Python Class 9

Making a recipe class, a todo list class, saving data

```python
def add(a,b):
    return a + b

def square(a):
    return a * a

def half(a):
    return a/2

x = add(5,10)
print(x)

x = add(5,square(7))
print(x)

x = add(5,half(square(7)))
print(x)

x = square(half(add(50,60)))
print(x)
```

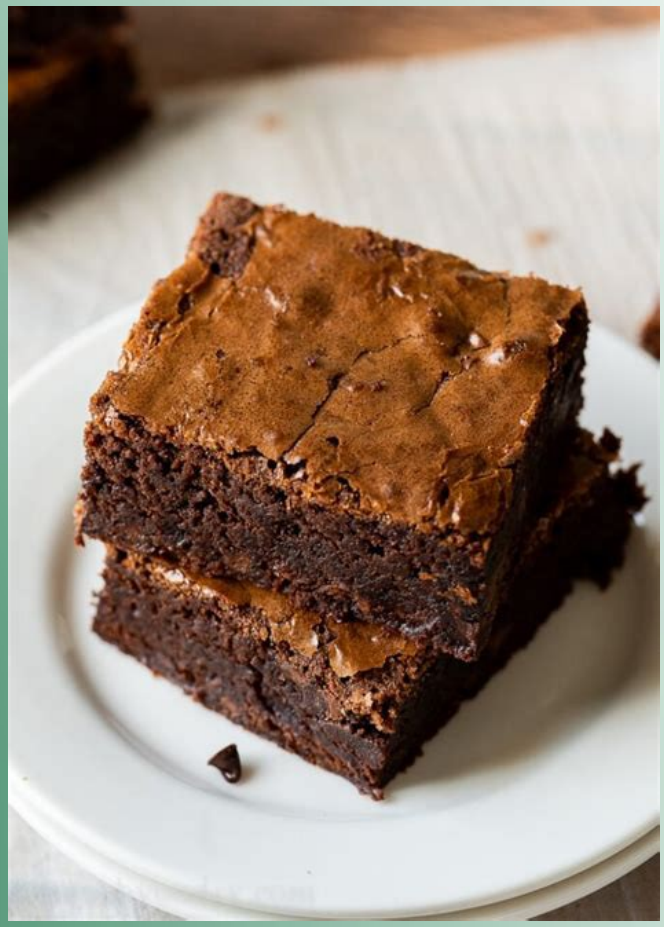
```python
def add(a,b):
    return a + b

def square(a):
    return a * a

def half(a):
    return a/2

x = add(5,10)
print(x)

x = add(5,square(7))
print(x)

x = add(5,half(square(7)))
print(x)

x = square(half(add(50,60)))
print(x)
```

```python
def add(a,b):
    return a + b

def square(a):
    return a * a

def half(a):
    return a/2

x = add(5,10)
print(x)

x = add(5,square(7))
print(x)

x = add(5,half(square(7)))
print(x)

x = square(half(add(50,60)))
print(x)
```

```python
def add(a,b):
    return a + b

def square(a):
    return a * a

def half(a):
    return a/2

x = add(5,10)
print(x)

x = add(5,square(7))
print(x)

x = add(5,half(square(7)))
print(x)

x = square(half(add(50,60)))
print(x)
```

Class Practice 2

- Make a recipe class.
- What variables do we need?
- What functions do we need?

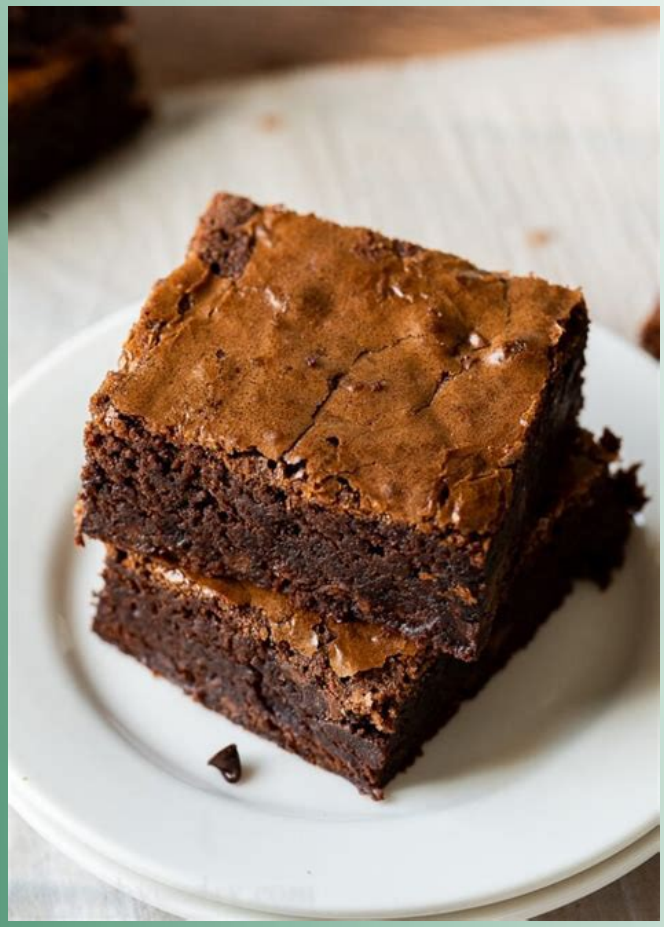


Using __repr__(self)

- __repr__() and __str__ are very similar functions.
- If you call print(obj), the computer will use obj.__str__()
- If you call print (objlist), the computer will use __repr__()
- They both take self and return a string version.

Class Practice 2

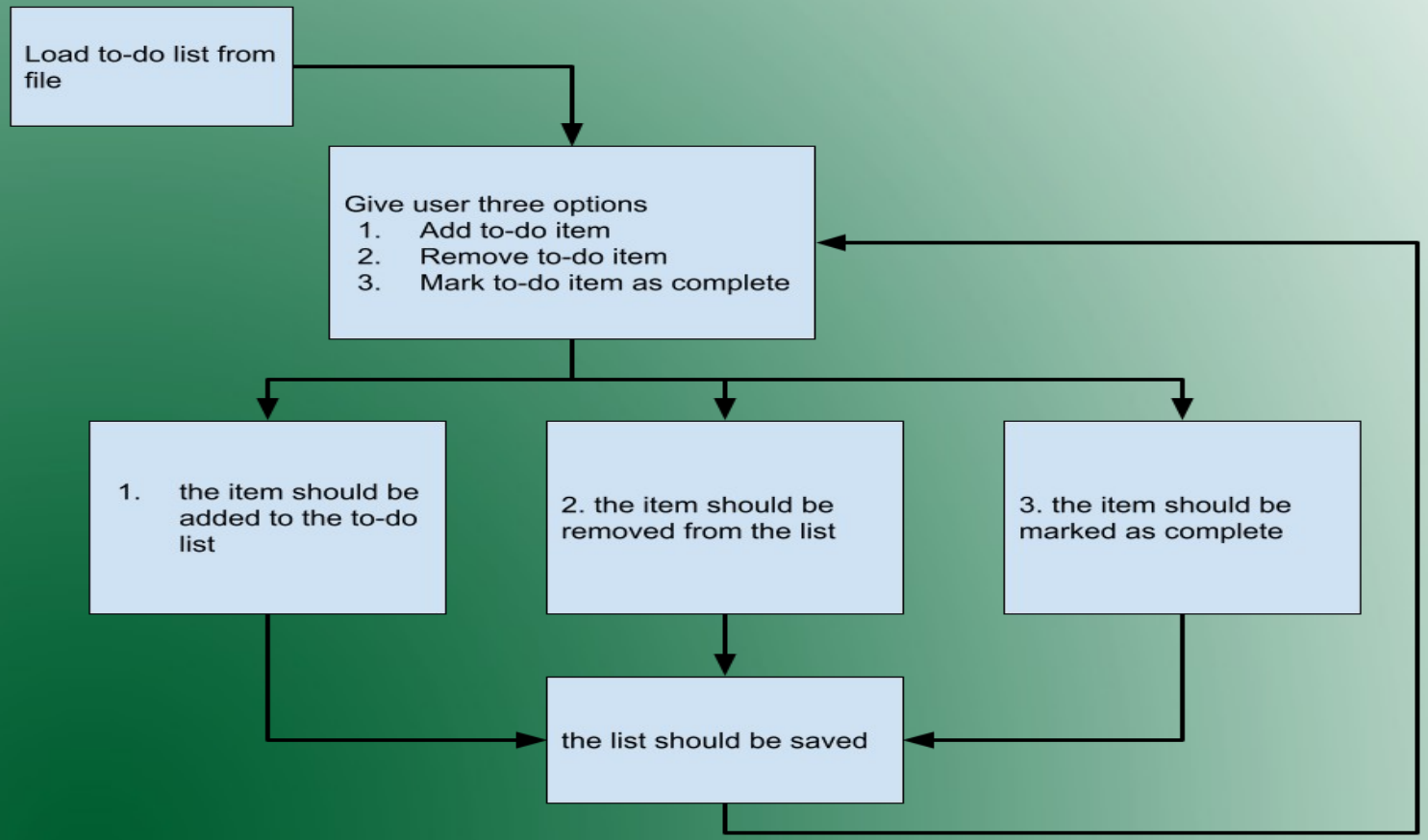
- Make a recipe class.
- What variables do we need?
- What functions do we need?



To-Do List Project

- How should a to-do list app work?
- What classes will we need to make?
- What variables will the classes need?
- What functions will the classes need?

How it should work



Open a file

- To open a file, we use the built-in function open()
- open() takes two arguments
  - *file*
  - *mode*

Open arguments

- open(***file***,mode)
  - file is the path and name of the file
  - **For example** *C:\Users\David Hunter\Documents\Seitoku Python Curriculum*
  - This is the path where I save my files for this class.
  - *C:\Users\David Hunter\Documents\Seitoku Python Curriculum/**2021 Python Class 10.odp***
  - The **BOLD** part is the file name.

Open arguments, 2

- open(file,***mode***)
  - mode is how you want to open the file
  - “r” -reads the file. Error if no file
  - “a” -appends. Creates the file if no file.
  - “w” - opens the file to write. Creates the file if no file
  - “x” - create the file. Error if the file exists.

Mode (ファイルの読み方)

- You can also say how the file should be read.
  - “b” is for binary (二進法 にしんほう) → People cannot read this format, but computers can.
  - “t” is for text

File variables

- file.closed – True or False
- file.mode- the mode
- file.name – the name
- ~~file.softspace – not really important~~

File functions

- file.close() - closes the file
- file.read() - this gives you the data/information in the file
- file.write(str) – this adds a string to the file
- file.writelines(sequence) – this adds a list of strings to the file

Practice

```
my_file = open(“Hello.txt”, “w”)
print(my_file.name)
print(my_file.mode)
#my_file.write(“Hello”)
my_file.close()
my_file = open(“Hello.txt”, “r”)
print(my_file.read())
```

Practice adding some information to a file

```
my_file = open(“Hello.txt”, “w”)
print(my_file.name)
print(my_file.mode)
my_file.write(“Hello”)
my_file.close()
my_file = open(“Hello.txt”, “r”)
print(my_file.read())
```

Practice reading information from a file

```
my_file = open(“Hello.txt”, “w”)
print(my_file.name)
print(my_file.mode)
my_file.write(“Hello”)
my_file.close()
my_file = open(“Hello.txt”, “r”)
data = my_file.read()
```

Using pickle to save data

- Pickle is a library, just like datetime or random.
- Pickle is used for saving data.
- It is especially useful for saving complicated data, such as class instances, lists, etc.

How to use

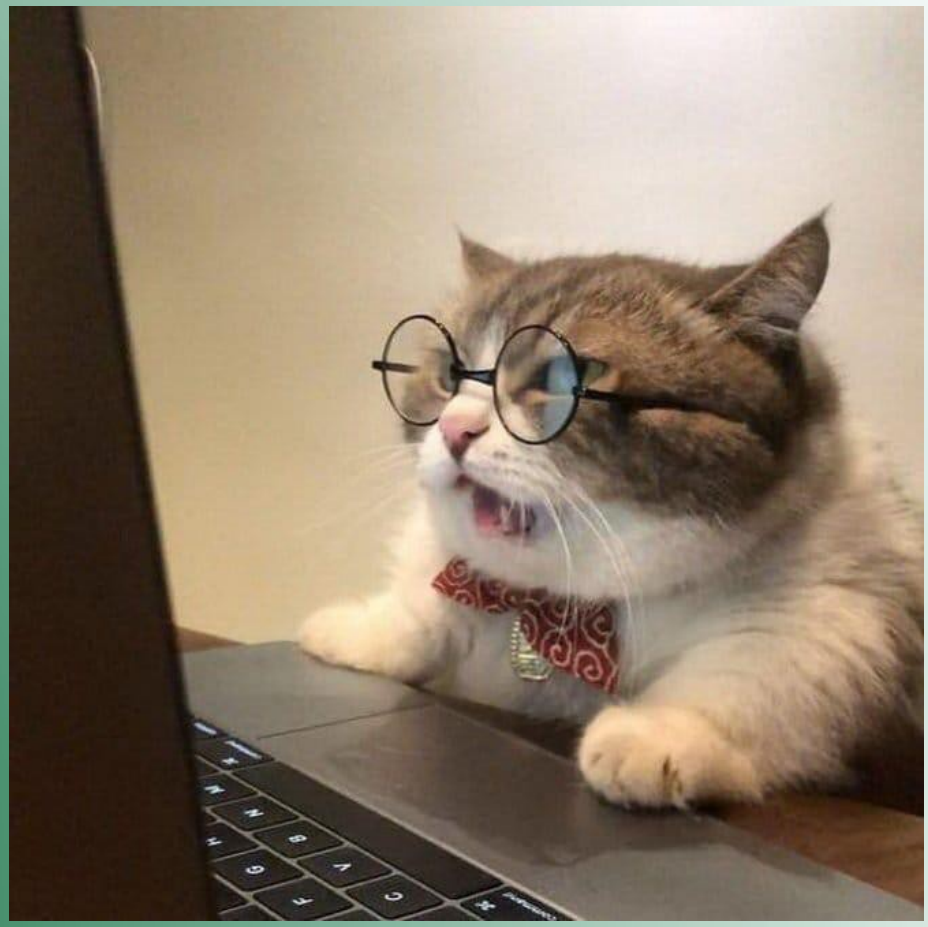
- Pickle has two main functions we will use
  - pickle.dump(obj,file)
    - obj is the information we want to save
    - file is a file that we opened
  - pickle.load(file)
    - this reads the data from the file and returns the data

Example

```
import pickle
import datetime

class Book:
    def __init__(self, n= "", a = "",d = datetime.date.today()):
        self.name = n
        self.author = a
        self.publish_date = d

    def __str__(self):
        return self.name + " by " + self.author + \
               " (" + str(self.publish_date.year) + ")"

book = Book("The Malazan Book of the Fallen", \
            "Steven Erikson", datetime.date(2001,1,1))


file = open("test","wb")
pickle.dump(book,file)

file.close()

infile = open("test","rb")

correct = pickle.load(infile)
infile.close()

print(correct)
```

Practice!!!

- Pick one of the projects we made in previous classes: DNA, recipe, manga-ka, vending machine, to-do list.
- Use pickle to save the data to a file.
- Use pickle to load the data.



DNA and RNA

- Why DNA/RNA?
  - All living things have DNA.
  - Even viruses, like covid-19, use DNA or RNA to store their genetic information(遺伝子情報).

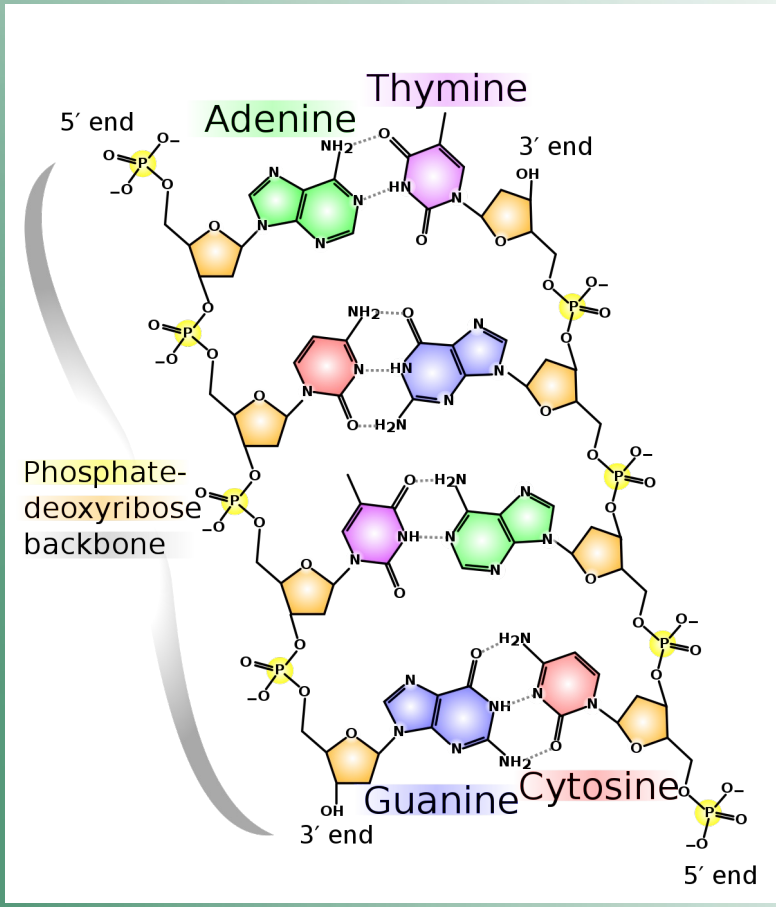
The Goal

- Make a DNA class with a list of genes
- Make 3 different DNA instances
  - Influenza has 8 genes
  - MERS has 11 genes
  - Covid-19 has 9 genes
- Let's compare them!

DNA Information

- DNA controls genes (遺伝子).
- Genes create different proteins (酵素/ タンパク質).

DNA Information 2

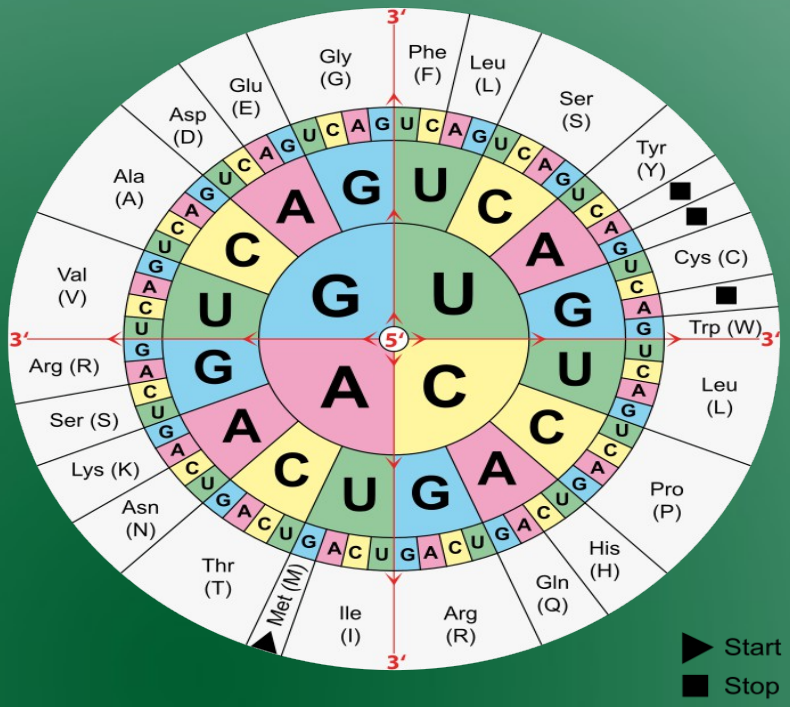


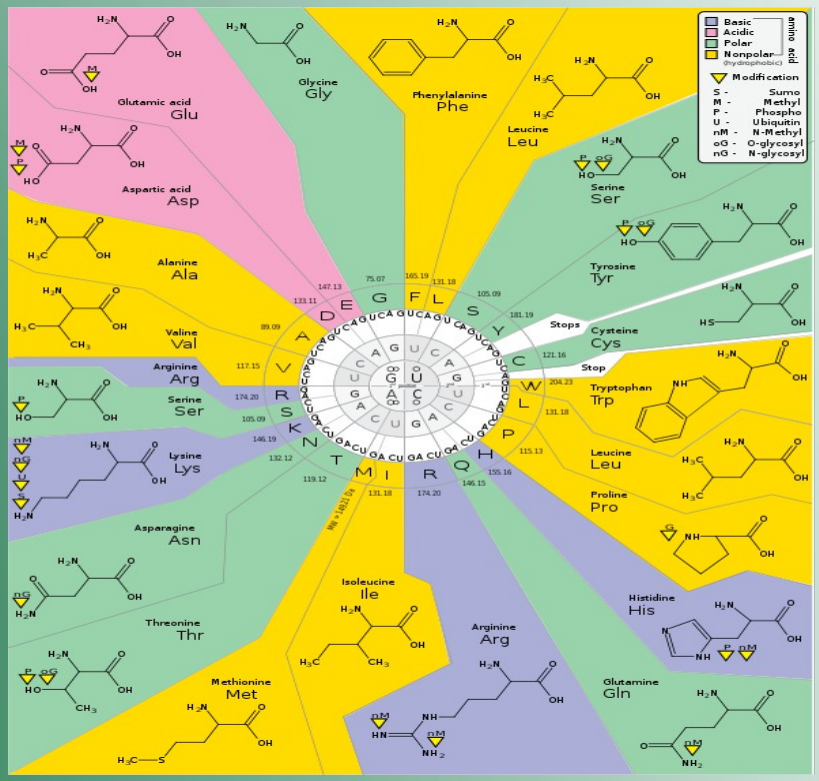
DNA Information 3

- DNA has different chemicals in groups of three.
- These chemicals are
  - Adenine
  - Thymine
  - Guanine
  - Cytosine

DNA, 4

- Each group of three chemicals creates a part of a protein.

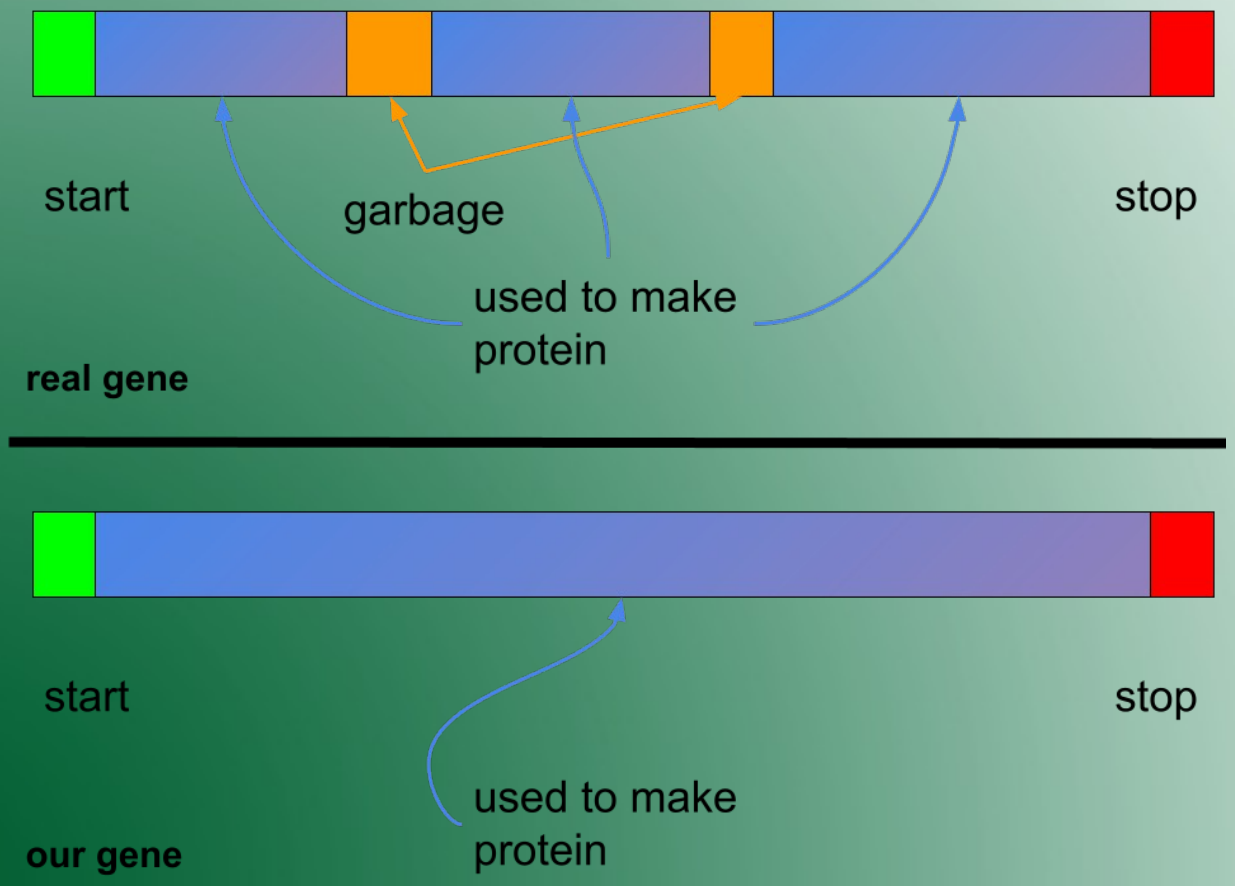




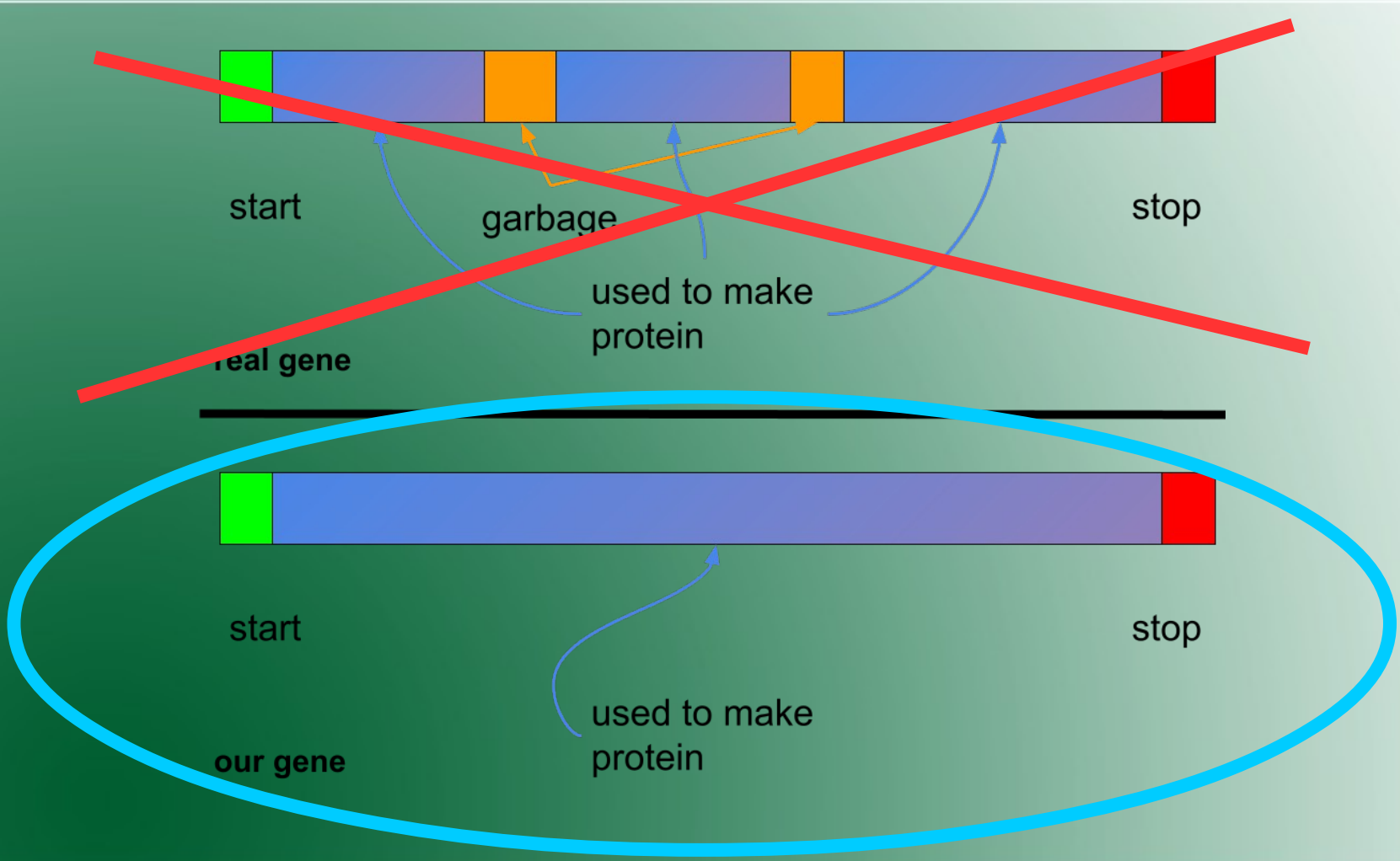
DNA 5

- Every gene starts with a special group of three
  - ATG (Adenine Thymine Guanine)
- There are three ways to stop a gene
  - TAG (Thymine Adenine Guanine)
  - TGA (Thymine Guanine Adenine)
  - TAA (Thymine Adenine Adenine)

Gene Structure



Gene Structure

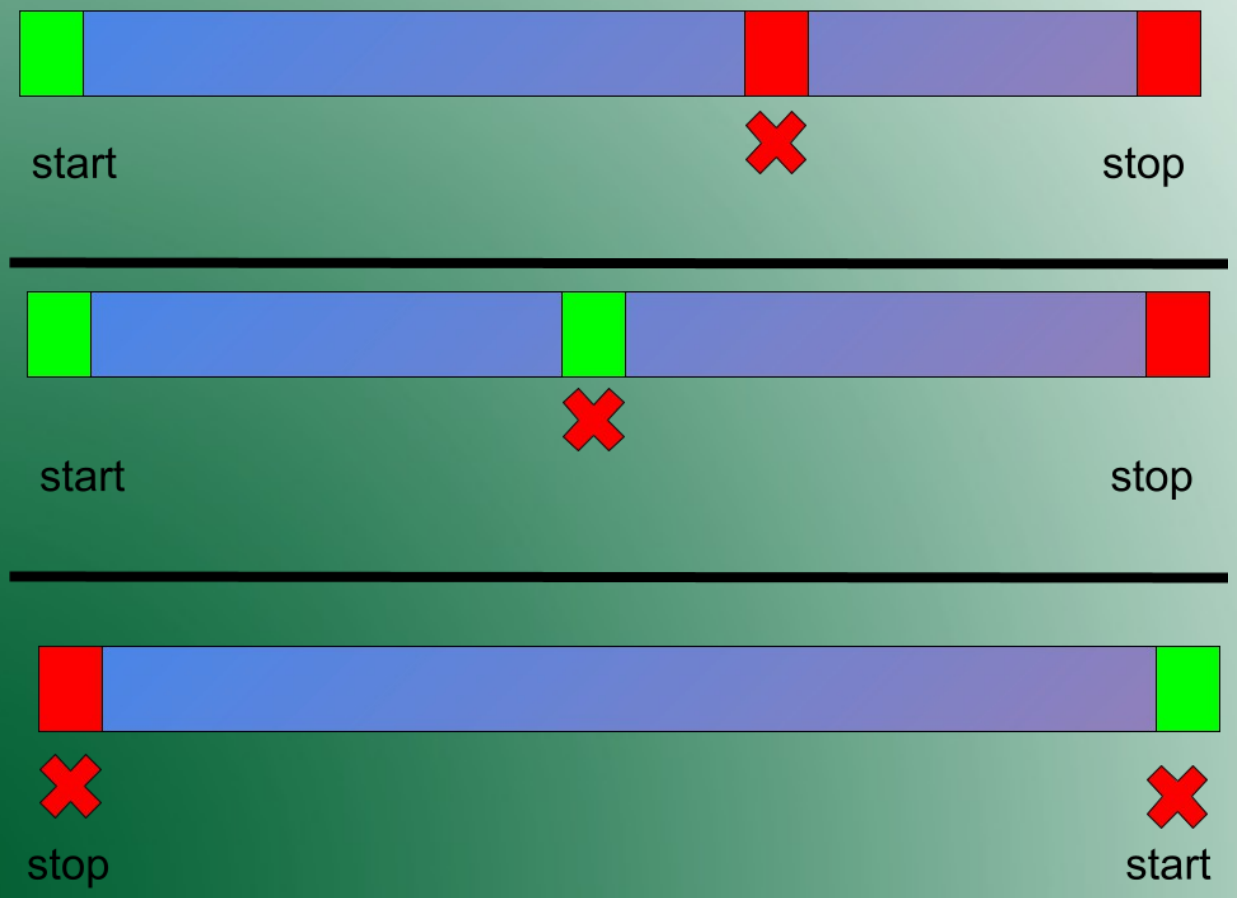


DNA exercise as list

- Let's write some code to make a random gene!

```
import random
#we will use random.randint(a,b)
chemicals = [“A”, “T”, “C” , “G”]
start = “ATG”
end = [“TAG”, “TGA” , “TAA”]
```

Rule for making codons



Let's change our code to a gene class

- What variables will it need?
- What functions will our gene class need?