

Python Class 7

Classes and Loops

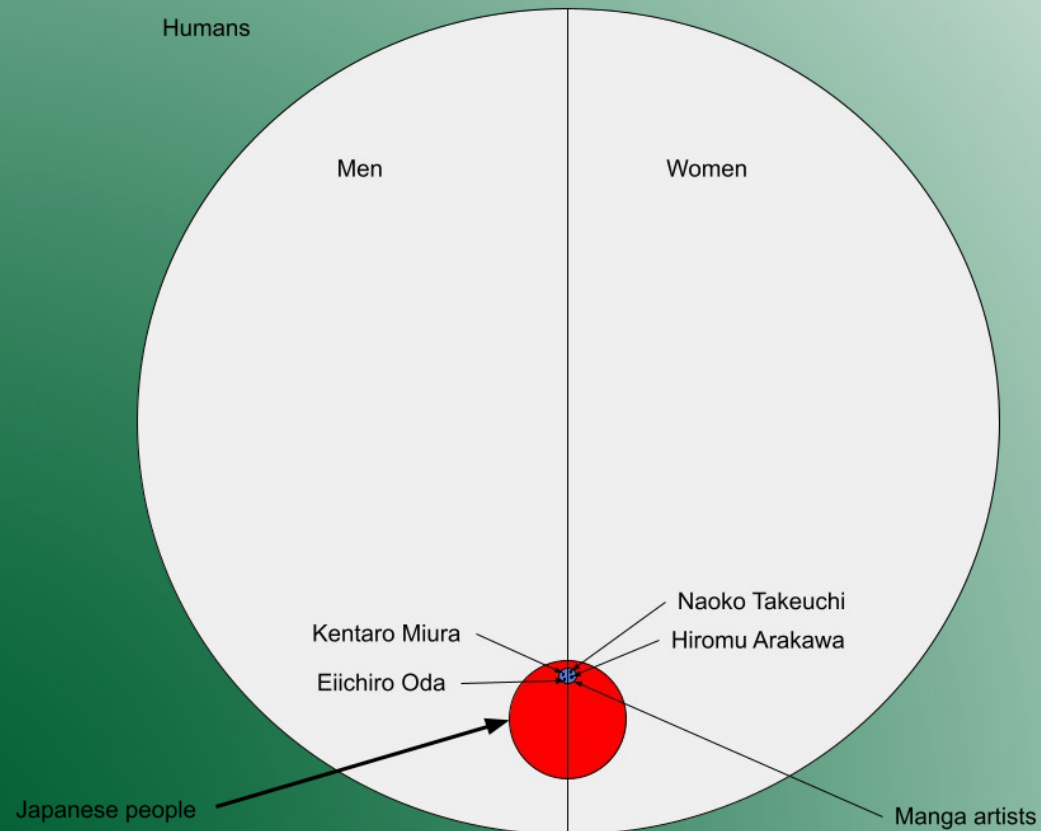
Classes (型 ; 種類)

- There are many different classes or types in Python.
- We have used lists, strings, ints, floats, bools, and dates.
- One of the most powerful abilities in programming is to create your own classes.

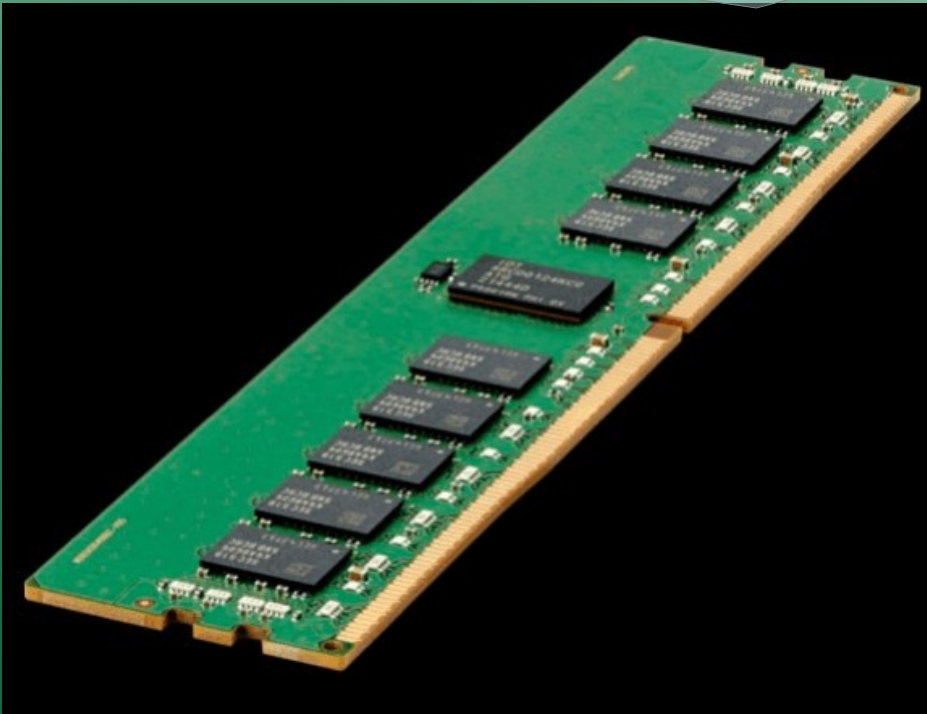
Instance versus Class

- There is an important difference between an instance and a class.
- The instance is the actual object (実体; 实例)
- The class is the group (種類).
- Humans, Japanese people, and Manga artists are all groups or classes.
- Kentarou Miura, Eiichiro Oda, Naoko Takeuchi, and Hiromu Arakawa are instances of each class.

Instance versus class



Instance versus Class 2



- An instance and a class both have an address in your computer's memory.
- This memory location (メモリアドレス) is NOT the same!
- You can test this using `id(instance)` and `id(class)`.
- You can also test this using a built-in class, like `str`.

variable = variable

- Another way to test this is by setting a variable equal to another variable.
- If you check the id, the address will be the same.
- Changing one variable's value will change the value for both.

Class Syntax; How to make a class

All classes have the `__init__` function. This is used to create an instance of the class. We also use this to create class variables.

```
class My_Class:
    def __init__(self):
        self.name = ""
        self.number_of_students = 0
```

The class keyword (予約語)

The “self” keyword. Self (自己) means the instance of the class.
We need to use the self keyword to create variables for the class.

Create a class instance (^{じったい}実体; ^{じつれい}実例)

- We use the name of the class to create an instance.

```
mc = My_Class()  
mc.name = "POP"  
mc.number_of_students = 4
```



Remember – the dot allows us to access variables inside the instance of a class

Review

- We have seen this a few times before.

```
my_birthday = datetime.date(1685, 3, 31)
```

The “self” keyword is always hidden when we create a class instance.

The `__init__` function is also hidden.

In the example above, when we call `datetime.date(year, month, day)`. This actually calls the `__init__` function.

Adding functions

```
class My_Class:
    def __init__(self):
        self.name = ""
        self.number_of_students = 0
        self.dates = []

    def AddClassDate(self, date):
        self.dates.append(date)

    def RemoveClassDate(self, date):
        self.dates.remove(date)
```

Adding a product class to our Python project



- What variables will be important for a product?
- What functions will we need?

*Use classes in our vending machine to
create different products*



Loop Practice

- Write a loop that adds all the numbers from 1 to 100.

Scope (ゆうこうはんい 有効範囲), *Again*

- Any variables you create inside a loop can only be used inside a loop.
- If you want to use a variable outside a loop, you need to create the variable outside the loop.

Gauss



- According to a story, he did something bad in class.
- His teacher made him add all the numbers from 1 to 100.
- Gauss did this in a few seconds!

How?

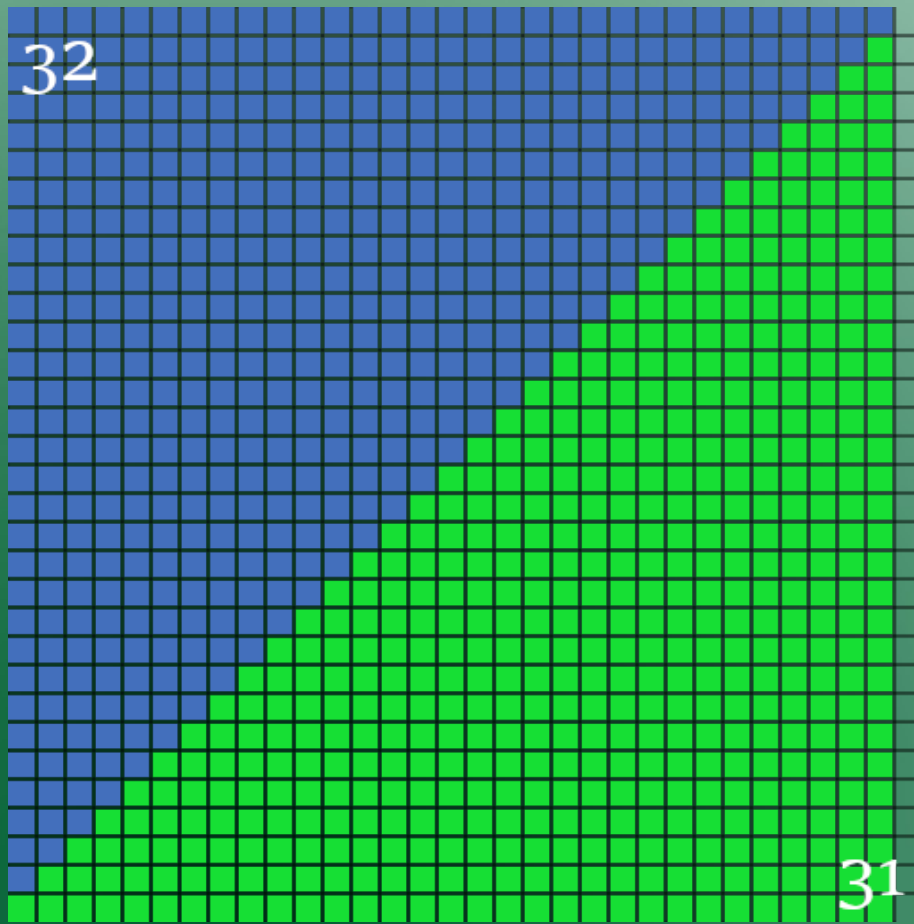


Way 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	1
1	2	3	4	5	6	7	8	9	10	11	12	13	14	31
32	32	32	32	32	32	32	32	32	32	32	32	32	32	32

- Line up the numbers forwards and backwards.
- Add the columns. This sum is always the same.
- Multiply the sum by the number of rows, divide by 2.

Way 2



- Imagine the numbers from 1 to 31 as squares.
- Copy this and reverse it.
- Now, you have a rectangle with 2x the number of squares.
- To find the area of a rectangle, multiply the height by the width.
- Divide this by two.

Loop exercises, continued

- Create a function with a loop that adds the first ten numbers.
- Create a function with a loop that adds the first num numbers.

Loop Guided Practice

- Write code that adds all the items in a list.
- `my_list = [56,37,82,2340,5]`

Loop Guided Practice 2

- Write code that multiplies all the items in a list.
- `my_list = [56,37,82,2340,5]`

Loop and If Exercise 1

- Write a program that finds the biggest number in a list.
- `my_list = [56,37,82,2340,5]`

Loop and If Exercise 2

- Write a program that finds the smallest number in a list.
- `my_list = [56,37,82,2340,5]`

Next Time!

- Option 1: Use loops, if-statements, and classes to create your own small program.
 - EX: Elevator, ATM, Train Ticket Machine, Text Game
- Option 2: Create a To-Do list program.