Day 4

Last session

- Sorting
- Reading and writing files
- Using *import* to import modules
- Parsing genomic regions
- Obtaining sequences and reverse complement using *ucscgenome* and *str.translate()*.

This session

- Dictionaries
- Functions
- Scripting
- Rosalind

Dictionaries

- Dictionaries are *collections*, just like lists but…
- They are indexed by values (*keys*) other than numbers
- They are unordered

Dictionaries

```
# an empty dictionary
a = {}
print(a)

# a dictionary, indexed by strings
a = {"one": 1, "two": 2, "three": 3}
print(a["two"])

# does a key exist? (the `in` operator)
print("one" in a)
print("four" in a)

# or like this
if "one" in a:
    print("a contains 'one'")
```

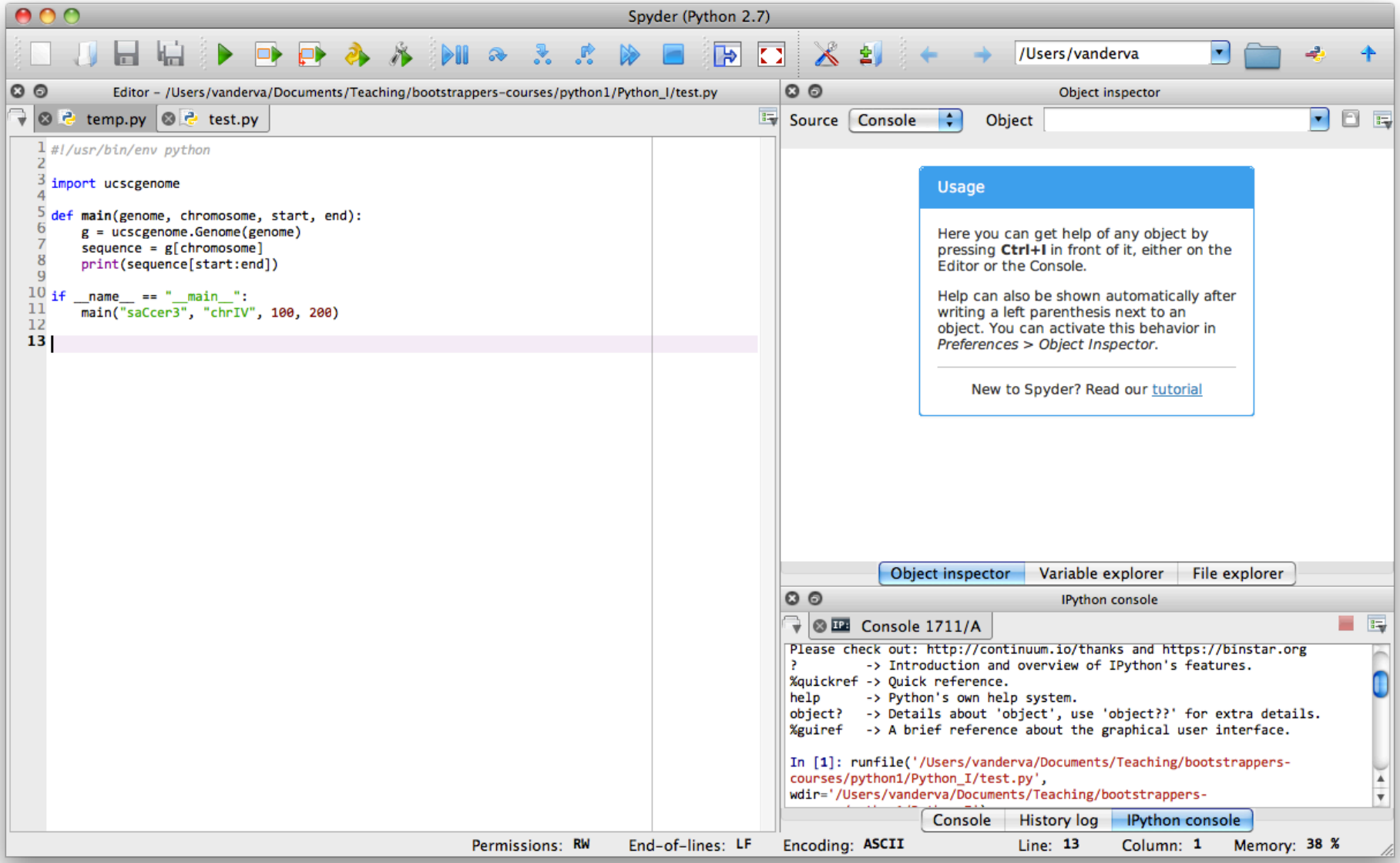
Functions

- Functions are “predefined blocks of code” that accept parameters and return values
- We have used quite a few functions already.

```
def area(length, width):
    return length * width

def mean(numbers): # this defines a function `mean()`
    mysum = sum(numbers)
    return float(mysum) / len(numbers)
```

Spyder IDE


Spyder (Python 2.7)
/Users/vanderva
Editor - /Users/vanderva/Documents/Teaching/bootstrappers-courses/python1/Python_I/test.py
temp.py
test.py
#!/usr/bin/env python
import ucscgenome
def main(genome, chromosome, start, end):
g = ucscgenome.Genome(genome)
sequence = g[chromosome]
print(sequence[start:end])
if __name__ == "__main__":
main("saCcer3", "chrIV", 100, 200)
Object inspector
Source
Console
Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Object Inspector.
New to Spyder? Read our tutorial
Object inspector
Variable explorer
File explorer
IPython console
Console 1711/A
Please check out: http://continuum.io/thanks and https://binstar.org
? -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help -> Python's own help system.
object? -> Details about 'object', use 'object??' for extra details.
%guiref -> A brief reference about the graphical user interface.
In [1]: runfile('/Users/vanderva/Documents/Teaching/bootstrappers-courses/python1/Python_I/test.py',
wdir='/Users/vanderva/Documents/Teaching/bootstrappers-
Console
History log
IPython console
Permissions: RW
End-of-lines: LF
Encoding: ASCII
Line: 13
Column: 1
Memory: 38 %

Scripting

```
#!/usr/bin/env python

def area(length, width):
    return length * width

def mean(numbers): # this defines a function `mean()`
    mysum = sum(numbers)
    return float(mysum) / len(numbers)

def hello():
    print("Hello World")

def main(): # this defines a function `main()`
    x = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
    print(mean(x))
    print(type(mean(x)))
    myvariable = hello()
    print(type(myvariable))
    print(area(10, 15))

if __name__ == "__main__":
    main()
```