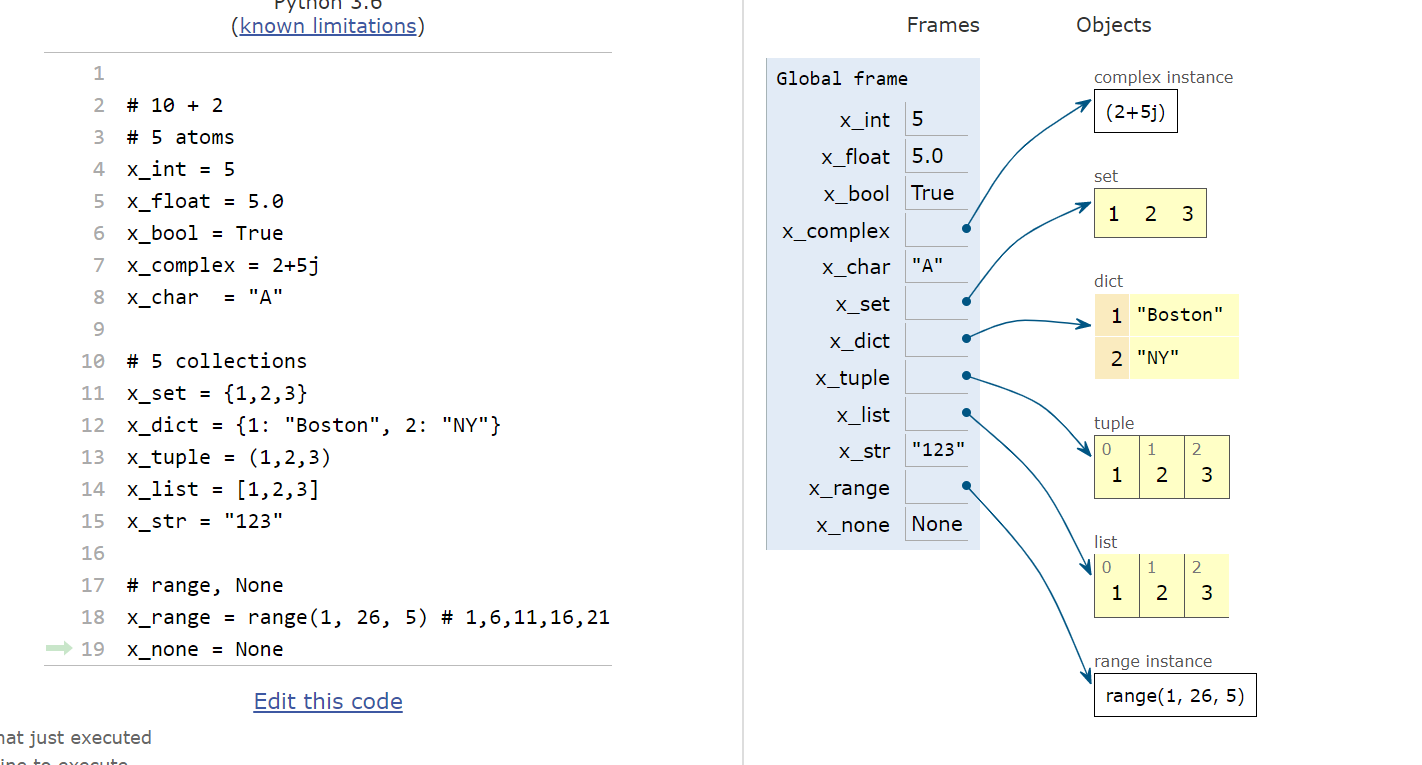
WELCOME TO CS-521!!!

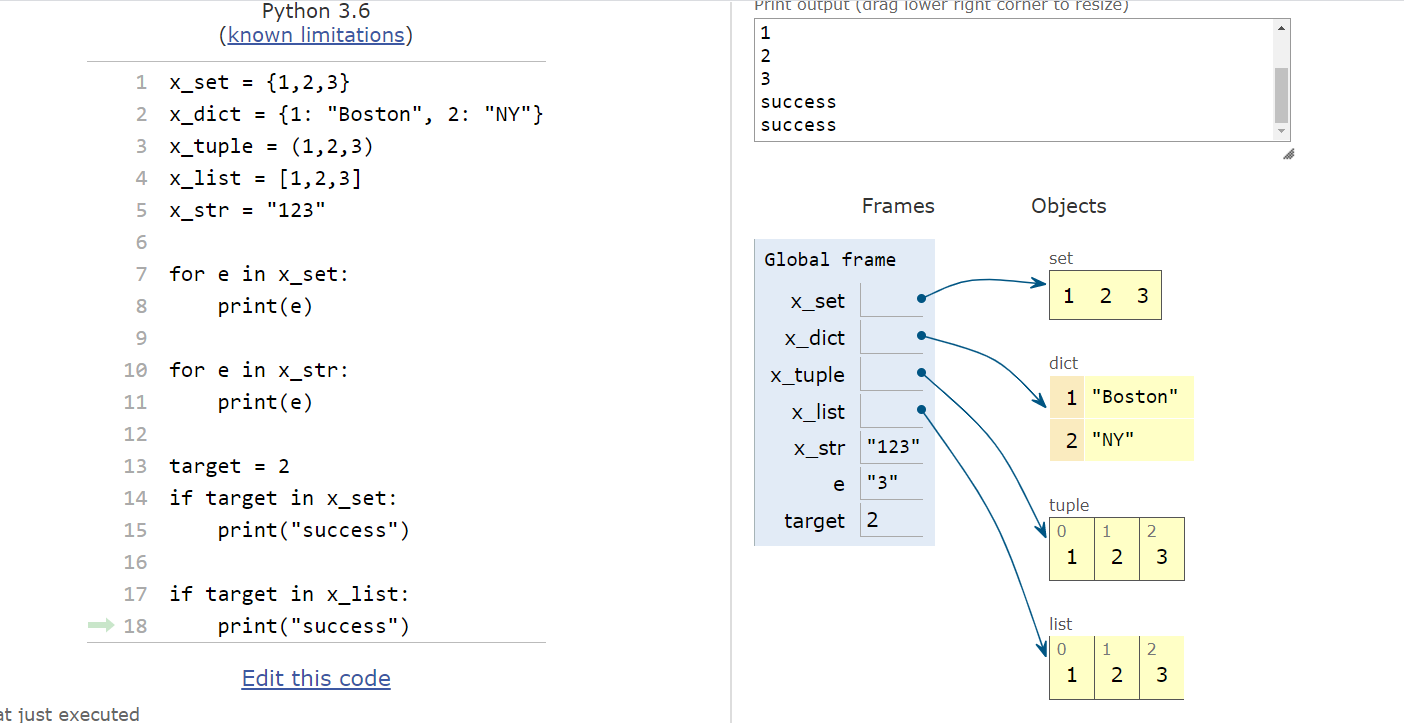
Tue - Thu, 6-7:30 p.m.

Lecture 4 (Nov 12, 2020)

Homework 2

Quiz 2



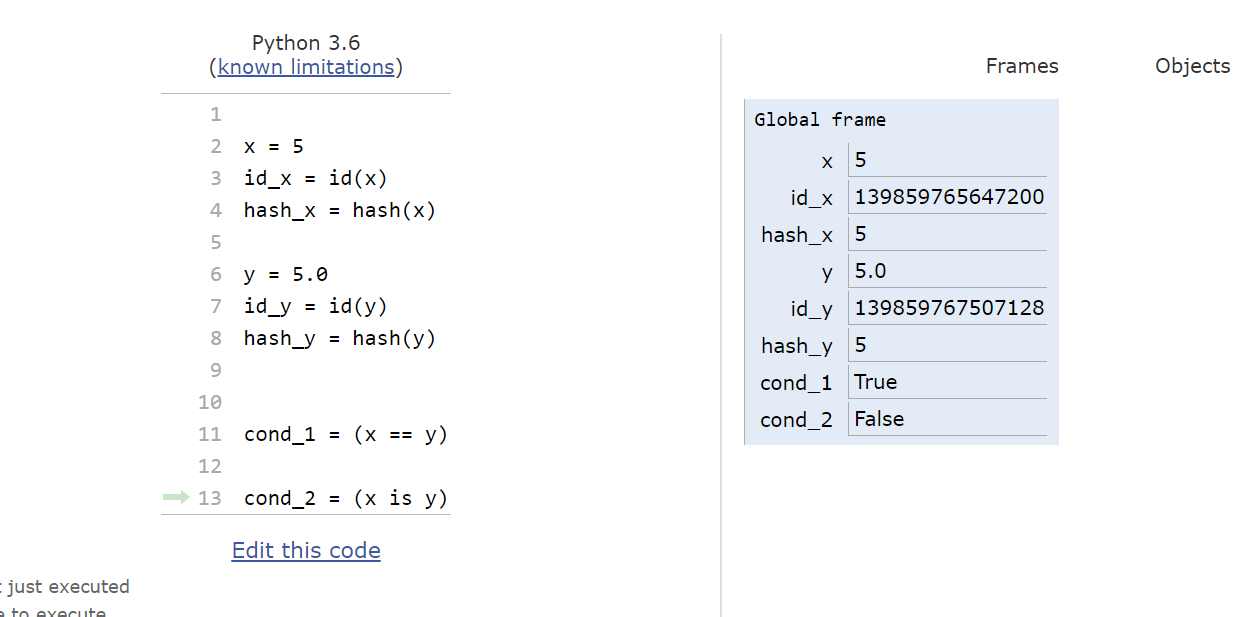


x = 5

y = 5.0

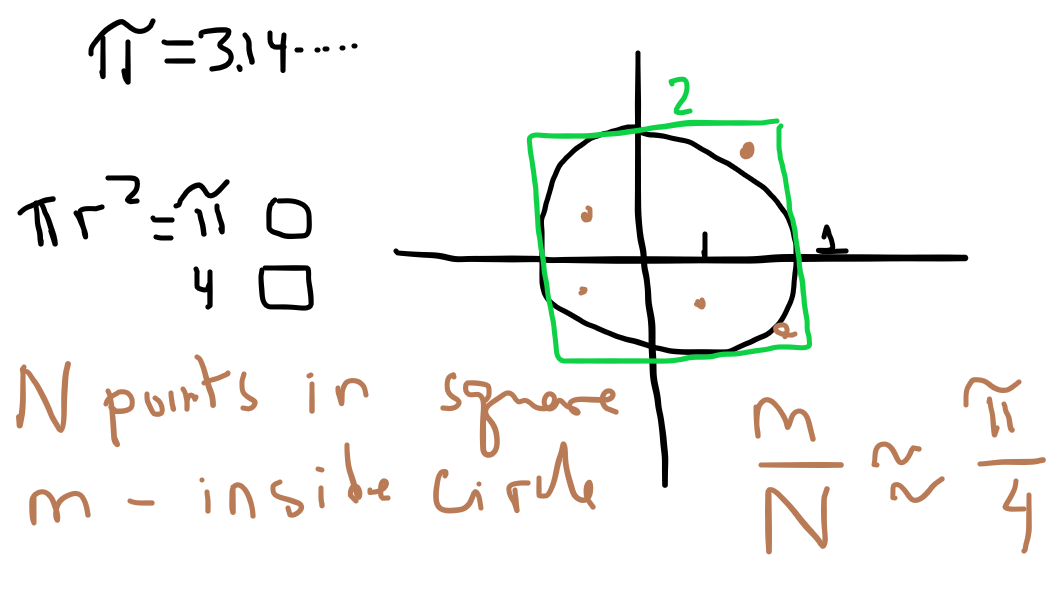
if x == y: compare values (hash) true

if x is y: compare addresses (id())

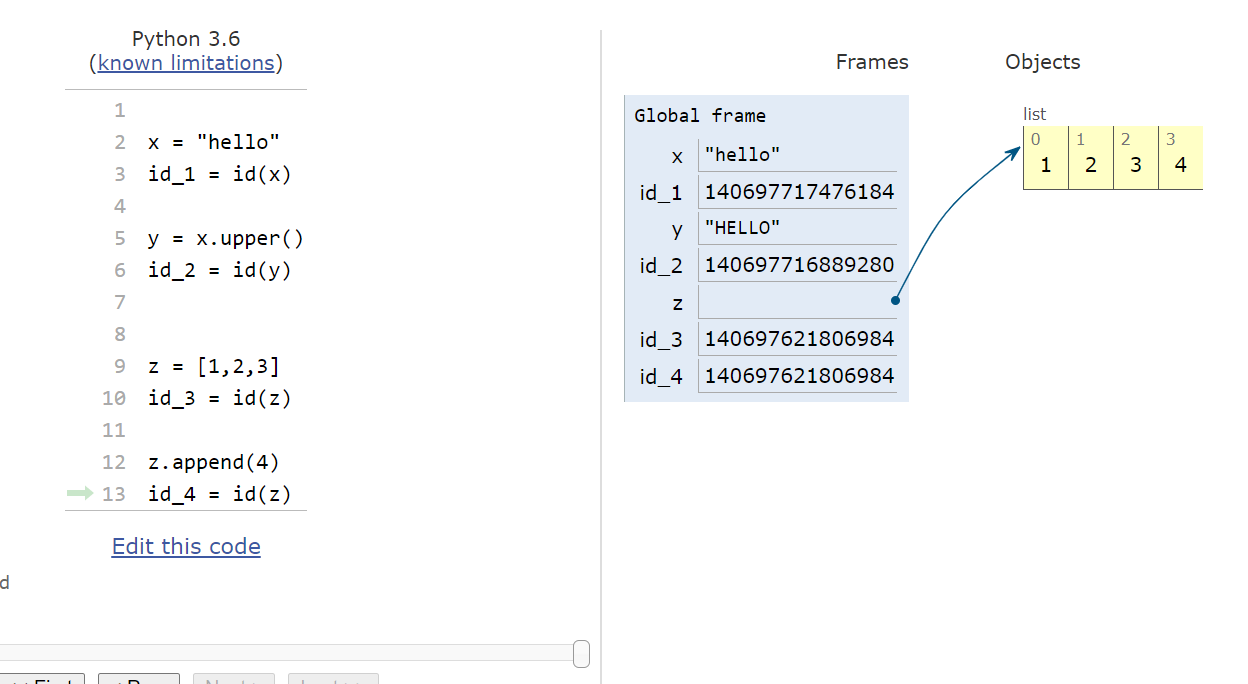


hash(object) -----------------🡪 int value

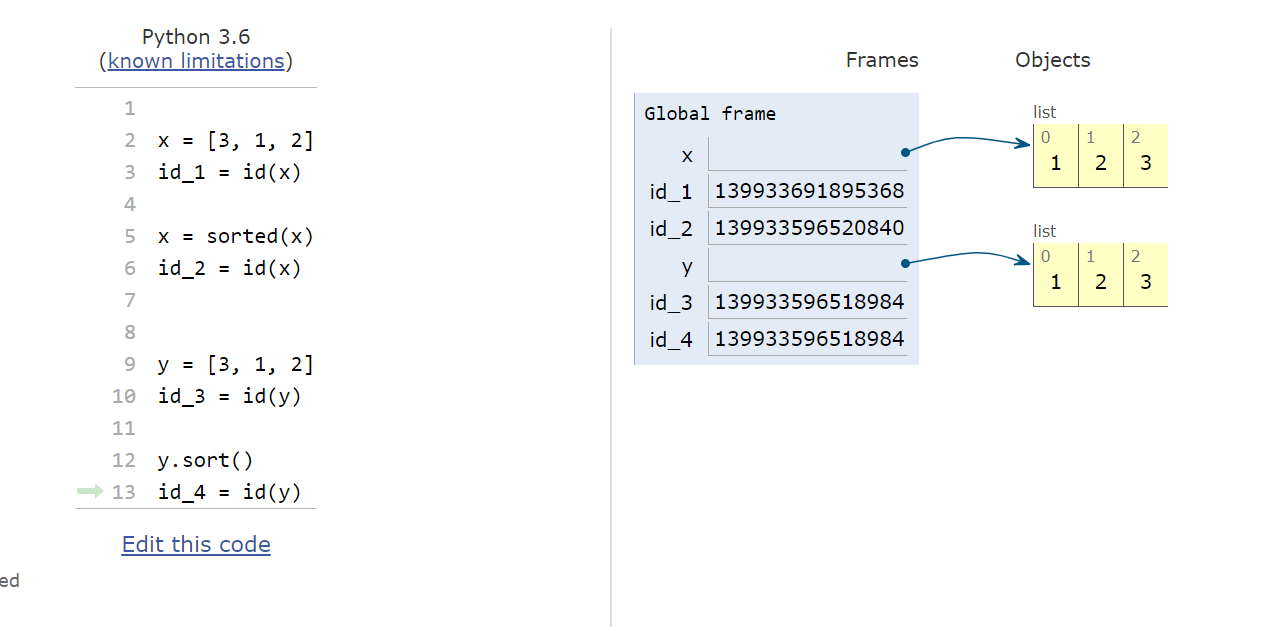




MUTABLE OBJECT: can change the composition of the object and the object “stays” in the same location (its id() does not change)



sort vs sorted



What python objects are mutable?

Why is it important?

* atoms (int, float, bool, complex, char)

int x;

x = 5;

x = 7;

x = 5

x = 7

ALL PRIMITIVE TYPES ARE IMMUTABLE!!!!

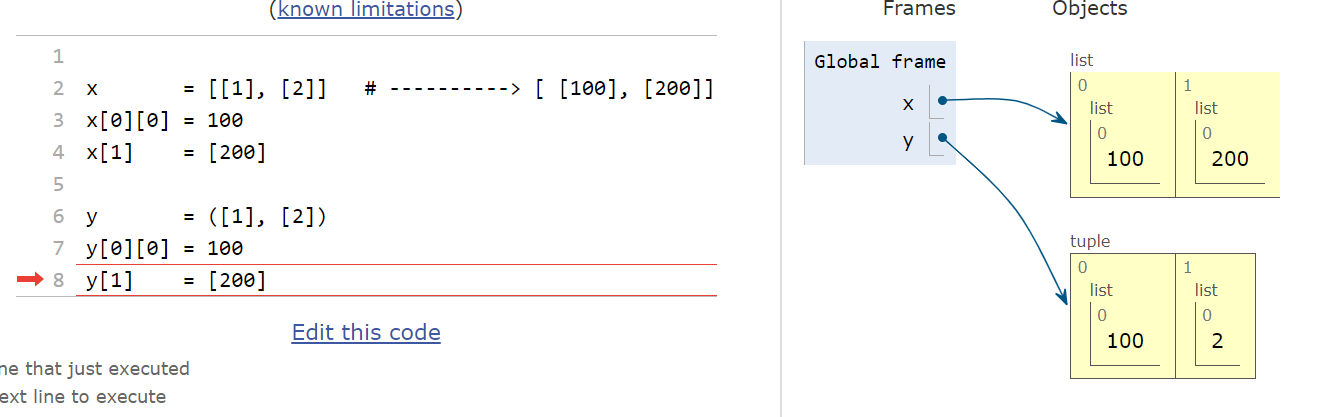
Collections:

1. strings are immutable
2. lists are mutable (they could contain any object within them
3. tuples
4. sets
5. dictionaries

Tuples are immutable

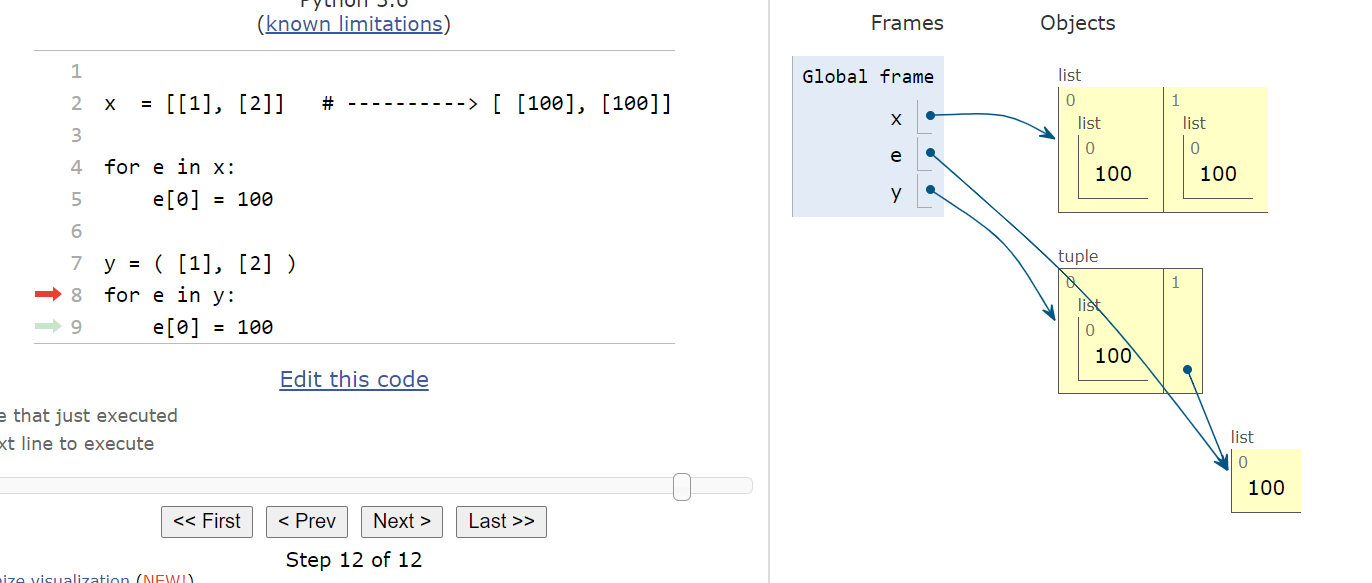


What is tuple contains a list



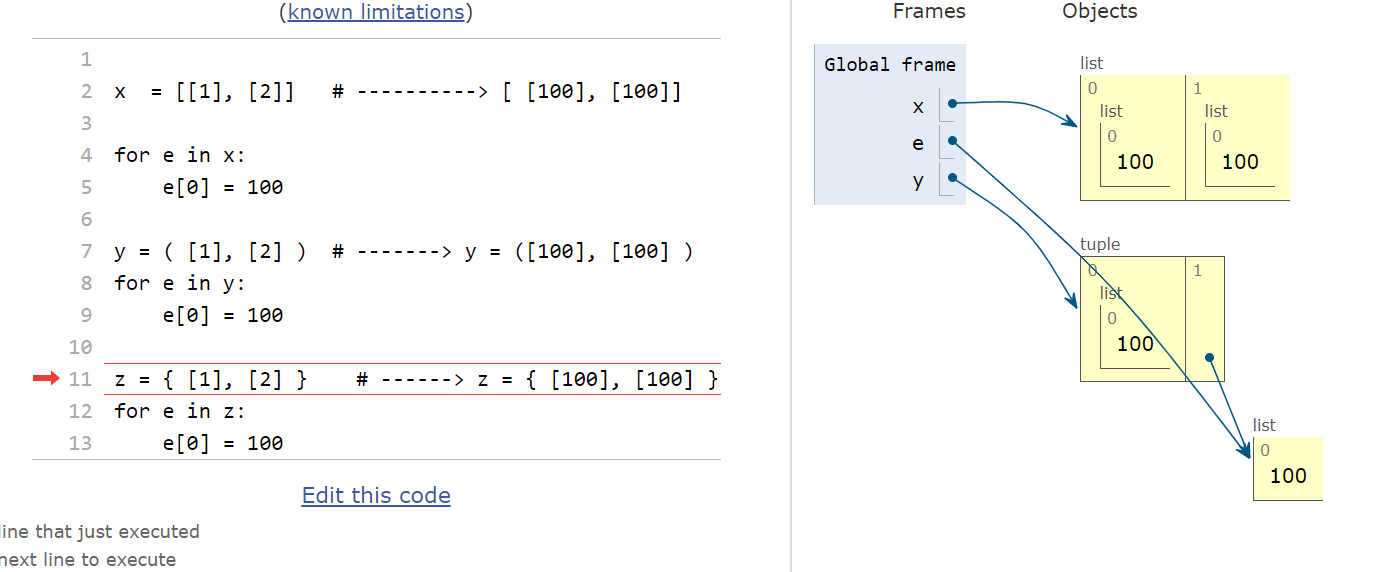
Note: tuples could contain any objects in Python

you can modify these objects “in-place”



if a collection contains mutable elements (e.g. lists) it is possibly to write an iteration so that these elements become identical

z = { [1], [2] }



in python, sets cannot contain any non-hashable elements

x = {1,2,3} is ok

x = { (1,2), (3,4) } ok

x = { [1], 3, 4} illegal

x = { (1, [2]), 3 } illegal

same restriction applied to dictionaries:

keys and values

values ---- any data type

keys: (same restrictions as sets)

x = { [1]: “boston”, [2] : “la”}

for next\_key in x.keys():

next\_key[0] = 100

x = { [100]: “boston”, [100] : “la”} illegal

Stay healthy and well