LAST LESSON

ERRORS

```
>>> selection = input('Choose an option from (1, 2, 3): ')
Choose an option from (1, 2, 3): 1
>>> index = selection - 1
```

```
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: unsupported operand type(s) for -: 'str' and 'int'
```

Tutorial errors

TRY-EXCEPT

```
try:
    surveilled code
except:
    handled exception
```

```
selection = input('Choose an option from (1, 2, 3): ')
try:
    index = selection -1
except TypeError:
    print('You tried to count with wrong types!')
```

TRY-EXCEPT

```
selection = input('Choose an option from (1, 2, 3): ')
try:
        index = selection -1
except TypeError as exc:
        print('You tried to count with wrong types! Exception:',
```

```
You tried to count with wrong types! Exception: unsupported opera
```

TRY-EXCEPT MORE EXCEPTIONS

```
try:
    index = selection -1
except TypeError:
    print('You tried to count with wrong types!')
except NameError:
    print('Some variable must be missing'
```

TRY-EXCEPT MORE EXCEPTIONS 2

```
try:
    index = selection -1
except (TypeError, NameError):
    print('Something went wrong with the computation')
```

TRY-EXCEPT-ELSE

```
try:
    index = selection -1
except TypeError:
    print('You tried to count with wrong types!')
else:
    print('Everything went as expected')
```

TRY-EXCEPT-ELSE-FINALLY

```
try:
    index = selection -1
except TypeError:
    print('You tried to count with wrong types!')
else:
    print('Everything went as expected')
finally:
    print('This is performed no matter what')
```

```
try:
    index = selection -1
finally:
    print('This is performed no matter what')
```

TRY-EXCEPT SUMMARY

```
try except
try except except ...
try except else
try finally
try except else finally
```

EXCEPTIONS

Built-in Exceptions

```
BaseException
 +-- SystemExit
 +-- KeyboardInterrupt
 +-- GeneratorExit
 +-- Exception
      +-- StopIteration
      +-- StopAsyncIteration
      +-- ArithmeticError
      |    +-- FloatingPointError
      |    +-- OverflowError
      |    +-- ZeroDivisionError
      +-- AssertionError
      +-- AttributeError
      +-- BufferError
      +-- EOFError
      +-- ImportError
```

RAISING AN EXCEPTION

```
def get_domain_from_email(email):
    if '@' not in email:
        raise ValueError(f'Email \'{email}\' does not contain @.'
    ...
```

```
>>> get_domain_from_email('text')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<stdin>", line 3, in get_domain_from_email
ValueError: Email 'text' does not contain @.
```

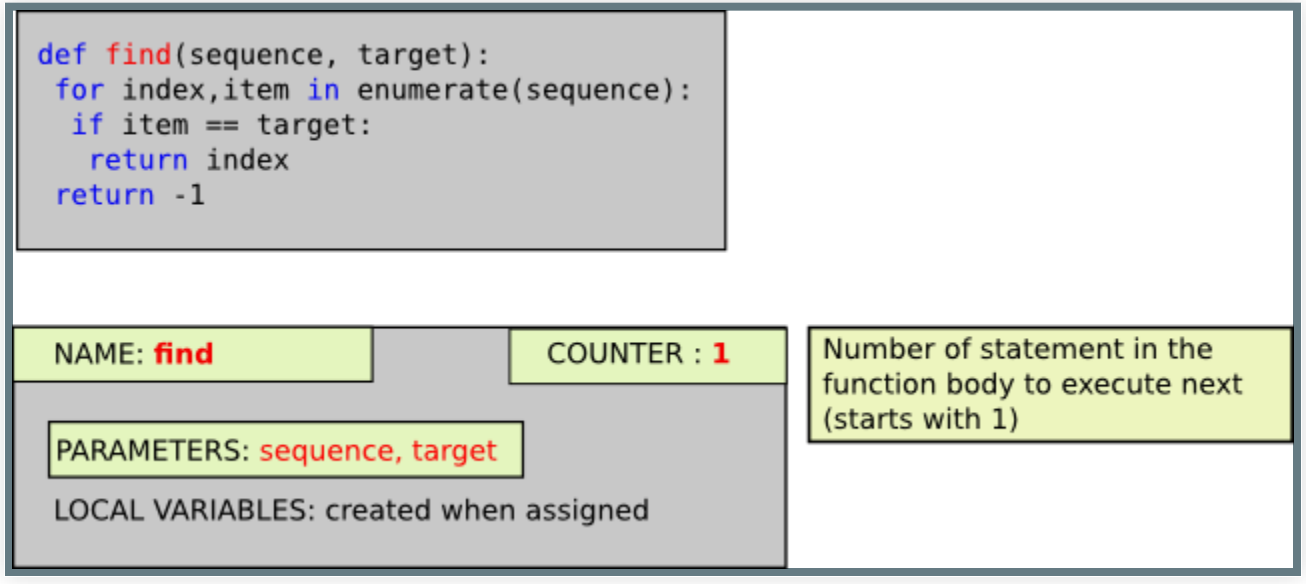
```
>>> try:
...     get_domain_from_email('text')
... except ValueError as exc:
...     print('Oops! Something went wrong. Exception:', exc)
...
Oops! Something went wrong. Exception: Email text does not contai
```

EASIER ASK FOR FORGIVENESS THAN PERMISSION

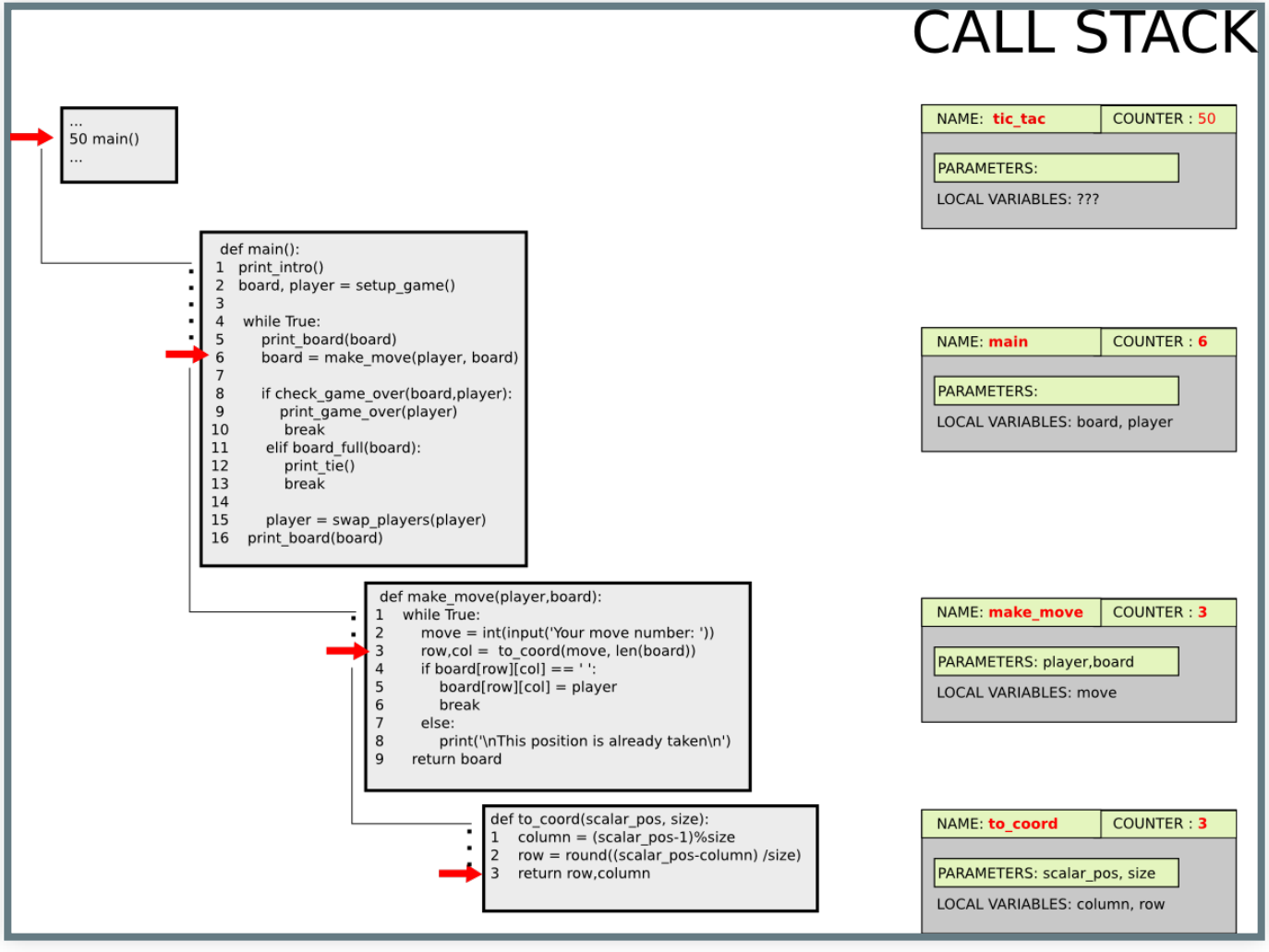
```
d = {...}
if key in d:
    return d[key] ** 2
else:
    return None
```

```
d = {...}
try:
    return d[key] ** 2
except KeyError:
    return None
```

DEBUGGING - FRAME



DEBUGGING - CALL STACK



TODAY'S LESSON

import

```
>>> import random
>>> random.random()
0.16541488247158354
>>> from random import randint
>>> randint(1, 10)
6
>>> from random import randint as my_randint
>>> my_randint(1, 10)
3
>>> from random import * # not recommended
>>> choice([1, 2, 3])
2
```

PYTHON STANDARD LIBRARY

Batteries included

Standard library

os

```
>>> import os
>>> os.listdir()
['prezentace']
>>> os.getcwd()
'/home/psebek/projects/engeto/10'
>>> os.mkdir('test_dir')
>>> os.listdir()
['test_dir', 'prezentace']
```

os.path

```
>>> import os.path  # or just os
>>> os.path.exists('test_dir')
True
>>> os.path.isfile('test_dir')
False
>>> os.path.isdir('test_dir')
True
>>> path = os.path.join('test_dir', 'sub_folder', 'file.txt')
>>> path
'test_dir/sub_folder/file.txt'
>>> os.path.abspath(path)
'/home/psebek/projects/engeto/10/test_dir/sub_folder/file.txt'
>>> os.path.basename(path)
'file.txt'
```

THIRD-PARTY PACKAGES

PyPI

VIRTUAL ENVIRONMENT

Solves package versions conflicts

Tutorial venv

```
$ python3 -m venv .env
$ ls
.env
$ source .env/bin/activate
(.env) $ python --version
3.7.5
```

pip

Installs third-party packages

```
(.env) $ pip list
Package    Version
---------- -------
pip        19.1.1
setuptools 41.2.0
(.env) $ pip install tabulate
...
Successfully installed tabulate-0.8.6
(.env) $ python
>>> import tabulate
>>> print(tabulate.tabulate([[1, 2], [3, 4]]))
-  -
1  2
3  4
-  -
```

EXERCISE

1. Create virtual environment
2. Install `tabulate`
3. Print table with file/directory and its size using `os.path.getsize(path)`

```
$ python my_ls.py
-----------  ----
compute.py     47
helpers.py     59
prezentace   4096
my_ls.py      150
__pycache__  4096
-----------  ----
```

IMPORTING OWN MODULE

helpers.py

```
def avg(sequence):
        return sum(sequence) / len(sequence)
```

compute.py

```
import helpers
print(helpers.avg([1, 2, 3, 4])
```

terminal

```
$ python compute.py
2.5
```

IMPORTING OWN MODULE

helpers.py

```
def avg(sequence):
        return sum(sequence) / len(sequence)

print('We are in helpers.py')
print(avg([1, 2])
```

compute.py

```
import helpers
print('We are in compute.py')
print(helpers.avg([1, 2, 3, 4])
```

```
$ python compute.py
We are in helpers.py
1.5
We are in compute.py
2.5
```

IMPORTING OWN MODULE

helpers.py

```
def avg(sequence):
        return sum(sequence) / len(sequence)
if __name__ == '__main__': # <---
        print('We are in helpers.py')
        print(avg([1, 2])
```

compute.py

```
import helpers
print('We are in compute.py')
print(helpers.avg([1, 2, 3, 4])
```

```
$ python compute.py
We are in compute.py
2.5
```

MODULE VS PACKAGE

```
package/
        - __init__.py
        - a.py
        - b.py
```

```
import package.a
import package.b

package.a.function_1()
package.b.function_2()
```

EXERCISE - `mathlib`

1. Create a new module `mathlib`
2. implement `avg(sequence)`
3. implement `hypotenuse(a, b)`
4. use it from different module

$$c = \sqrt{a^2 + b^2}.$$

```
>>> import mathlib
>>> mathlib.avg([1, 2])
1.5
>>> mathlib.hypotenuse(3, 4)
5.0
```

END