Psychoinformatics & Neuroinformatics

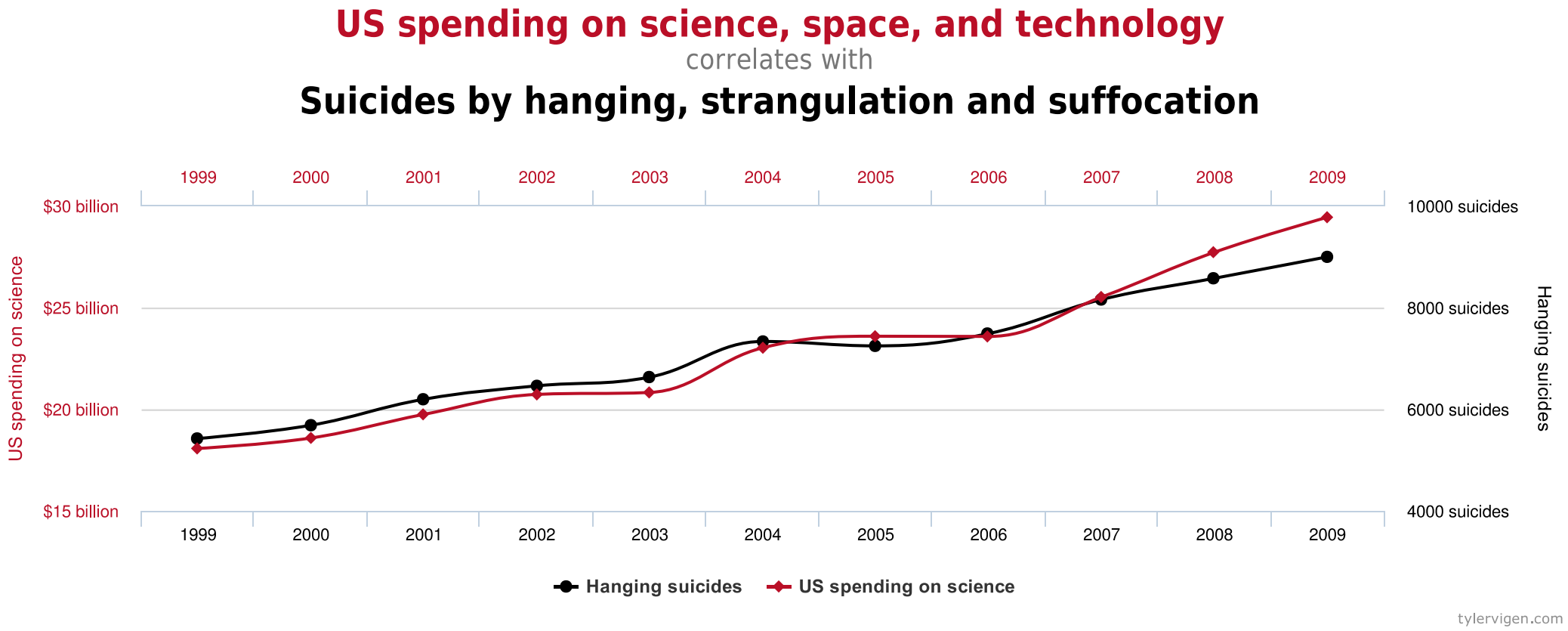
Week 2

Developing Experiments

by Tsung-Ren (Tren) Huang 黃從仁

Spurious Correlation

Correlation does not imply causation



r=0.99789126

Policy based on incorrect conclusions

can be detrimental…

Duke University pays $112M to settle faked-research lawsuit

nature medicine

Explore content ⌄ About the journal ⌄ Publish with us ⌄

nature › nature medicine › articles › article

Published: 22 October 2006

Genomic signatures to guide the use of chemotherapeutics

Anil Potti, Holly K Dressman, Andrea Bild, Richard F Riedel, Gina Chan, Robyn Sayer, Janiel Cragun, Hope Cottrill, Michael J Kelley, Rebecca Petersen, David Harpole, Jeffrey Marks, Andrew Berchuck, Geoffrey S Ginsburg, Phillip Febbo, Johnathan Lancaster & Joseph R Nevins ✉

Nature Medicine **12**, 1294–1300 (2006) | Cite this article

6460 Accesses | **435** Citations | **98** Altmetric | Metrics

ⓘ A Retraction to this article was published on 07 January 2011

ⓘ A Corrigendum to this article was published on 01 August 2008

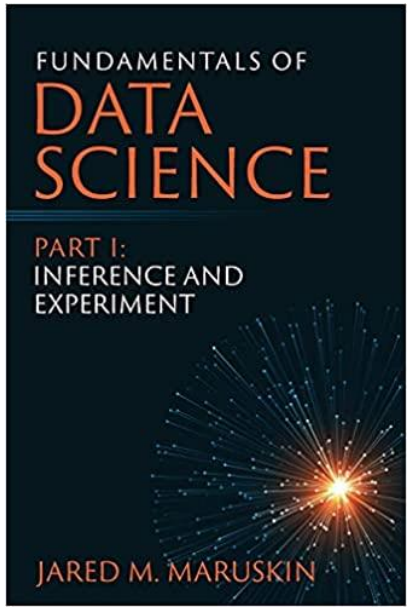
ⓘ A Corrigendum to this article was published on 01 November 2007

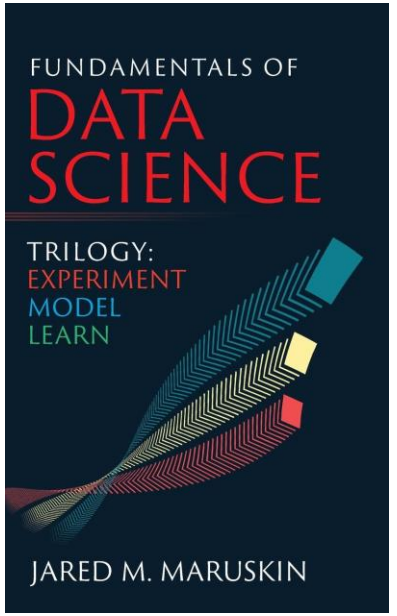
ⓘ This article has been updated

Important Facts

Garbage in, garbage out

2nd parties can't produce high-quality results from low-quality (and often decontextualized) data





Experimentation is an integral part of data science

High-value data scientists often participate in designing exp.

NETFLIX
presents

EXPERIMENTATION
&
CAUSAL INFERENCE

https://www.youtube.com/watch?v=WRGW6xHLy3k

Goals for today

Learning to set up basic within-subject experiments

Within-subject design is statistically more powerful than between-subject design

Learning the client-server architecture for advanced experiments

E.g., human social experiments, remote control of animal experiments

Developing interest to learn more about experimentation

E.g., multi-armed bandits (example), difference in differences, regression discontinuity design

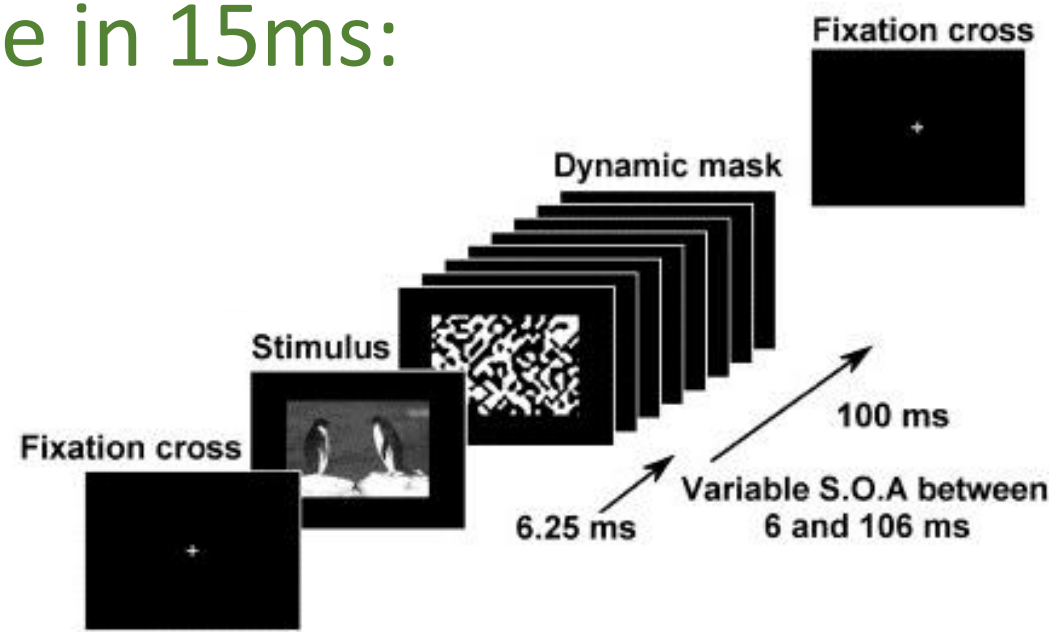
https://www.youtube.com/watch?v=KxXJ57OgURg

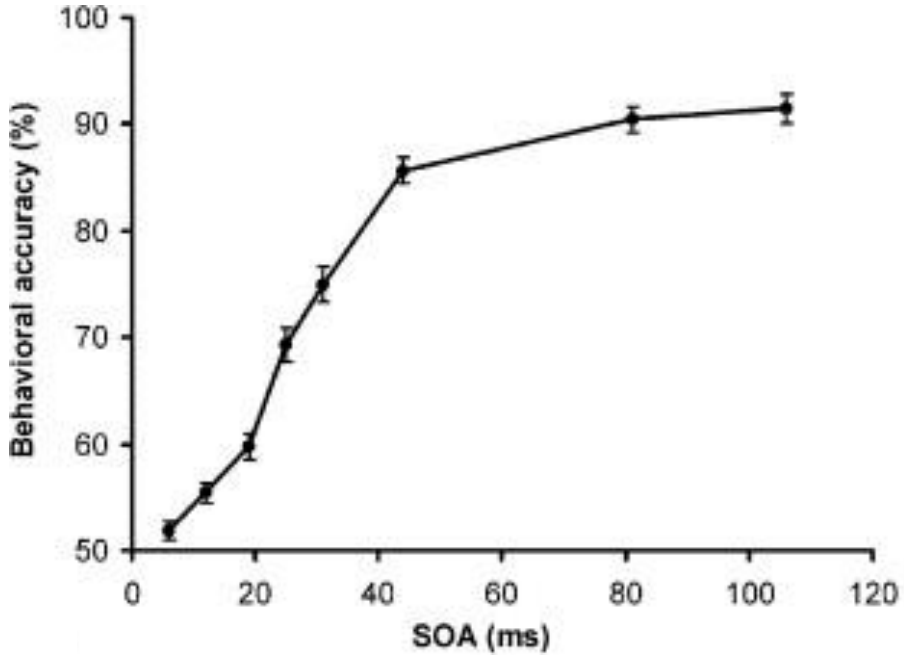
How fast can we understand an image?

First studied by Potter using Rapid Serial Visual Presentation

Potter, M. C. (1975). Meaning in visual search. *Science, 187*(4180), 965-966.

Bacon-Mace et al. (2005) found that animal detection can be done in 15ms:

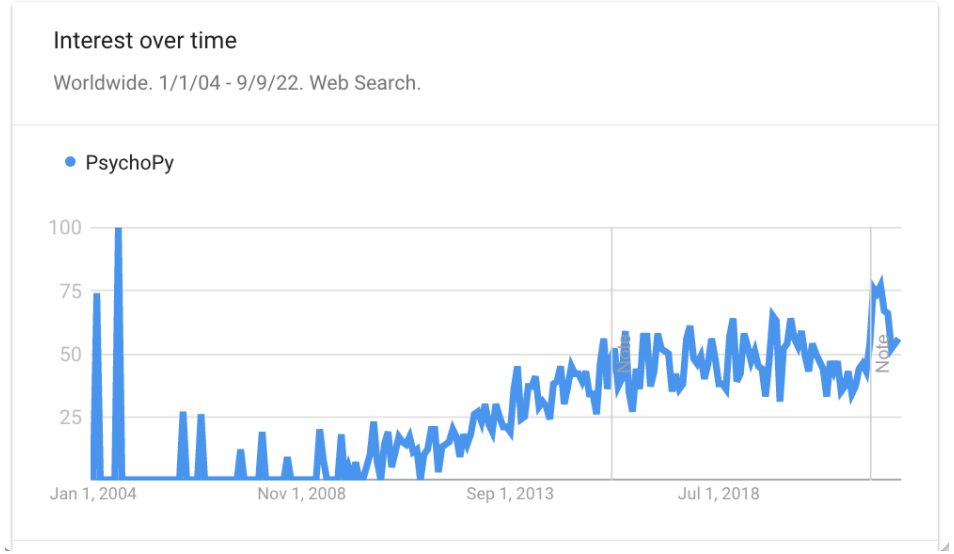




Let’s try to replicate it using PsychoPy

PsychoPy

Now running studies online



Easy
enough for teaching

Precise
enough for psychophysics

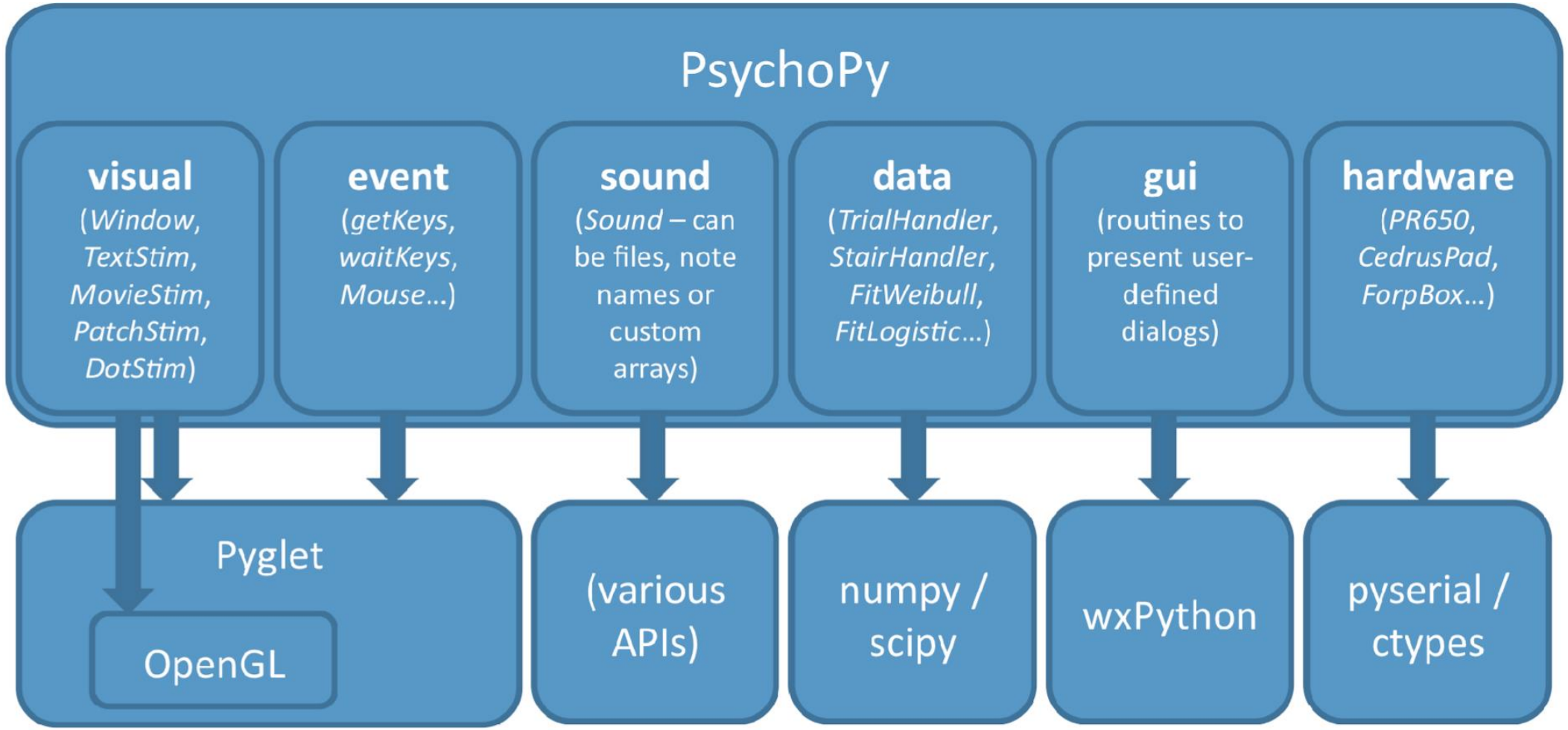
Flexible
enough for everything else

Online
or lab-based, your choice

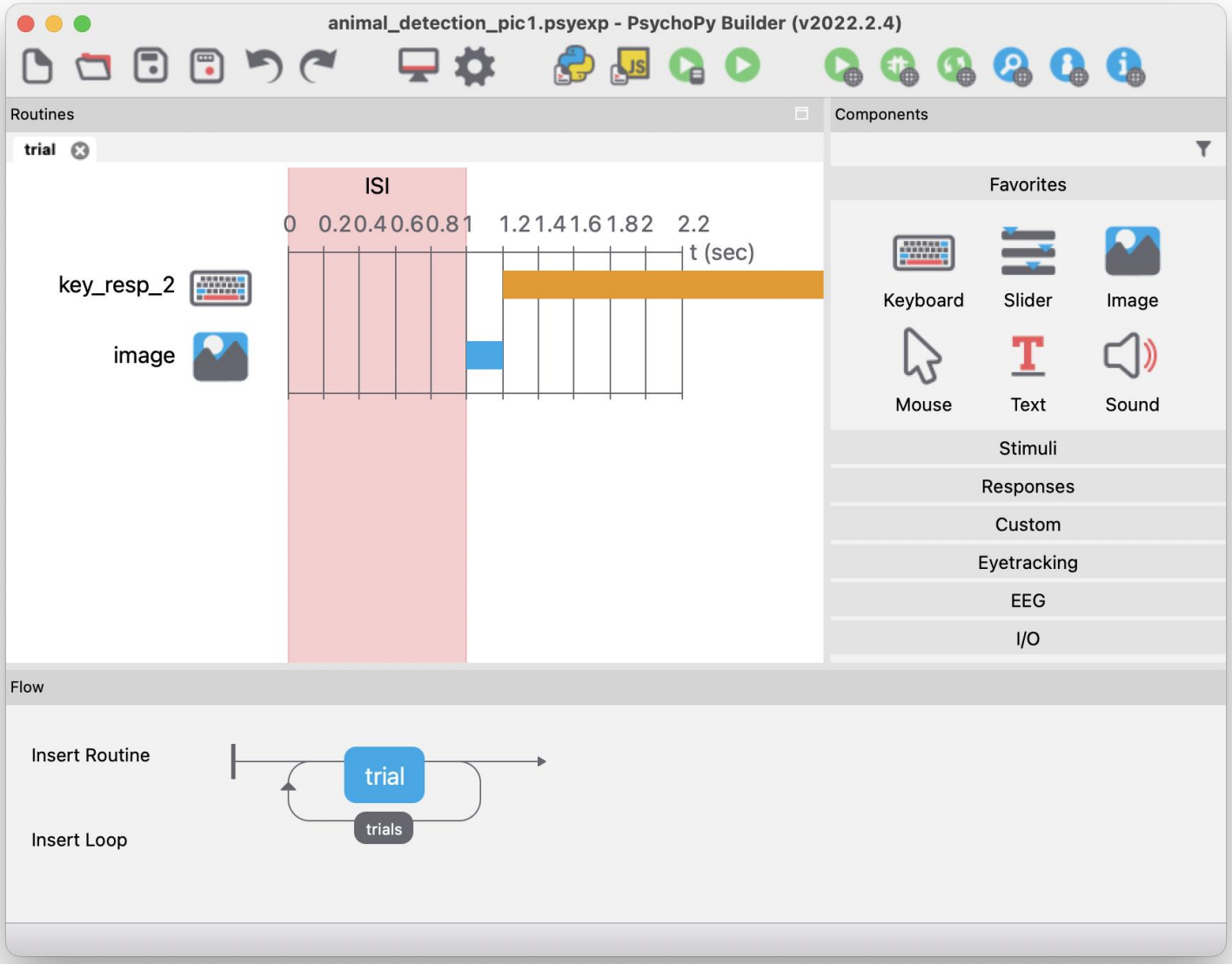
Open Science
because it's what we do

Overview of PsychoPy

[2007 Paper] [2009 Paper] [Video Tutorial] [Manual] [API]



Builder (animal_detection_pic1.psyexp)









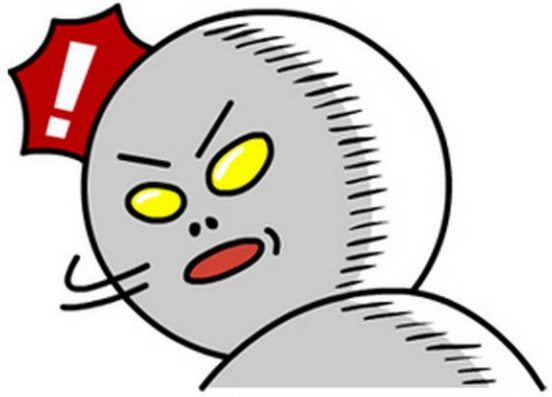



The problems of Builder

Needing to *manually* establish a stimulus table

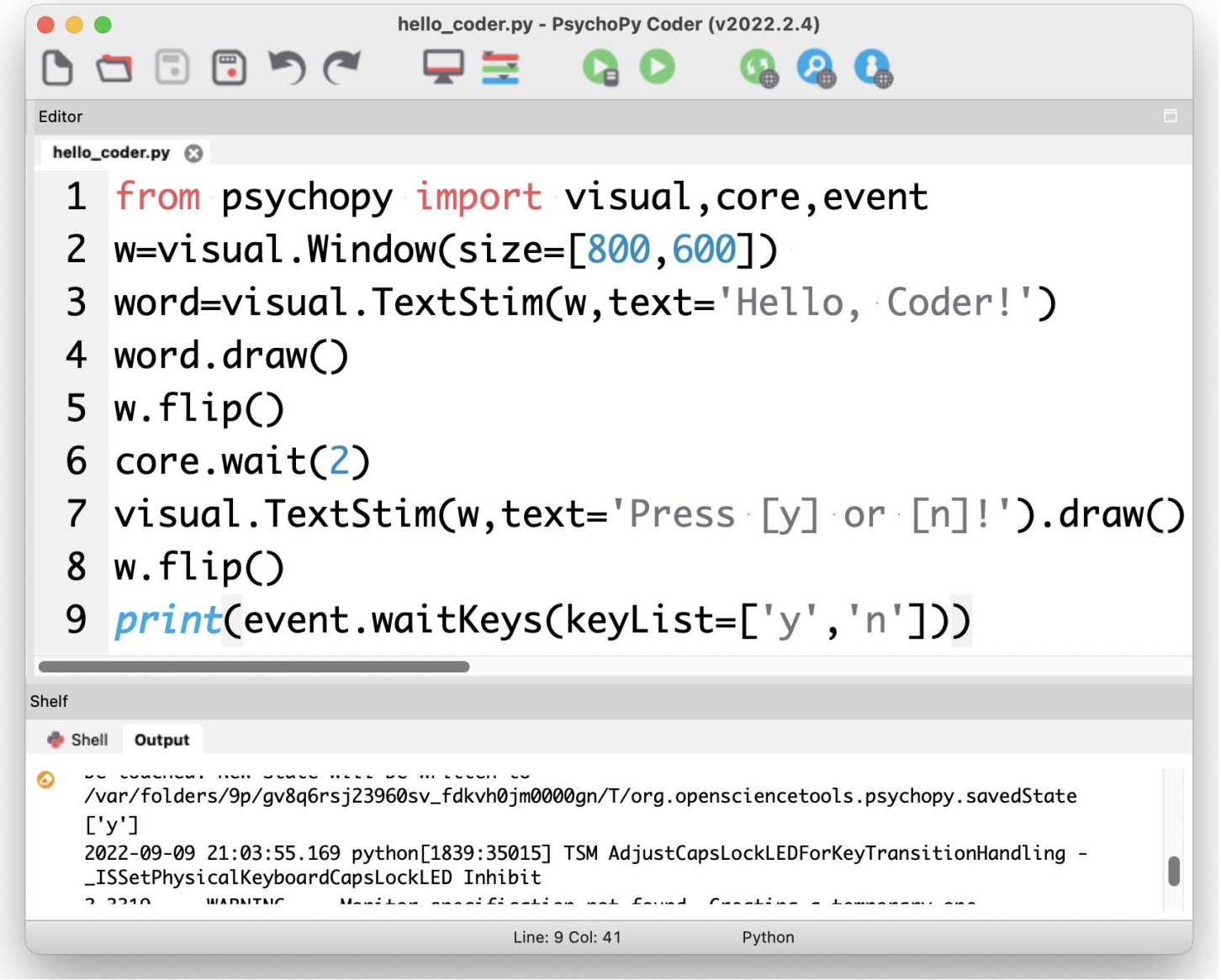
The *absolute* event timings are inconvenient for adjustments

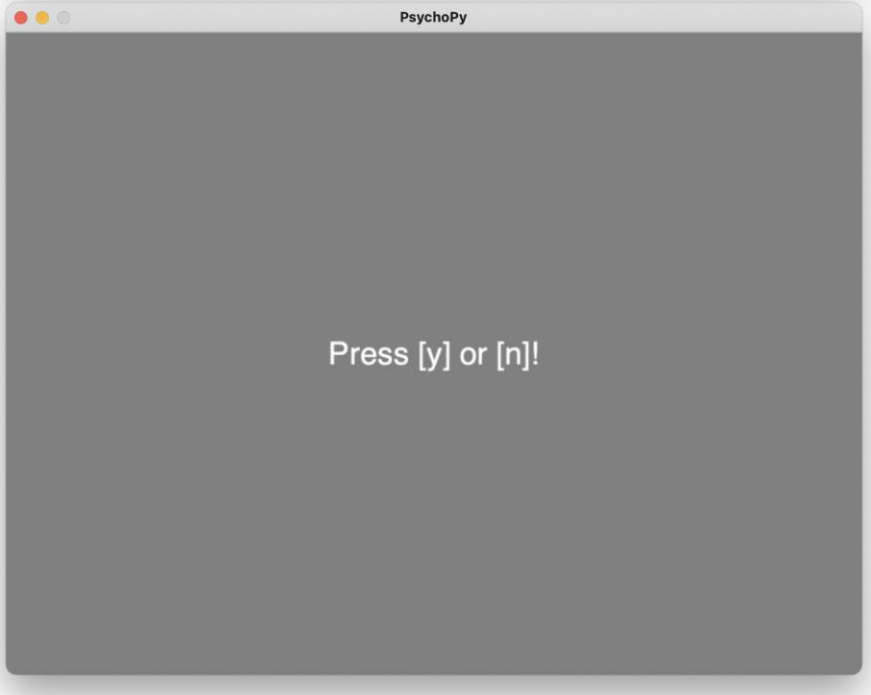
Difficult to develop more *complex* experiments



I have a solution!

Coder (hello_coder.py)





Coder (animal_detection_pic2.py)

/Users/tren/Library/CloudStorage/OneDrive-g.ntu.edu.tw/Teach/2022_心理與神經資訊學/02/02_Codes/local/animal_detection_pic2.py - PsychoPy Coder (v2022.2.4)

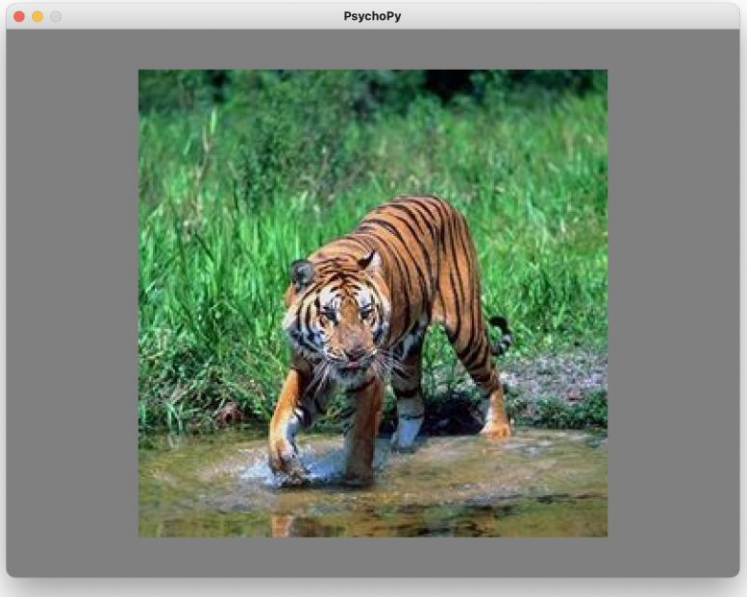
Editor

animal_detection_pic2.py

```
1  from psychopy import core,visual,event
2  import numpy as np, glob as gb
3  f=[gb.glob('T*.jpg'),gb.glob('D*.jpg')] #files
4  f=np.append(np.random.choice(f[0],3,False),np.random.choice(f[1],3,False))
5  corAns=np.array(['y']*3+['n']*3)
6  ACC=np.array([]); RT=[]
7  w=visual.Window(fullscr=True)
8  for i in range(1):
9    keys=np.array([])
10   trials=np.random.permutation(range(len(f)))
11   for j in trials:
12     core.wait(1)
13     visual.ImageStim(w,f[j],units='pix',size=[512,512]).draw()
14     w.flip(); core.wait(0.2); w.flip()
15     r=event.waitKeys(keyList=['y','n'],timeStamped=core.Clock())
16     keys=np.append(keys,r[0][0]); RT=np.append(RT,r[0][1])
17   ACC=np.append(ACC,keys==corAns[trials])
18 np.savetxt('data.txt',np.vstack([ACC,RT]).T,['%d','%f'])
```

Shelf

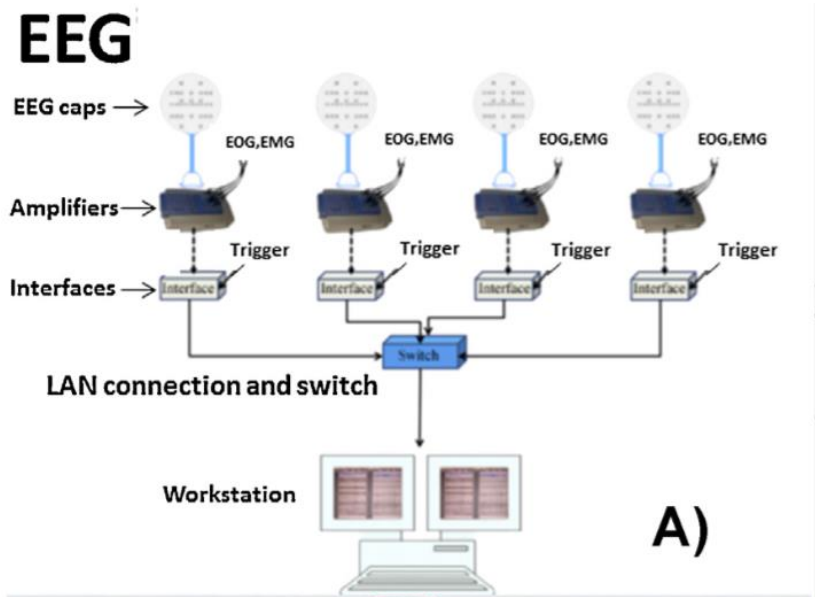
Line: 18 Col: 57 Python

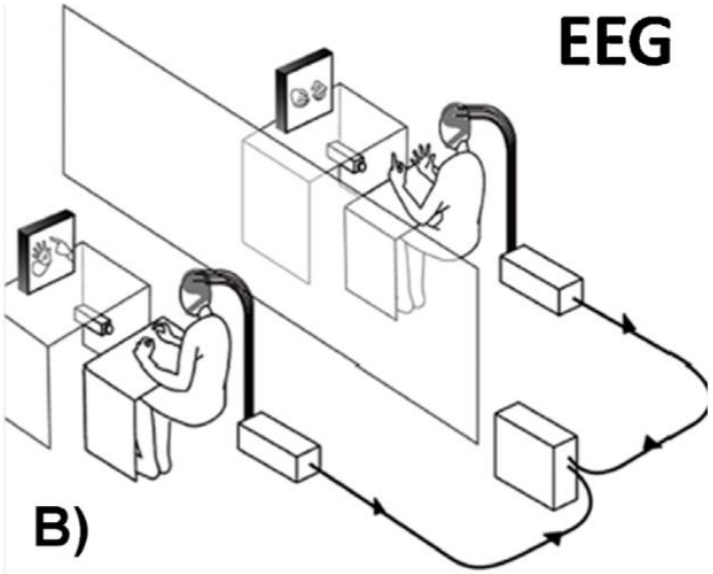


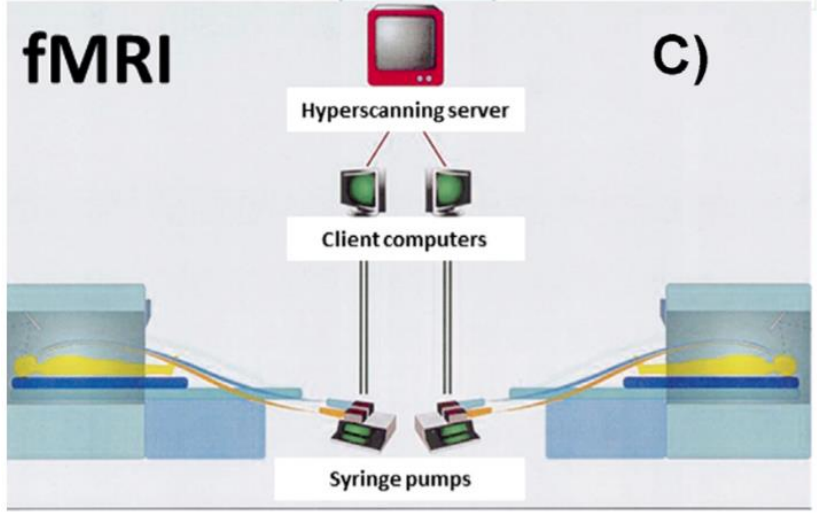
That’s all for local experiments!

Let’s move on to network ones!

Social (Neuroscience) Experiments

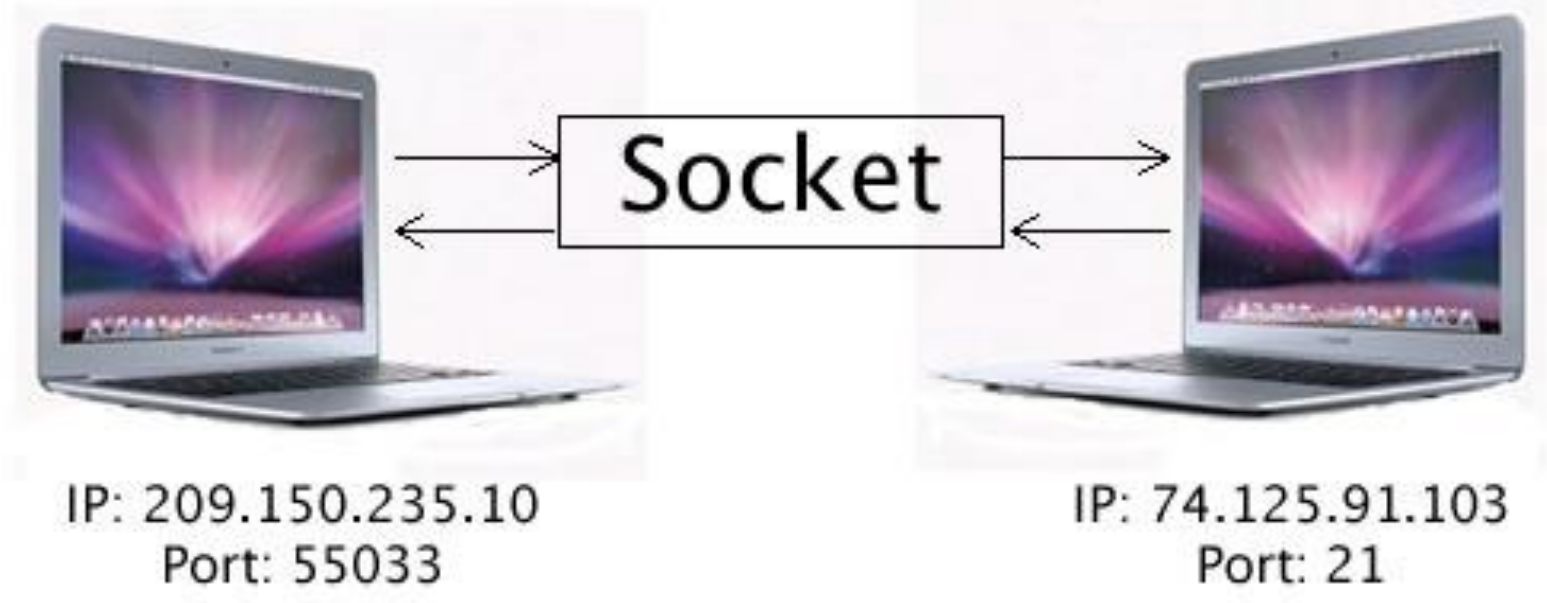






Socket Programming?

A network socket is a software structure within a network node of a computer network that serves as an endpoint for sending and receiving data across the network.



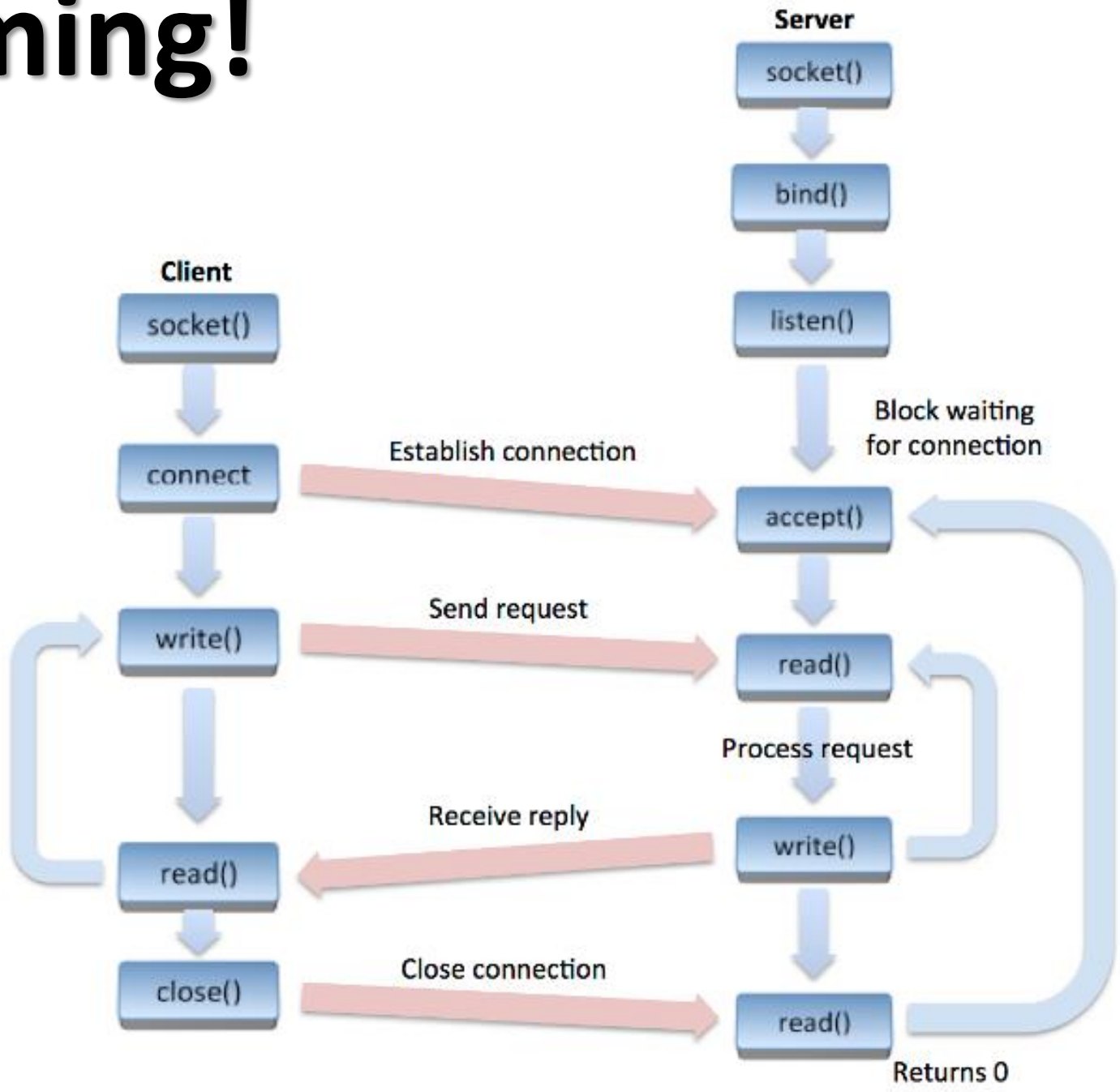
Socket Programming!

Clients are (often) active

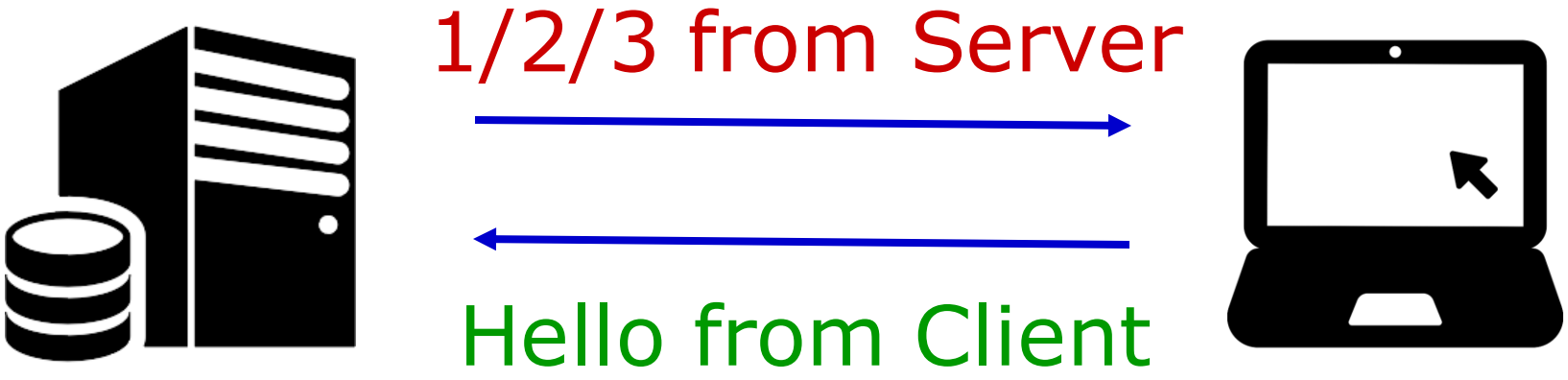
A client who makes requests whenever needed

Servers are (often) passive

A server keeps waiting for requests



Socket Programming 101 (client.py)




```
import socket
code='utf-8'
c=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
c.connect(('localhost', 1234)) # Server IP=localhost
c.sendall('Hello from client!'.encode(code))
print(c.recv(1024).decode(code)) # up to 1024 bytes
c.close()
```

Socket Programming 101 (server.py)

```
import socket
code='utf-8'
s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
s.setsockopt(socket.SOL_SOCKET,socket.SO_REUSEADDR,1)
s.bind(('localhost', 1234)) # Srvr IP=localhost, port =1234
s.listen(1) # max number of clients
s.settimeout(60) # max waiting time=60s
for i in range(3):
    (c,adr)=s.accept(); print('Client: ',i,adr)
    msg=c.recv(1024) # up to 1024 bytes
    print(msg.decode(code))
    c.sendall((str(i)+' from server!').encode(code))
s.close()
```

Animal Detection: client_pic.py

client_pic.py - PsychoPy Coder (v2022.2.4)

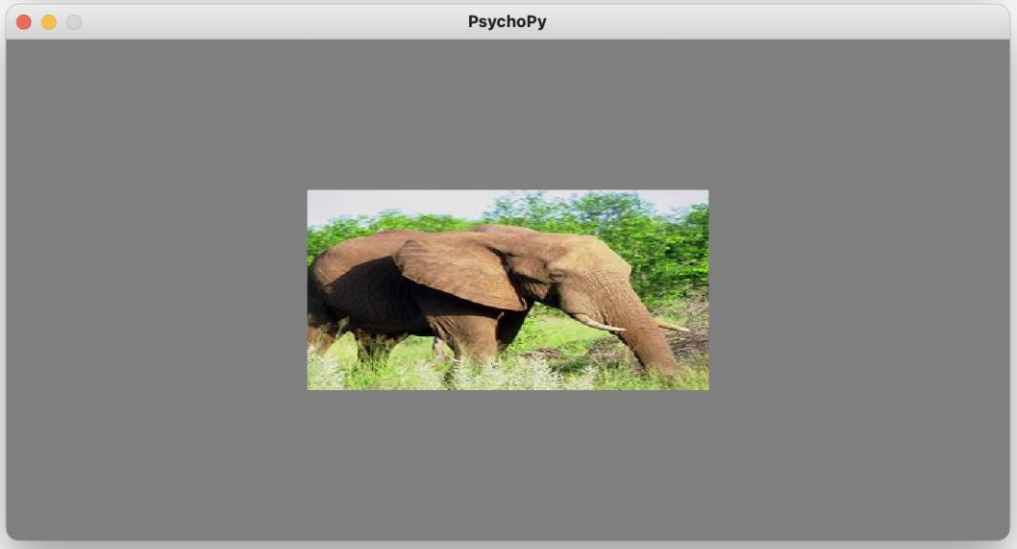
Editor

client_pic.py

```
1  from psychopy import core,visual,event
2  import numpy as np,glob as gb,socket
3
4  w=visual.Window(size=[800,400],units='norm')
5  visual.TextStim(w,text='Waiting for the server').draw()
6  w.flip(); code='utf-8'; nTrials=6; ACC=np.array([]);
7
8  c=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
9  c.connect(('localhost', 1241))
10
11 for i in range(nTrials):
12     msg=c.recv(6).decode(code); print(msg);
13     sz=int(msg[1:]);
14     with open('tmp.jpg','wb') as file:
15         file.write(c.recv(sz))
16     visual.ImageStim(w,'tmp.jpg',size=[0.8,0.8]).draw(); w.flip()
17     key=event.waitKeys(keyList=['y','n'])
18     print(key);w.flip()
19     ACC=np.append(ACC,key[0]==msg[0])
20     c.sendall(key[0].encode(code))
21
22 c.close()
23 print(np.mean(ACC))
```

Shelf

Line: 23 Col: 20 Python

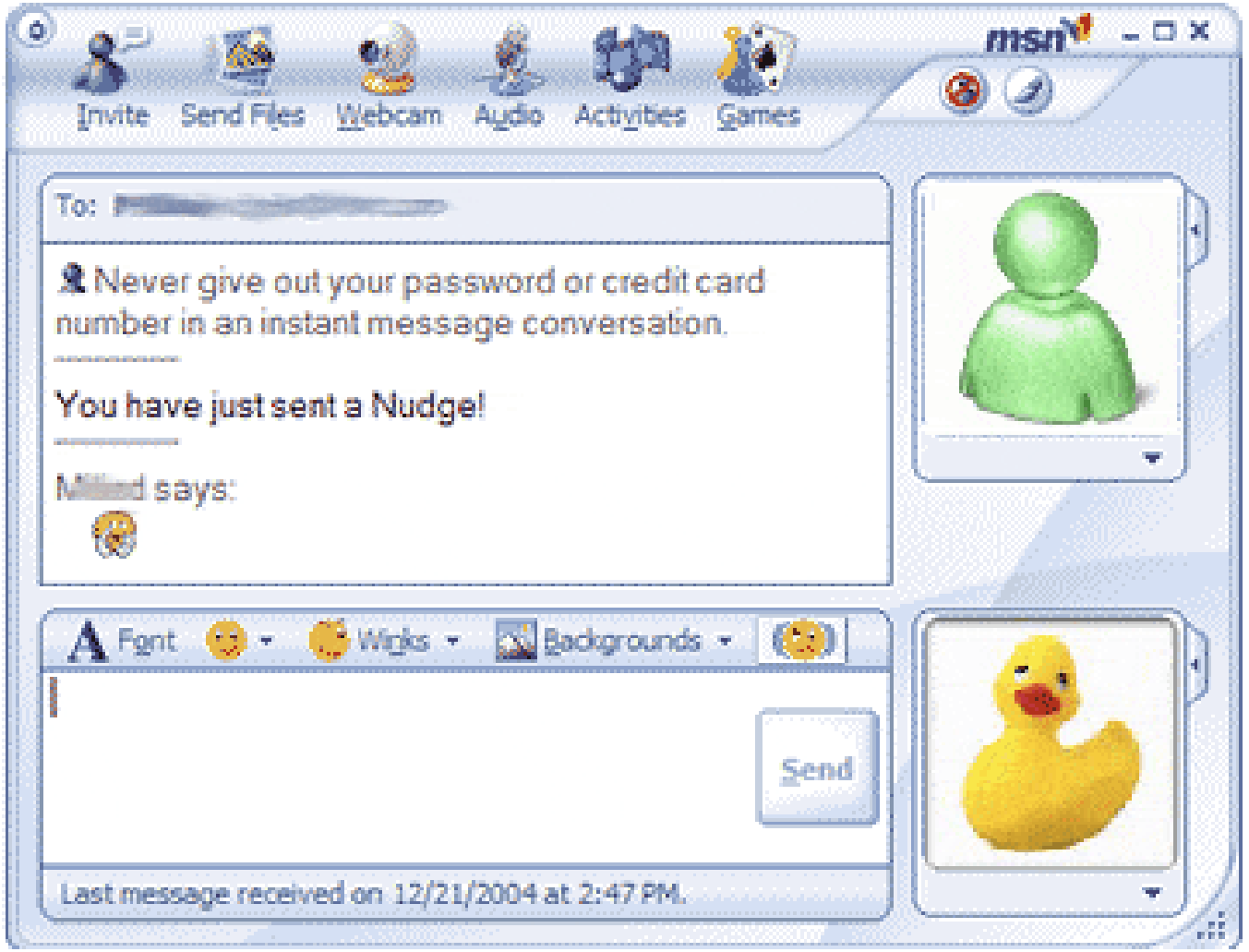


Animal Detection: server_pic.ipynb

```
import numpy as np,glob as gb,socket
from PIL import Image
from matplotlib.pyplot import *

code='utf-8'; nTrials=6
words=['elephant','fish','bird','road','houses','trees']
instruct='Press 1, 2, ..., 6 to pick a word:\n'
instruct=instruct+', '.join(words)
corAns=np.array(['y']*3+['n']*3)

print('Waiting for a client')
s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
s.setsockopt(socket.SOL_SOCKET,socket.SO_REUSEADDR,1)
s.bind(('localhost', 1241))
s.listen(1) # allow 1 client
(c,adr)=s.accept()

for t in range(nTrials):
  print(instruct)
  key=input('Server choice:')
  idx=int(key[0])-1 # choice as the array index
  with open(f[idx],'rb') as file:
    data=file.read() # read in pic file
  c.sendall((corAns[idx]+str(len(data))).encode(code)) # inform ans + pic size
  c.sendall(data) # send pic file to the client
  print('Waiting for the client')
  print(c.recv(1).decode(code)) # waiting for a response from the client

s.close()
```

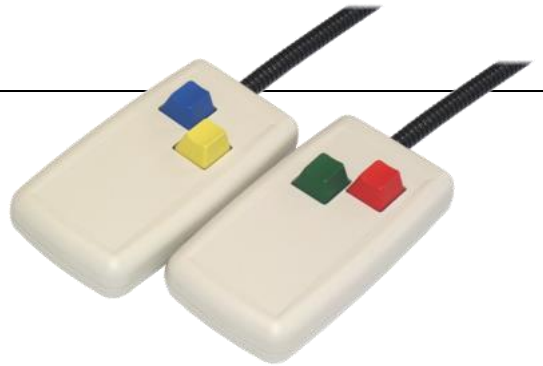
```
Waiting for a client
Press 1, 2, ..., 6 to pick a word:
elephant, fish, bird, road, houses, trees
Server choice:1
Waiting for the client
y
Press 1, 2, ..., 6 to pick a word:
elephant, fish, bird, road, houses, trees
Server choice:2
Waiting for the client

Press 1, 2, ..., 6 to pick a word:
elephant, fish, bird, road, houses, trees

Server choice:
```

You’re not far away from developing a messenger!



psychopy.hardware / pyserial

身體・心靈・文化 整合影像研究中心
IMAGING CENTER for Integrated
BODY, MIND AND CULTURE Research



```
from psychopy import core,visual,event
from psychopy.hardware.forp import ButtonBox
w=visual.Window(size=[400,300],units='norm',color=(-1,-1,-1))
BB=ButtonBox()
while(1):
  BB.getEvents(returnRaw=False,asKeys=True,allowRepeats=False)
  keys=event.getKeys()
  visual.TextStim(w,text=keys).draw(); w.flip();
  core.wait(0.5)
```

Homework for this week

1. Making a Stroop task using the server-supplied “word” dict.
2. Responding to the 8 trials of the task with an accuracy ≥ 85%

```python
from psychopy import core,visual,event
import socket,pickle

code='utf-8'
c=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
c.connect(('hpc.psy.ntu.edu.tw', 1234))
c.sendall('b87201025'.encode(code)) # Change your ID here!
print(c.recv(9).decode(code)) # Should show your ID!

w=visual.Window(size=[800,400],units='norm')
for t in range(8):
    sz=int(c.recv(2).decode(code)) # size
    word=pickle.loads(c.recv(sz))
    print(word)
    # Make your changes here (4 points for graphical displaying):
    #...
    #r=event.waitKeys(keyList=['r','g','b','y'],timeStamped=core.Clock())
    #c.sendall(pickle.dumps(r[0]))

#print(c.recv(5).decode(code)) # 4 points for passing
```

Game Over