Intro to Creative Computing

Week 10: Classes and Objects

Organization of code

- What is the purpose of Functions?
- Most of programming languages have them …
- **Purpose?**
  - effectively cleaning up your code, partitioning it into simpler blocks that you can rely on …

Organization of code

- Today’s lecture – **classes**!
  - Another way to organize your code.
  - A whole theory about this called **Object Oriented Programming (OOP)**, but that’s just Software Development people being bored and calling things with fancy names!
- Objects and classes - another way how to organize your code when you have bigger projects

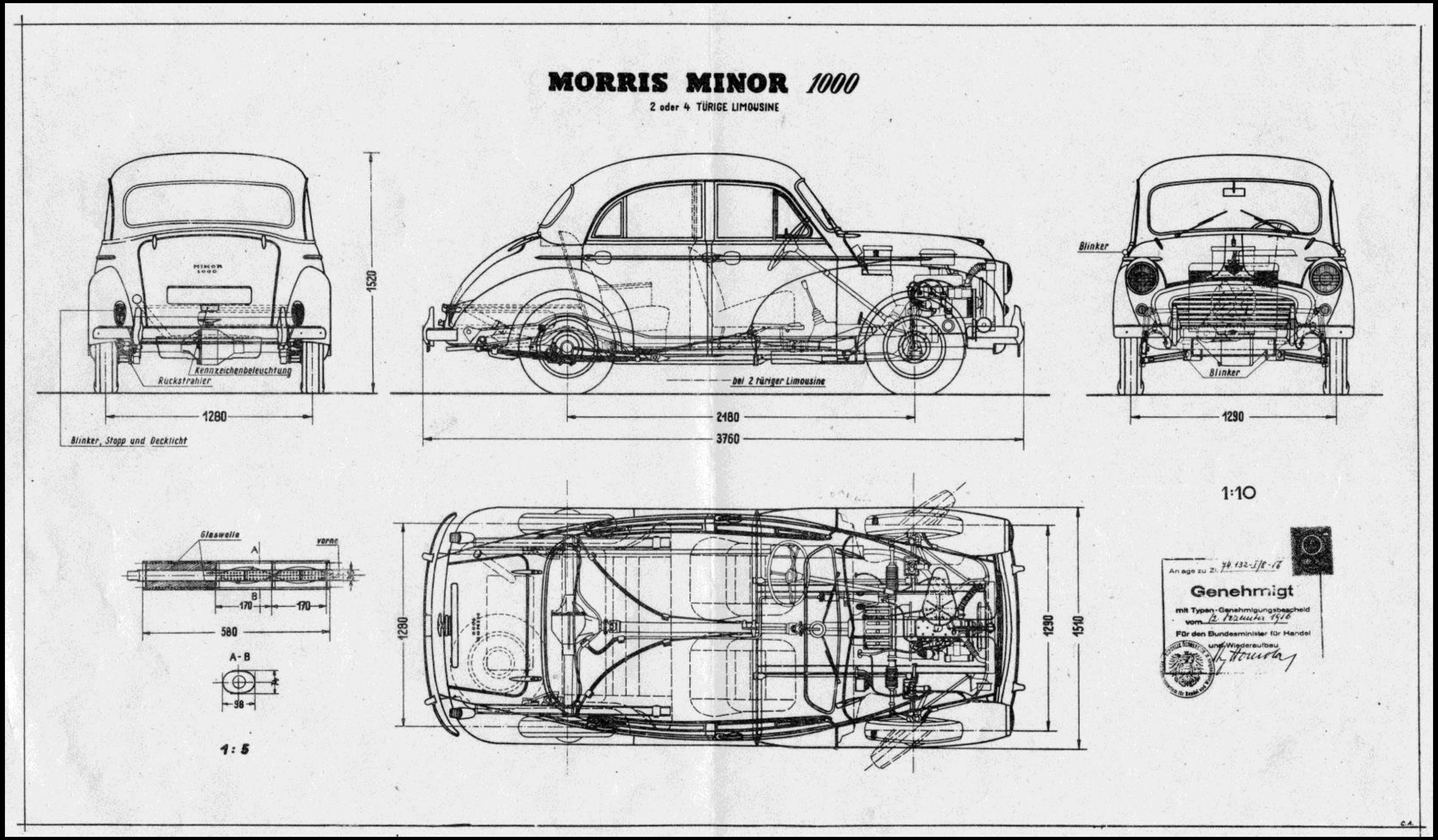
So what are they?

- Class = similar to a blueprint for some object we repeatedly use
- Instance of the class *or* Object = one particular example of this type

- Example:
  - Blueprint to a car (design document) describing the concept
  - Actual cars, many of them, each one a bit different

One out of many

- **Class** = Blueprint for some type of an object:



One out of many

- **Objects** = It turns out that you can have many **instances** of that class – but they can each be different:

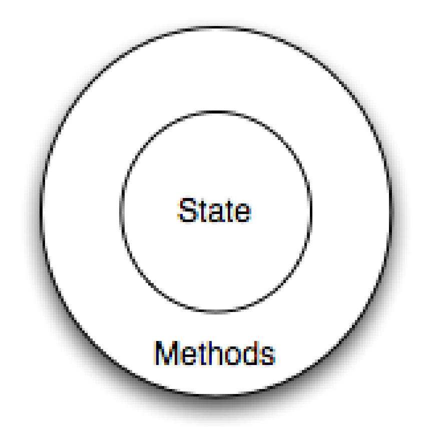
Classes + Objects:

- Pseudocode description

```
CLASS Car
   has 4 wheels
   has a specific color
   I can ride it somewhere
   I can repaint it
```

Values it **has**

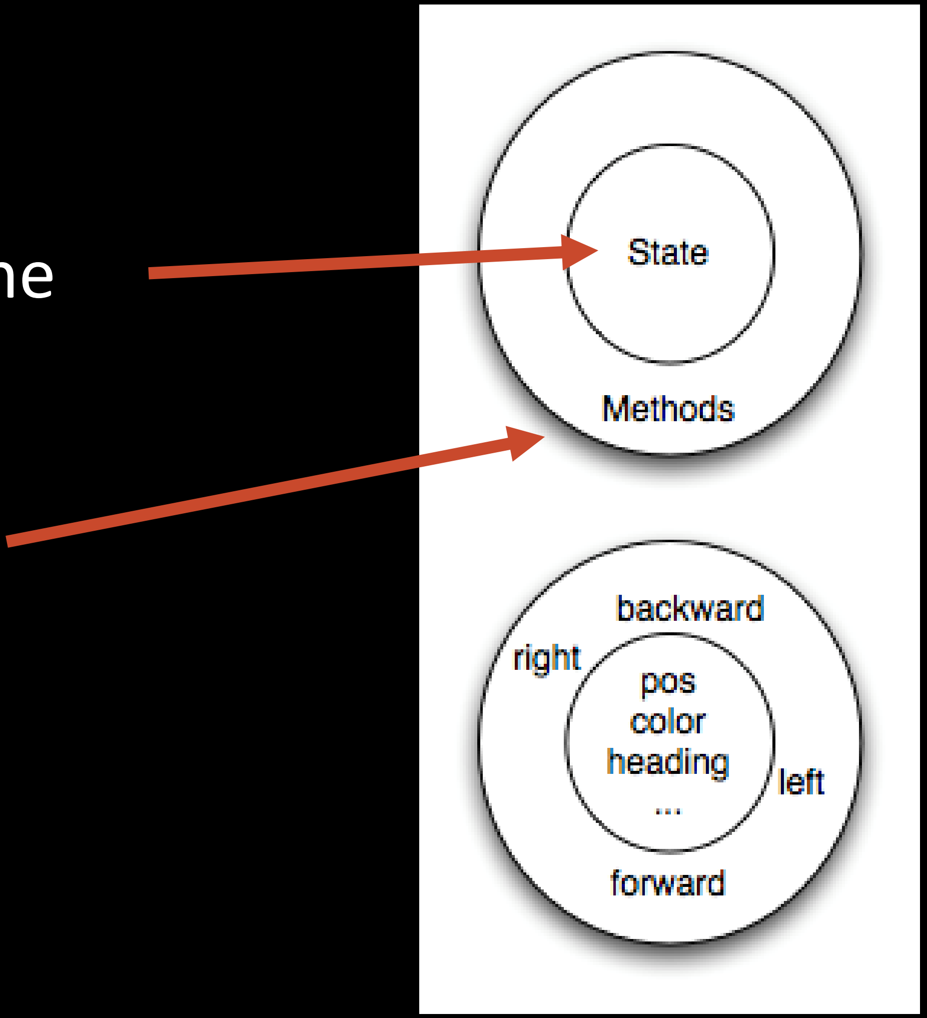
Functions it can **do**



- Silly examples … seems like pointless to specify it when I have just few cars (or when I have none :D)
- But if there’s a factory with 1000s of cars then it starts making sense, that each car would be represented like this – as one instance of some generic type of a thing (car)

Classes + Objects:

- **Values of variables** that remain for that one instance of an object
- And **methods** that we can call (to all the instances)



Classes + Objects:

- Right … what am I talking about this whole time … how does it relate to programming in any way?

Classes + Objects:

- Right … what am I talking about this whole time … how does it relate to programming in any way?

- Well its because we can write these **templates** in code – templates which have the same placeholders with different values (like color of different instances of cars) and functionalities that are the same for every object.

Code samples:

- Syntax basics:
  - The basic syntax of a **class**

```
class Car:

    # more to come in here
```

Code samples:

- Syntax basics:
  - The basic syntax of a **class**
  - With a **constructor** __init__ function

```
class Car:

    def __init__(self):
        """ Create a new car """
```

Constructor is a function called when the object is created – it’s always called __init__

Code samples:

- Syntax basics:

  - The basic syntax of a **class**
  - With a **constructor** __init__ function
  - Values it has (color)

```
class Car:

    def __init__(self):
        """ Create a new car """
        self.color = "red“
```

Self is a reference to the object, you can store values there such as this self.**some_variable_name**

Code samples:

- Syntax basics:
  - The basic syntax of a **class**
  - With a **constructor** __init__ function
  - Values it has (color)
- Creating a object instance of a class c = Car()

```
class Car:

    def __init__(self):
        """ Create a new car """
        self.color = "red“

# then you can make it:

c = Car()
print(c)
```

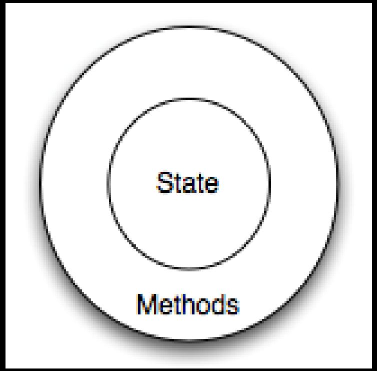
Code samples:

- Syntax basics:
  - The basic syntax of a **class**
  - With a **constructor** __init__ function
  - Values it has (color)
- Creating a object instance of a class c = Car()
  - Accessing it’s values

```
class Car:

    def __init__(self):
        """ Create a new car """
        self.color = "red“

# then you can make it:

c = Car()
print(c)

# and access it’s values:
print(c.color)
```

Self:

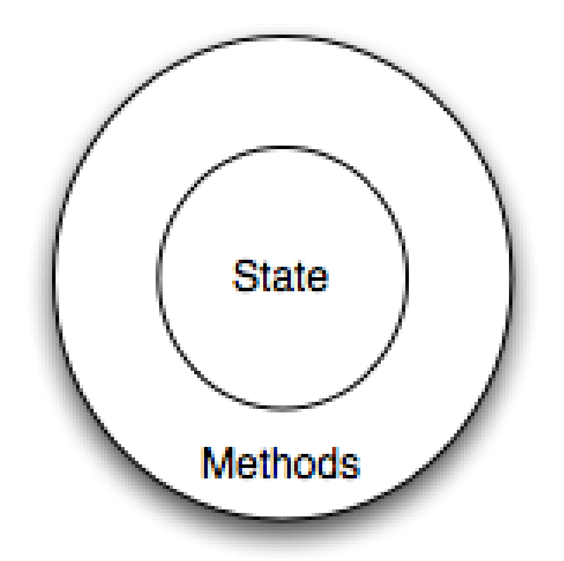
- **Self** was used to refer to the variables of the object



- If you don’t place the variable inside the self of the object, it will simply be a normal **local variable**

```
class Car:

    def __init__(self):
        """ Create a new car """
        self.color = "red“
        some_other_variable = 13
        print(some_other_variable)

c = Car()
print(c)

print(c.color)
print(c.some_other_variable) < ERR
```

Summary: Objects …



- Have **variables**
- Can do **functions / methods**
- There can be **multiple instances** each one unique

```
first_car = Car()
first_car.color = “red”

second_car = Car()
second_car.color = “blue”
```

Objects we already know…

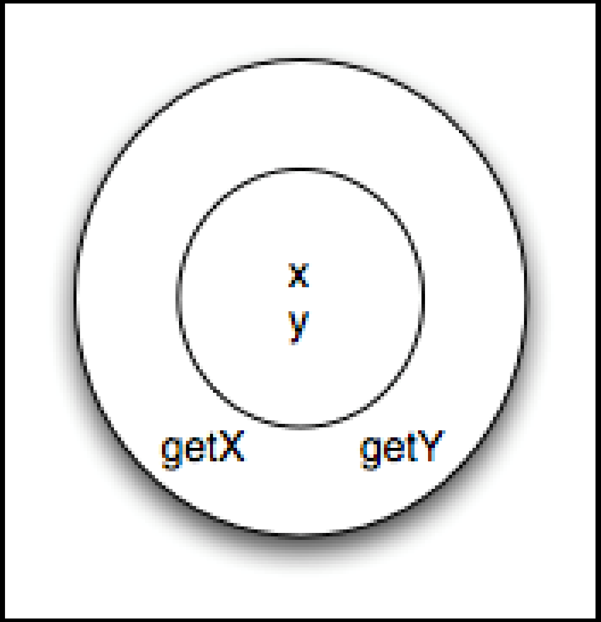
- Strings with their **.lower()** functions
- Files with **.read()**
- Turtles with individual values and functions

```
import turtle

bob = turtle.Turtle()
bob.color('red')
bob.forward(5)
bob.left(90)

tess = turtle.Turtle()
tess.color('blue')
tess.forward(12)
tess.left(180)
```

Sensible example:

- Let’s think about a point. Points in math have 2 common elements, x and y

- **Example: class101.py**



PS:

- In real life this is really useful.
- Not even when dealing with tons of objects, but even when there's just one instance – the way it separates the code into clear chunks (*rendering_handler* does all the rendering while *model_handler* does all the machine learning … I don't need to search for where each of these is …).

Pause 1

Reading:

- Lecture in book – ch17!
- https://runestone.academy/runestone/books/published/thinkcspy/ClassesBasics/toctree.html

- Topics there include:
  - Arguments to functions in objects
  - Printing out objects
  - … we will have a look at these again in later classes

Exercise:

- Class **Rectangle**
  - Write / draw it down on a paper first (yeah it will be simple, but let’s still do it) and show me:
    - Functions – what can it do?
    - Values – what should it know about itself?
    - Reporting function (print it’s values)

  - Probably would like to have
    - area(),
    - surface()
    - **Advanced:** contains/intersects(with another Rectangle)

Pause 2

Libraries

- What is a library?

Libraries

- What is a library?

```
import os
import math
import sys
import matplotlib

print(math.pi)
print(sys.version)
```

Libraries

- What is a library?

- *(semi-definition) Python library is a collection of functions and methods that allows you to perform many actions without writing your code.*
  - Often community made
  - Often ready made for fast work
  - Often with cool examples!

Versions?

- python, pip,
  python2, pip2,
  python3, pip3

- conda

```python
import sys
print(sys.version)
```

```
# linux + mac

which python
python --version

which pip

conda info
```

```
# windows

where python*
python --version

where pip

conda info
```

Versions?

```
conda info
```

- conda

- *Conda is an open source package management system and environment management system that runs on Windows, macOS and Linux. Conda quickly installs, runs and updates packages and their dependencies. Conda easily creates, saves, loads and switches between environments on your local computer. It was created for Python programs, but it can package and distribute software for any language.*

Path?

- Example (my pc without python):

- C:\Users\vitek-ntb-win>echo %PATH%
- *C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\WINDOWS\System32\WindowsPowerShell\v1.0\;C:\WINDOWS\System32\OpenSSH\;D:\Program Files\CalibreEPUB\;C:\Program Files\PuTTY\;D:\Program Files\Git LFS;C:\Users\vitek-ntb-win\AppData\Local\Microsoft\WindowsApps;C:\Users\vitek-ntb-win\AppData\Local\GitHubDesktop\bin*

```
# linux + mac
echo $PATH

# windows
echo %PATH%
```

Libraries … ?

- How does this have anything to do with libraries?

```python
import matplotlib
```

- Besides some default libraries which come in all python versions (*standard libraries*), many other require installation and also have versions – you might need to do this first:

```
pip install matplotlib
```

… actually

```
pip install matplotlib
```

- This ^^ only works in a case where someone has released a built code for us
- If it’s a small and obscure code, we might need to

```
pip install git+https://github.com/matplotlib/matplotlib.git
```

… actually

```
pip install matplotlib
```

- This ^^ can also sometimes install a new version of the code, while the rest of our code relies on some older version, in that case …

```
pip install matplotlib==


pip install matplotlib==2.0.2
```

… actually

- All of this is **usually relatively doable** … you just test if it works and if it doesn’t *someone online probably encountered the same problem* and their solution can help …

The task?

- Find a python library of interest
- Install it
- Try it’s examples
- And show it to me!

The task?

- Find a python library of interest
- Install it
- Try it's examples
- And show it to me!

- Some places to start?
  - https://medium.mybridge.co/34-amazing-python-open-source-libraries-for-the-past-year-v-2019-93d6ee11aceb
  - https://github.com/vinta/awesome-python
  - Bokeh (nice graphs), Pyxel and Pygame (games), BeautifulSoup and nltk (natural language processing)

The task?

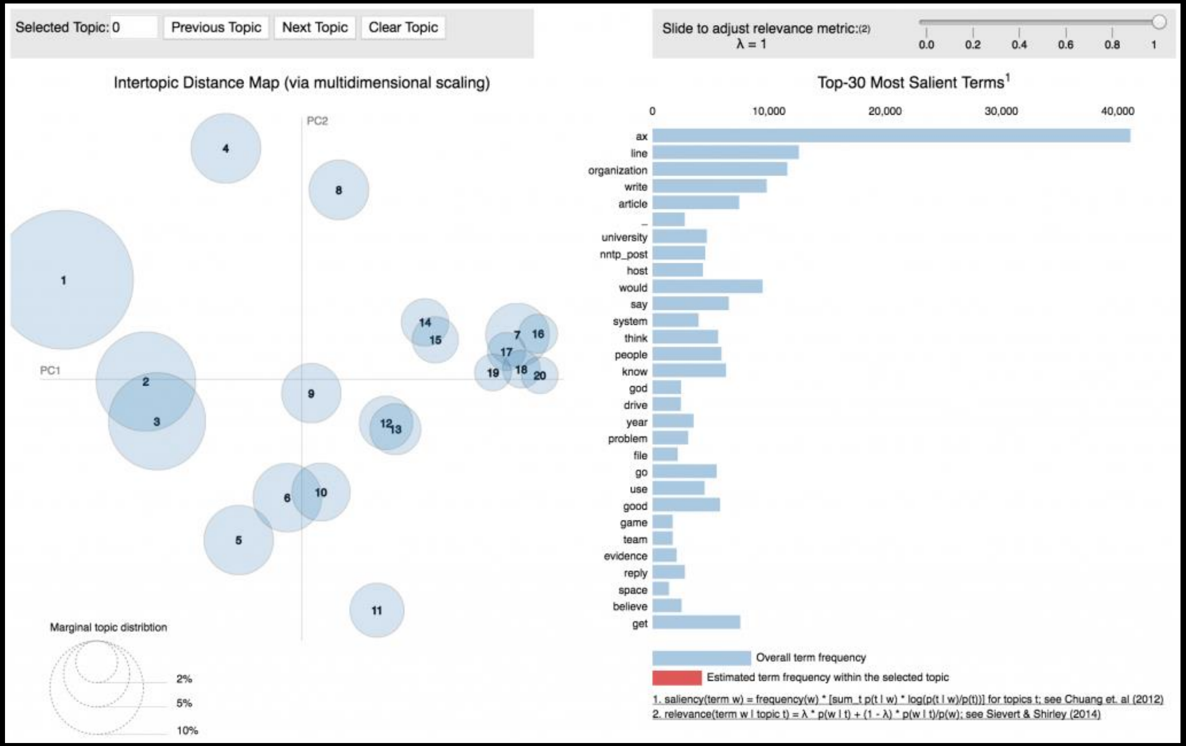
- Find a python library of interest
- Install it
- Try it’s examples
- And show it to me!

```
# hint:
# if things don’t work on local, use
# google colab and this

!pip install matplotlib
```

- Some places to start?
  - https://medium.mybridge.co/34-amazing-python-open-source-libraries-for-the-past-year-v-2019-93d6ee11aceb
  - https://github.com/vinta/awesome-python
  - Bokeh (nice graphs), Pyxel and Pygame (games), BeautifulSoup and nltk (natural language processing)

Bonus: Natural Language Processing?

- Tutorial with **gensim** library:
    - https://radimrehurek.com/gensim/auto_examples/index.html
    - https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/



Selected Topic: 0

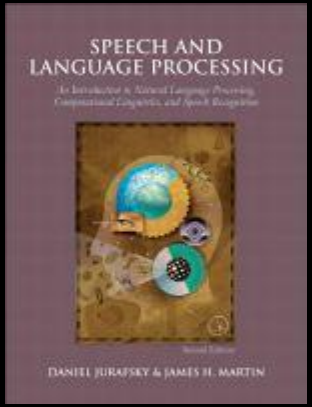
Slide to adjust relevance metric:(2) λ = 1

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

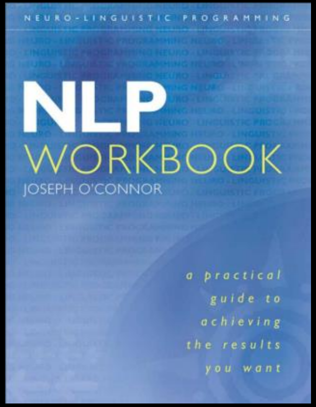
Bonus: Natural Language Processing?

Books

- **Academic** one: https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf can be borrowed in our library though!


SPEECH AND LANGUAGE PROCESSING
DANIEL JURAFSKY & JAMES H. MARTIN

- Hands on with **nltk** library: http://www.nltk.org/book/


NEURO-LINGUISTIC PROGRAMMING
NLP WORKBOOK
JOSEPH O'CONNOR
a practical guide to achieving the results you want

- Book with **just concepts**: https://doc.lagout.org/science/0_Computer%20Science/3_Theory/Neural%20Networks/Neuro%20Linguistic%20Programming%20WorkBook.pdf