongest
Combinatoric generators	product, permutations, combinations, combinations_with_replacement

itertools

```
from itertools import takewhile
books_publish_year = zip(books, publish_years)

# All books published before 1990
# Assuming books are sorted

books_before_1990 = takewhile(lambda (book, year): year <
1990, books_publish_year)
```

```
[The Hitchhiker's Guide to the Galaxy, The Restaurant at the End of the Universe,
Life, the Universe and Everything, So Long, and Thanks for All the Fish]
```

itertools

```python
# Taking 2 books for to read on my vacation

from itertools import combinations

for book1, book2 in combinations(books, 2):
    print "\"%s\"\t\"%s\""%(book1, book2)
```

```
"The Hitchhiker's Guide to the Galaxy"	"The Restaurant at the End of the Universe"
"The Hitchhiker's Guide to the Galaxy"	"Life, the Universe and Everything"
"The Hitchhiker's Guide to the Galaxy"	"So Long, and Thanks for All the Fish"
"The Hitchhiker's Guide to the Galaxy"	"Mostly Harmless"
"The Hitchhiker's Guide to the Galaxy"	"And Another Thing..."
"The Restaurant at the End of the Universe""Life, the Universe and Everything"
...
```

itertools

```
# But which one should I read first?

from itertools import permutations

for book1, book2 in permutations(books, 2):
    print "\"%s\"\t\"%s\""%(book1, book2)
```

itertools

```
# group by - books by decades

from itertools import groupby

for decade, gr in groupby(books_publish_year, lambda x:
10*(x[1]/10)):
    print decade, ";".join(["\"%s\""%(g[0]) for g in gr])
```

```
1970 "The Hitchhiker's Guide to the Galaxy"
1980 "The Restaurant at the End of the Universe";"Life, the Universe and
Everything";"So Long, and Thanks for All the Fish"
1990 "Mostly Harmless"
2000 "And Another Thing..."
```

Dates

`time` - Time access and conversions

`datetime` - Basic date and time types, dates manipulations

`calendar` — General calendar-related functions

Datetime

```python
from datetime import datetime
```

```python
# from string
my_time = '2016-02-05 09:37:11'
d = datetime.strptime(my_time, "%Y-%m-%d %H:%M:%S")
```

```
datetime.datetime(2016, 2, 5, 9, 37, 11)
```

```python
# to string
d.strftime("%Y-%B-%d %H:%M:%S")
```

```
2016-February-05 09:37:11
```

Datetime

```python
from datetime import timedelta

delta = timedelta(hours=1)
time_in_1_hour = now + delta


print now
```

```
2016-01-31 17:07:03.080847
```

```python
print time_in_1_hour
```

```
2016-01-31 18:07:03.080847
```

Datetime

```
and_now = datetime.now()

# who much time passed?
time_diff = and_now - now


print "time_diff: %s"%time_diff
```

```
time_diff: 0:00:00.000088
```

```
print "time_diff.seconds: %s"%time_diff.seconds
```

```
time_diff.seconds: 0
```

```
print "time_diff.total_seconds: %s'%time_diff.total_seconds()
```

```
time_diff.total_seconds: 8.8e-05
```

Datetime

```
tomorrow = now + timedelta(days=1)
time_diff_tomorrow = tomorrow - now
```

```
print "time_diff_tomorrow: %s"%time_diff_tomorrow
```

```
time_diff_tomorrow: 1 day, 0:00:00
```

```
print "time_diff_tomorrow.seconds: %s'%time_diff_tomorrow.seconds
```

```
time_diff_tomorrow.seconds: 0
```

```
print "time_diff_tomorrow.total_seconds: %s'%time_diff_tomorrow.
total_seconds()
```

```
time_diff_tomorrow.total_seconds: 86400.0
```

Text

```
print 'zürich'
```

```
SyntaxError: Non-ASCII
character '\xc3'
```

```
# -*- coding: utf-8 -*-

print 'zürich'
```

```
zürich
```

Text

- string - plain sequence of bytes, default ASCII
- unicode - encoded , str := unicode in Python 3

Text

```
# -*- coding: utf-8 -*-
```

```
len('ü')                                  2
```

```
len(u'ü')                                 1
```

```
len(u'ü'.encode('utf-8'))                 2
```

```
len(u'ü'.encode('latin1'))                1
```

RE

```
import re

sentence = "\"The Hitchhiker's Guide to the Galaxy\" was published in 1979"

regex = "\"([\w ']+)\" was published in (\S+)"


re.findall(regex, sentence)
```

```
[("The Hitchhiker's Guide to the Galaxy", '1979')]
```

RE

```
match1 = re.match(regex, sentence)
```

`match1.groups()`	("The Hitchhiker's Guide to the Galaxy", '1979')
`match1.group(1)`	The Hitchhiker's Guide to the Galaxy
`match1.span(1)`	(1, 37)
`match1.groupdict()`	{}

RE

```
match2 = re.search("\"(?P<book>[\w ']+)\" was published in (?
P<year>\S+)", sentence)
```

```
match2.groups()
```

```
("The Hitchhiker's Guide to the Galaxy", '1979')
```

```
match2.group(1)
```

```
The Hitchhiker's Guide to the Galaxy
```

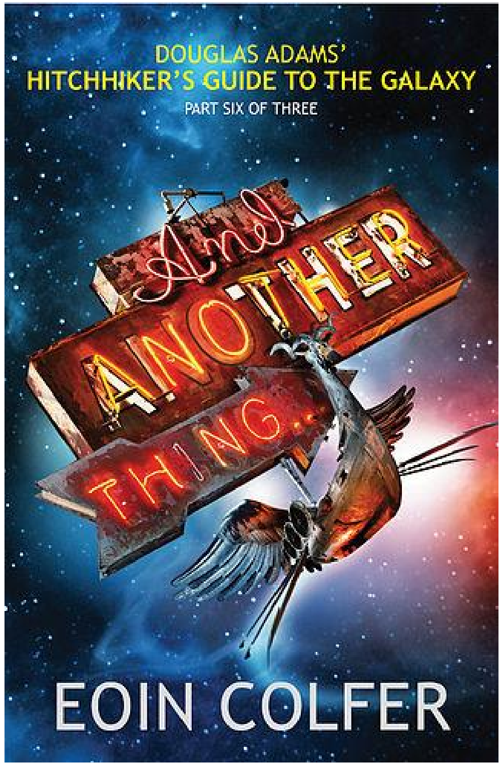
```
match2.span(1)
```

```
(1, 37)
```

```
match2.groupdict()
```

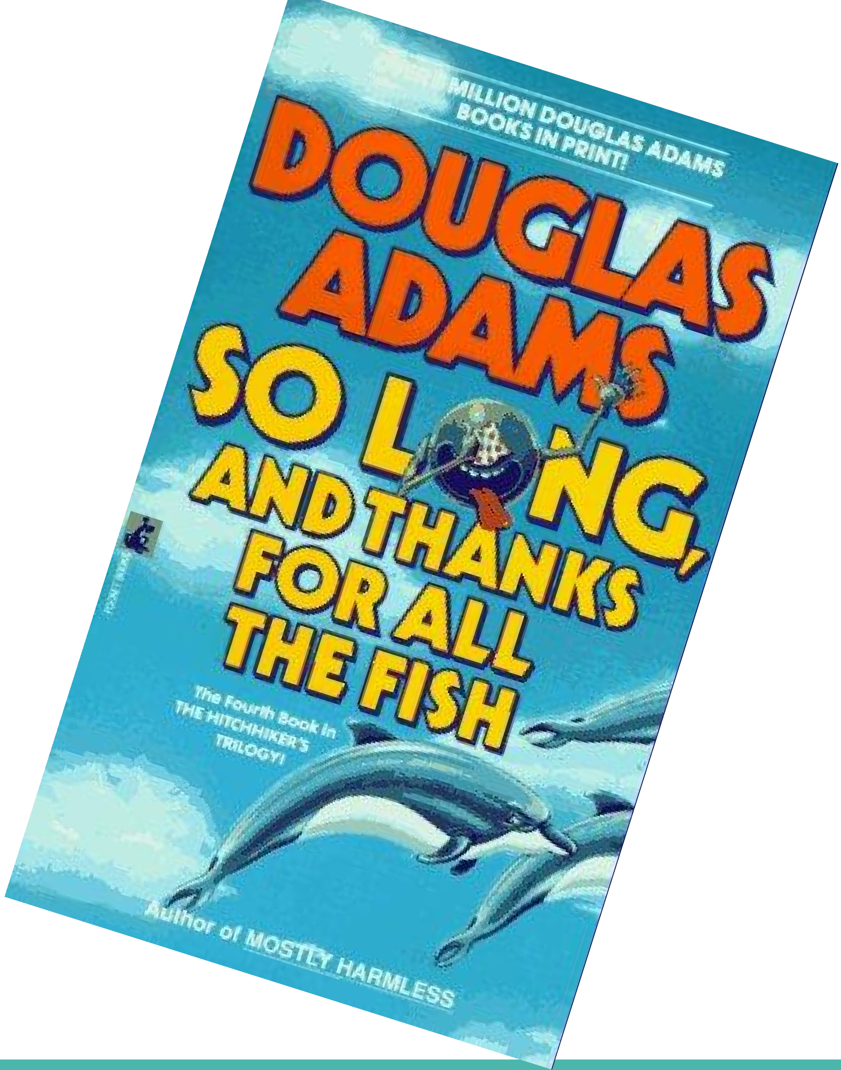
```
{'book': "The Hitchhiker's Guide to the Galaxy", 'year':
'1979'}
```

And..



- Reading data from web (urllib, urllib2)
- Async
- Profiling
- More about text

So long, as Thanks for All the Fish


MILLION DOUGLAS ADAMS BOOKS IN PRINT!
DOUGLAS ADAMS
SO LONG, AND THANKS FOR ALL THE FISH
The Fourth Book in THE HITCHHIKER'S TRILOGY!
Author of MOSTLY HARMLESS