A Short and Incomplete Introduction to Python

Part 6: Exception handling

Riccardo Murri <riccardo.murri@uzh.ch>,
Sergio Maffioletti <sergio.maffioletti@uzh.ch>
S3IT: Services and Support for Science IT,
University of Zurich

Exception handling

Exceptions

"Exceptions" is the name given in Python to error conditions.

You can write code that intercepts some error conditions and reacts appropriately.

See also: http://docs.python.org/library/exceptions.html

What does an exception look like?

```
>>> stream.write('foo')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
IOError: File not open for writing
```

What does an exception look like?

```
>>> stream.write('foo')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
IOError: File not open for writing
```

This is the exception *message*: it is supposed to be read by the (human) user.

What does an exception look like?

```
>>> stream.write('foo')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
IOError : File not open for writing
```

This is the exception *class name*; it is used for catching exceptions (syntax in the next slide).

```python
try:
    # code that might raise an exception
except SomeException:
    # handle some exception
except AnotherException as ex:
    # the actual Exception instance
    # is available as variable 'ex'
finally:
    # performed on exit in any case
```

The optional `finally` clause is executed on exit from
the `try` or `except` block in *any* case.

Reference: http://docs.python.org/reference/compound_stmts.html#try

Raising exceptions in your code

Use the **raise** statement with an `Exception` instance:

```python
if an_error_occurred:
  raise RuntimeError("Spider sense is tingling.")
```

Exercise 6.A: Try loading file `values2.txt` with the `load_data()` function from Exercise 4.A – what exception does Python raise?

Edit the `load_data()` function into a `load_data2()` that *ignores* any line that does not contain an integer number.

Bonus points if you can write `load_data2()` so that it has exactly the same output of `load_data()`, i.e. minimize the number of rejected input lines.