Recap Day 2

Dict, Set not seq data type ∵ they are not ordered collect') like seq → str

- list
- tuple

Strings

seq of single characters

- it can hold nos also
- each stored as indiv characters
- """ ML strg """

↙

don't care ideally for

commenting

N = "Prime Input"

N[6::-1] = tiutnI

if -1 starts from end towards stop

- if -ve step, start from stop to start

→ + concatined

- No space inserted automatically unless you specify manually.

% with string = formatting

└ based on value

X.capitalize( )

index is used to handle exception in, try ~~again~~ except

as it throws error

for
else

else will be executed when 1st iteration itself fail

└ even if 1 round runs, else won't be executed

→ else is use to catch exception

list / tuple

ordered collection of heterogenous ~~dat~~ items

diff

mutable / im [ ] , ( )

list . append

list defining

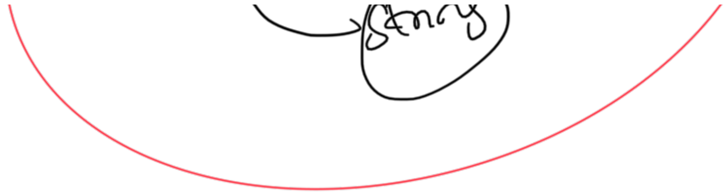
ls = [ ]

ls using append = list compren

@ anoth list .

ls2 = [i+1 for i in range(0,10)]

ls3 = [i+1 for i in l1]



pop returns & removes value @ particular index

it can be store

list.reverse()

sort then reverse - order

list

pop
remove #
~~take~~ n
& return.

clear vs del

var ✓ cleared ✗

↓
variable ✗
& list

remove

~~take value~~

just error

diff

Random

choose(seq) → random pick 1 items

Ex: roll a dice program

randrange(start, stop, step)

random → float.

seed

within random

shuffle

↳ randomise list.

while Vs for diff

for(condition) ✗ can't be put into while

~~i in rge(2,10)~~ / i in L2 ✗

while(cond) ✓ can be put into for

i < 10, → for i in rge(0,10)

for: iterable

while ⇔ (for const numb)

↳ we if condⁿ: