

# Classless but Ebbinghaus-polite Python Coding

Howard Schneider

Sheppard Clinic North, Toronto, Canada

[github.com/howard8888](https://github.com/howard8888)

PyCon Canada 2018

# Anything that let's me transform an idea into code is great



# Python

# OOPs, ...

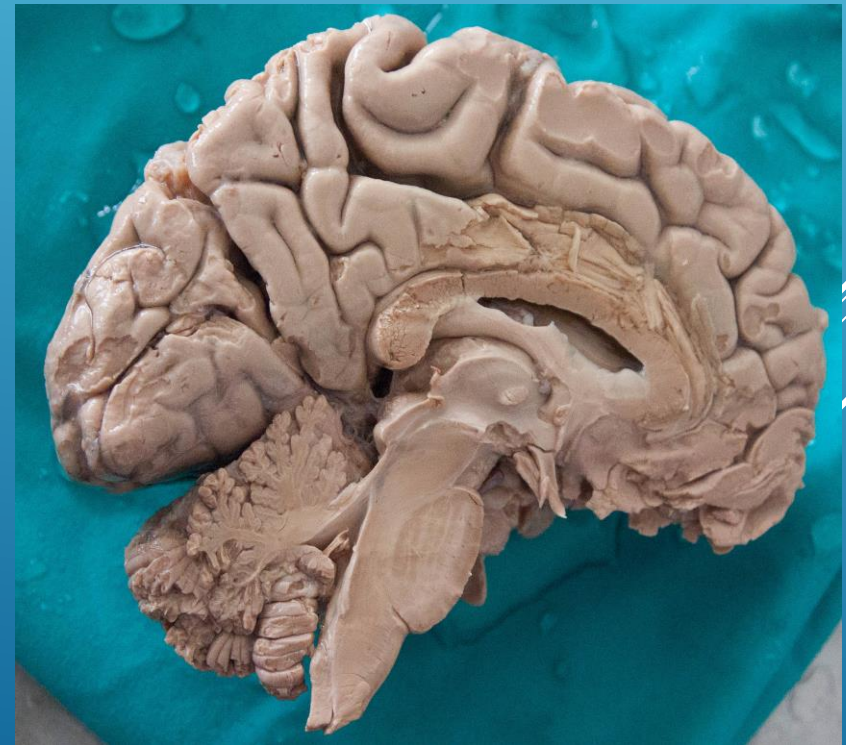


This talk is not 'against' OOP....  
In the Python textbook, OOP seems  
like a wonderful idea....

*But*, as I code and code my project,  
the logic behind OOP – for my project  
anyway– does not really hold up....



# My Project: Simulation of a Biologically Inspired Cognitive Architecture





Large problem to solve – “*Neural Symbolic Gap*”

- **Neural Network** – phenomenal image processing and reinforcement learning
- **Child** – phenomenal causal learning with few examples (eg, Gopnik)





+



=

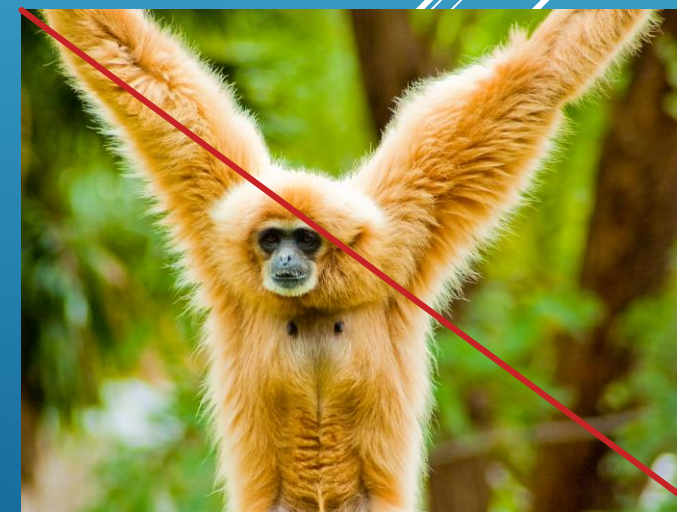


**“panda”**  
**57.7% confidence**

**“gibbon”**  
**99.3 % confidence**

Goodfellow, I.J., Shlens, J. and Szegedy, C. (Google Mountainview), Explaining and Harnessing Adversarial Examples, ICLR 2015.

It's still a Panda – and the 3 year old boy would know this!!  
(and.... 3 year old only needs 1 or 2 photos for training, not 1000s)





## Deep Learning Neural Network

Pattern Recognition  
→ Recognize the World

Need 1000's examples for learning

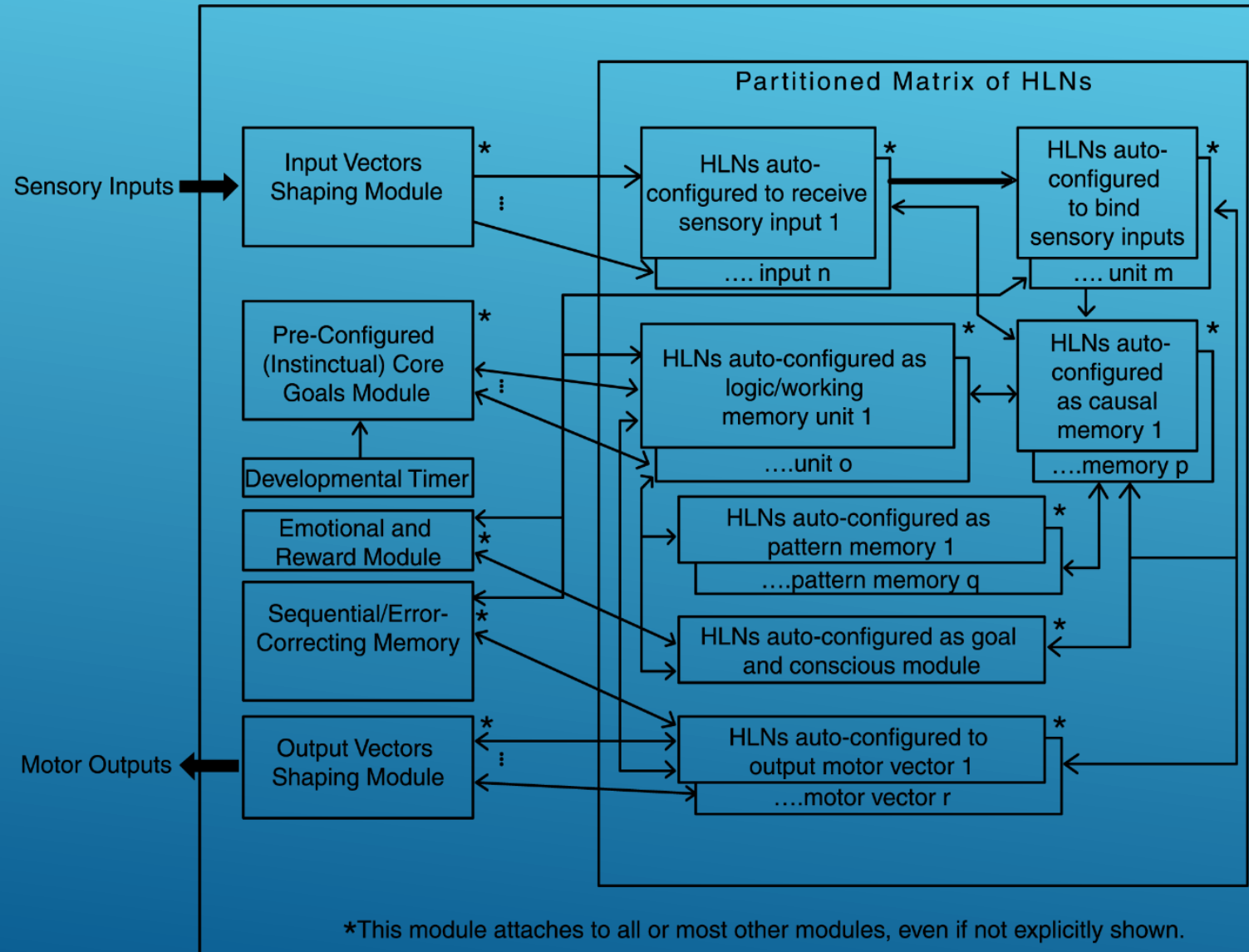
## 3 Year Old Human Child

Model Building + also Pattern Recognition  
→ Explain the World

A few examples enough



# Cognitive Architecture to simulate....



MBLS ("Meaningful-Based Learning System) based on this "Meaningful-Based Cognitive Architecture"



# Build small MBLS simulation in Python....

```
File Edit Search View Encoding Language Settings Tools Macro Run
mbls203.py
1 #
2 #
3 #MBLS Simulation
4 #Meaningful Based Learning System
5 #Language: Python 3.6
6 #CPU, GPU, OS: Independent unless noted below
7 #
8 #Howard Schneider
9 #howard.schneider@gmail.com
10 '''At the time of this writing, despite the human
11 in sensory processing and reinforcement learning
12 Kavukcuoglu, Silver, et al., 2015), such neural
13 cannot causally make sense of their environment
14 (Gopnik, Glymour, Sobel et al., 2004; Waismeyer,
15 Recent successful work by Graves, Wayne, Reynolds,
16 gap. Their model involves an ANN which can read
17 However, like the human brain, the meaningful-based
18 the sensory processing associated with ANNs and
19 cognition, without the use of an external memory
20 '''
21 #
22 #
23 import random
24 import sys #Warning: DEPENDENCIES win64
25 import os.path #Warning: DEPENDENCIES win64
26 import time
27 #
28 VERSION_NUMBER = 2.03
29 '''Migration history: 1.0 Pyth27 -> 2.0 Pyth36
30 -> 2.0x Basic MBLS ->-> goal 2.1x MBLS 5000 HLN'''
31 VERSION_FILE_NAME = 'mbls203.py'
32 CHECKPOINT_ON = False
33 DEVELOPER_USER = True
34 STOP_SCROLLING_BETWEEN_INPUTS = True
35 DEPENDENCIES = ['python36', 'win64']
```

```
Command Prompt - python mbls203.py
Would you like to turn CHECKPOINT tracer on? (y/Y): y
---->NEW INPUT CYCLE -- Press any key to continue ('s' to stop scroll prompts)
A simple camera that can detect the presence or absence of
8 different lines of pixels, sends an input into the MBLS.
  1_
2 | | 3
  4_
5 | | 6
 |8 _7_
[0, 1, 1, 1, 1, 1, 1, 1]
The camera can also input special codes and error codes. If you
enter any valid such code at anytime it will be immediately recognized.
(code 10 -- creates a random input vector, 11 -- creates input with all segs
33 -- history of input vectors,
09 will give hard exit of program, ____ will reset input vector history)
Is line segment 1 there?
Enter 'y' or 'Y' if line segment input, any other key if no line segment: y
Is line segment 2 there?
Enter 'y' or 'Y' if line segment input, any other key if no line segment: y
Is line segment 3 there?
Enter 'y' or 'Y' if line segment input, any other key if no line segment: n
Is line segment 4 there?
Enter 'y' or 'Y' if line segment input, any other key if no line segment: y
Is line segment 5 there?
Enter 'y' or 'Y' if line segment input, any other key if no line segment:
```

# Attempt larger simulation in Python....



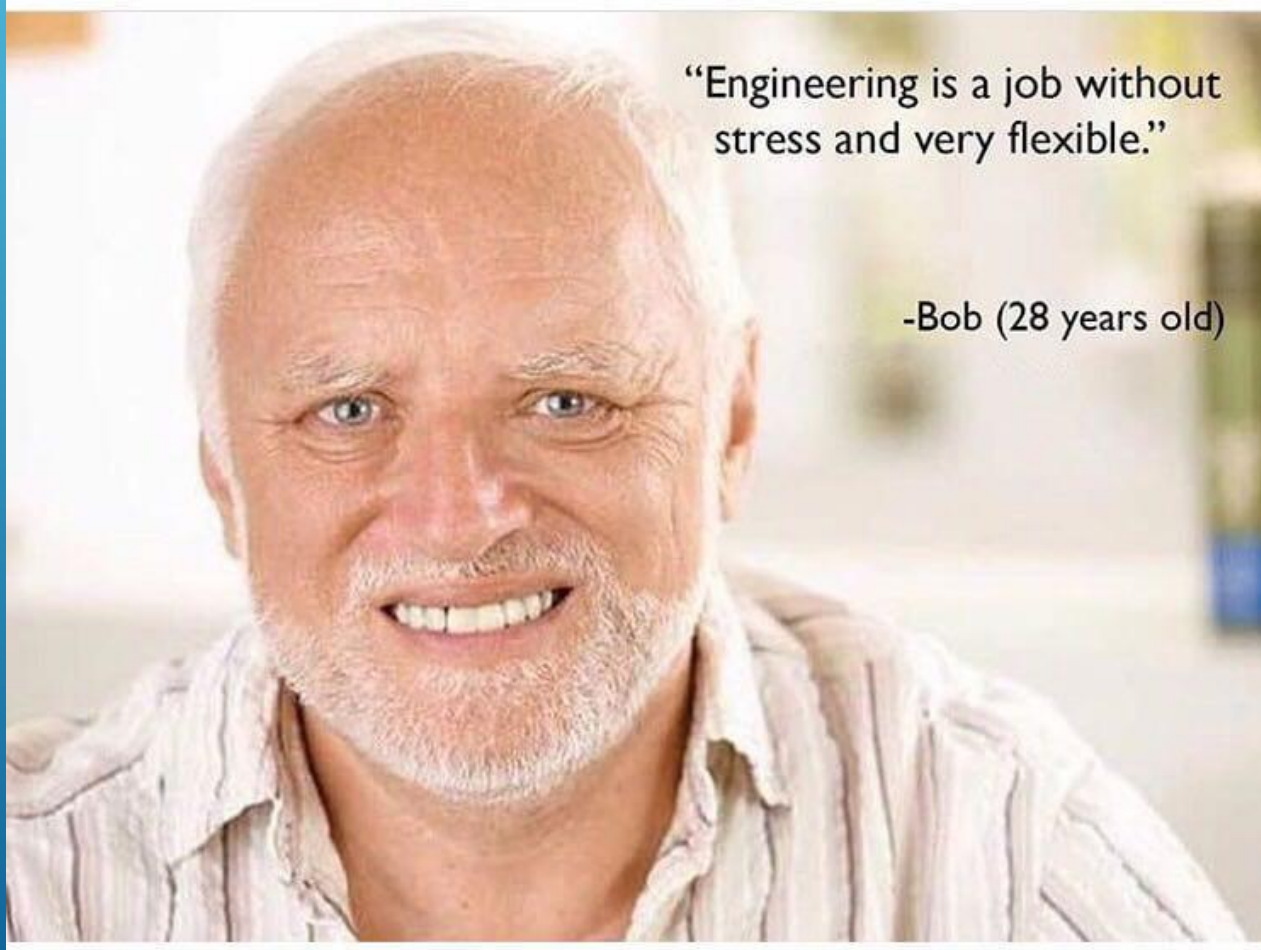
← Controlled by an MBLs  
(hypothetical!!)



Issue: Programmer does not have enough working memories in programmer's human brain to do the Python coding of the simulated cognitive architectures....

Takes too long to learn, ie create memories in human brain, of the computer structures we create in the Python code....

*Too many names and classes and names and  
mess of hierarchies.... on and on.....*



Point of this talk:

If a paradigm uses up very limited and very valuable human working memories:

- Paradigm may be more disadvantageous than advantageous



- Harder to construct a very large code in first place
- Harder for person reading code to figure out what is going on
- **MUST take human 'memory bandwidth' into account**

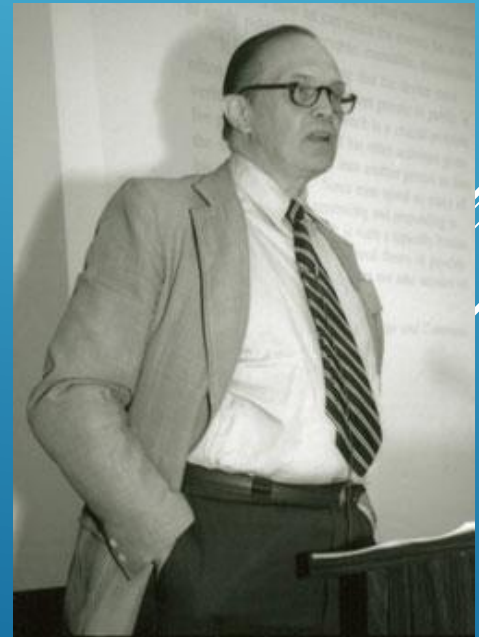
**Hermann Ebbinghaus** – learning curve  
(1885) -- memorizing series of nonsense  
syllables

→ **Human memory has limits and  
characteristics**



**George Miller – (1956) – “The Magical Number Seven, Plus or Minus Two”**

**→ the number of objects an average human can hold in working memory is  $7 \pm 2$**





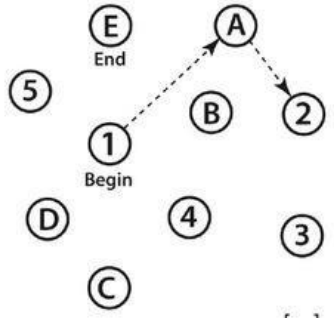
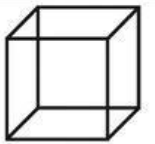
# MOCA Test

## MONTREAL COGNITIVE ASSESSMENT (MOCA) Version 7.1 Original Version

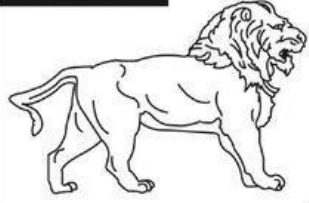
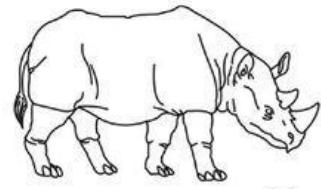
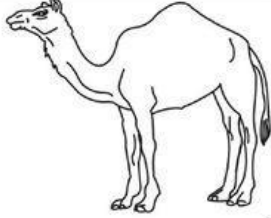
NAME :  
Education :  
Sex :

Date of birth :  
DATE :

### VISUOSPATIAL / EXECUTIVE

		Copy cube	Draw CLOCK (Ten past eleven) (3 points)	POINTS
[ ]	[ ]	[ ]	[ ]	___/5
		Contour	Numbers	Hands

### NAMING

			POINTS
[ ]	[ ]	[ ]	___/3

### MEMORY

Read list of words, subject must repeat them. Do 2 trials, even if 1st trial is successful. Do a recall after 5 minutes.		FACE	VELVET	CHURCH	DAISY	RED	No points
	1st trial						
	2nd trial						

### ATTENTION

Read list of digits (1 digit/ sec.). Subject has to repeat them in the forward order	[ ]	2	1	8	5	4
	Subject has to repeat them in the backward order [ ] 7 4 2					
Read list of letters. The subject must tap with his hand at each letter A. No points if ≥ 2 errors						
[ ] FBACMNAAJKLBFAFAKDEAAAJAMOF AAB						
Serial 7 subtraction starting at 100 [ ] 93 [ ] 86 [ ] 79 [ ] 72 [ ] 65						
4 or 5 correct subtractions: 3 pts, 2 or 3 correct: 2 pts, 1 correct: 1 pt, 0 correct: 0 pt						
___/3						

### LANGUAGE

Repeat : I only know that John is the one to help today. [ ]	___/2
The cat always hid under the couch when dogs were in the room. [ ]	
Fluency / Name maximum number of words in one minute that begin with the letter F [ ] _____ (N ≥ 11 words)	___/1

### ABSTRACTION

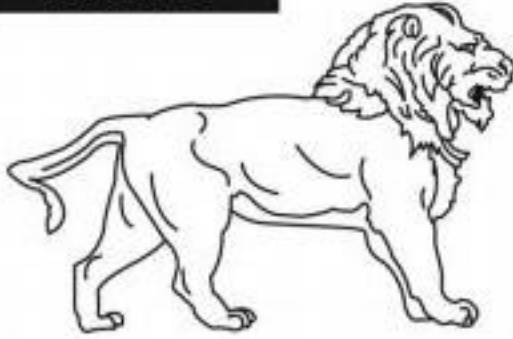
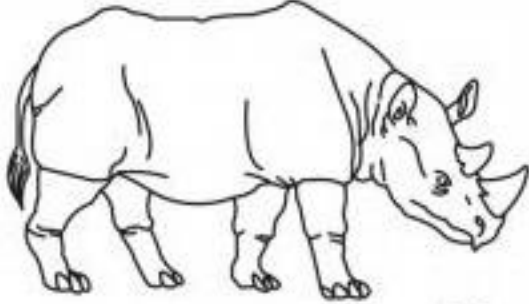
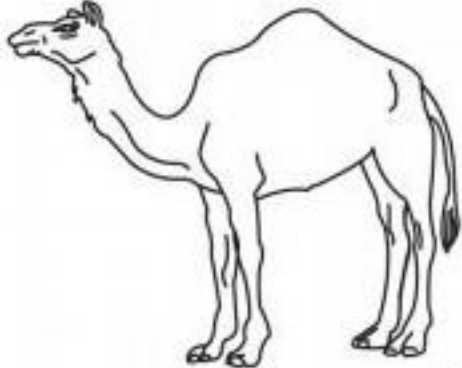
Similarity between e.g. banana - orange = fruit [ ] train - bicycle [ ] watch - ruler	___/2
---	-------

DELATED RECALL	Has to recall words	FACE	VELVET	CHURCH	DAISY	RED	Points for UNCUED recall only	___/5
	WITH NO CUE	[ ]	[ ]	[ ]	[ ]	[ ]		
Optional	Category cue							
	Multiple choice cue							

### ORIENTATION

[ ] Date	[ ] Month	[ ] Year	[ ] Day	[ ] Place	[ ] City	___/6
----------	-----------	----------	---------	-----------	----------	-------

# Humans have a tiny number of working memories

NAMING							
			[ ]	[ ]	[ ]	___/3	
MEMORY		Read list of words, subject must repeat them. Do 2 trials, even if 1st trial is successful. Do a recall after 5 minutes.					
		FACE	VELVET	CHURCH	DAISY	RED	No points
1st trial							
2nd trial							
ATTENTION		Read list of digits (1 digit/ sec.).					
		Subject has to repeat them in the forward order			[ ] 2 1 8 5 4		___/2
		Subject has to repeat them in the backward order			[ ] 7 4 2		
Read list of letters. The subject must tap with his hand at each letter A. No points if $\geq 2$ errors		[ ] FBACMNAAJKLBAFAKDEAAAJAMOF AAB					___/1
Serial 7 subtraction starting at 100		[ ] 93	[ ] 86	[ ] 79	[ ] 72	[ ] 65	___/3
4 or 5 correct subtractions: <b>3 pts</b> , 2 or 3 correct: <b>2 pts</b> , 1 correct: <b>1 pt</b> , 0 correct: <b>0 pt</b>							

- *My simulation deals with hundreds of millions of cortical columns ('HLNs' in the simulation), not a fictional bookstore example in the OOPs chapter of a computer science textbook*
- *Class structure offers very little advantage in creating my simulation but causes me to have to store in my limited brain memory a complex mess of rigid (aghh!!) inheritance hierarchies, class names,....*



***Eliminate!!***



# Replace OOP with 'HOP' (‘Human Oriented Programming’)

- Must minimize anything that taxes the very limited human working memory (or else take months and months to memorize all the variables and cute coding blocks)
- This also allows maintenance by humans with the same limited working memory.

- Avoid short variable names but try to use names which allow the reader to follow what the variable represents
- Use any sort of database, NOT inheritance, to code relationships (pure hierarchies are a myth in real world)
- From point of view of human memory:  
W.E.T. (Wrote Everything Twice) – can be good  
D.R.Y. (Do Not Repeat Yourself) – can be bad

- Ok to allow many and changing variables to participate in an action and share data as long as all is working-memory friendly
- All functions must have built-in unit tests, reflecting weaknesses of human working-memory in evaluating code behavior, even in small decomposed functions

***“Classless but Ebbinghaus-polite Python code”***

“To date there have been 3  
Programming Paradigms .... There  
are unlikely be any others....” -  
Robert C. Martin, “Clean  
Architecture”, 2018





# Martin's 3 Paradigms

1. **Structured Programming**  
(restrict direct transfer control)
2. **Object-Oriented Programming**  
(restrict indirect transfer control)
3. **Functional Programming**  
(restrict assignment)  
-----
4. **? Human Oriented Programming?**  
(restrict anything > working memories)

# Where are the References??

Surprising thing is that despite use of OOP by millions of developers for decades, there is very little reproducible evidence in the literature that OOP is more effective than other paradigms.

# “Classless but Ebbinghaus-polite Python coding”

- (*HOP not OOP*)

[github.com/howard8888](https://github.com/howard8888)  
[howard.schneider@gmail.com](mailto:howard.schneider@gmail.com)

