Day 5: Dictionaries, Apply What you’ve learned in Tournament

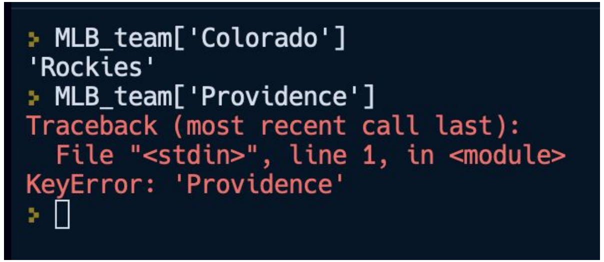
July 14th 2023

Previous Day Clarifications

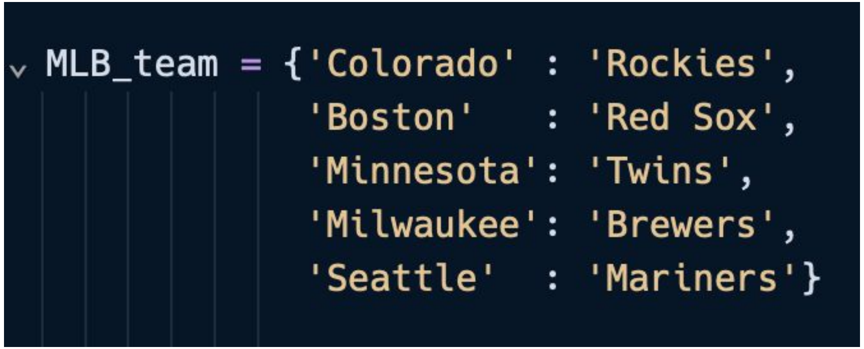
- *Calling a function*
- return statements – nothing goes after this!! special word that means end function

New Data Type: Dictionaries

- just like a paper dictionary for human language, we can make a dictionary in code to remember **key**: **value** pairs that go together
- we will use curly brackets { } to denote a dictionary
    - entries should be separated by commas
- and then use square brackets to ask for values from inside [ ]
    - just like asking for values from a list!
    - and, to add new values / update old ones



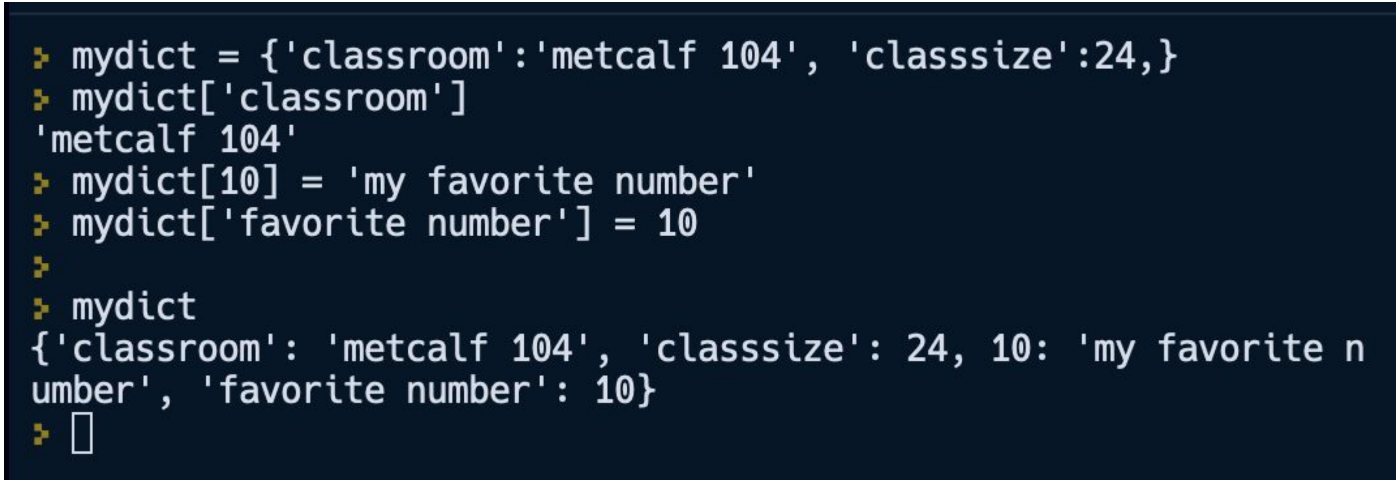
```
❯ MLB_team['Colorado']
'Rockies'
❯ MLB_team['Providence']
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
KeyError: 'Providence'
❯ 
```



```
MLB_team = {'Colorado' : 'Rockies',
            'Boston'   : 'Red Sox',
            'Minnesota': 'Twins',
            'Milwaukee': 'Brewers',
            'Seattle'  : 'Mariners'}
```

Dictionaries Continued

- keys and values in dictionaries can be (almost) anything, and they don’t all have to match

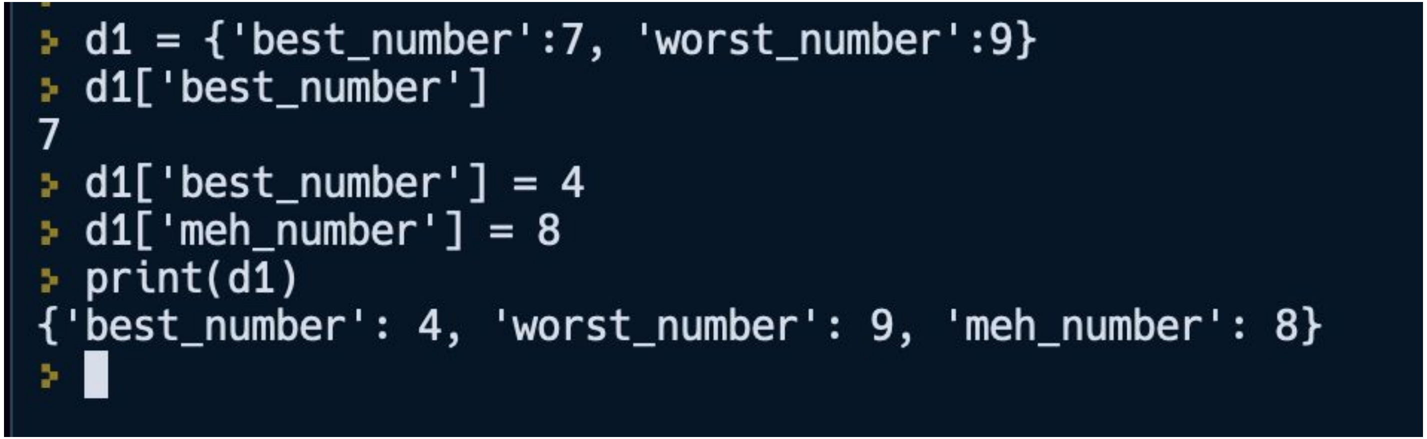


```
> mydict = {'classroom':'metcalf 104', 'classsize':24,}
> mydict['classroom']
'metcalf 104'
> mydict[10] = 'my favorite number'
> mydict['favorite number'] = 10
>
> mydict
{'classroom': 'metcalf 104', 'classsize': 24, 10: 'my favorite n
umber', 'favorite number': 10}
> 
```

Dictionary Mutation and Addition

we can change the values in dictionaries by resetting their values with an equals sign

we can also add new key: value pairs this way!



```
> d1 = {'best_number':7, 'worst_number':9}
> d1['best_number']
7
> d1['best_number'] = 4
> d1['meh_number'] = 8
> print(d1)
{'best_number': 4, 'worst_number': 9, 'meh_number': 8}
> 
```

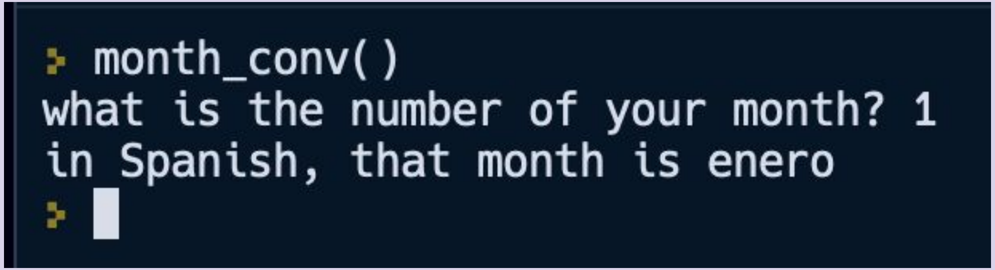
Dictionary Practice

make a dictionary that maps the number of a month to it’s word name!

you can name your months in English or any other language you prefer

then, use this dictionary inside a function which takes no arguments, asks the user for their numbered month, and then prints the message “In [your language], that month is [answer]”

for example:



```
> month_conv()
what is the number of your month? 1
in Spanish, that month is enero
> 
```

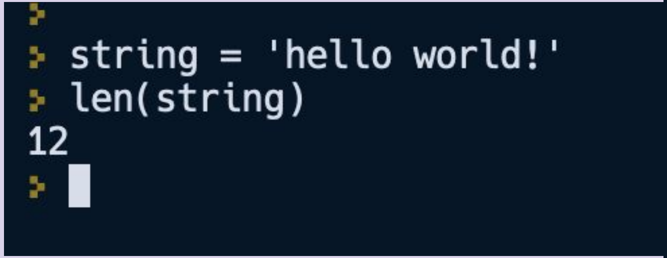
Class Problem

strings function like lists in python in that we can ask for how long they are using **len()**

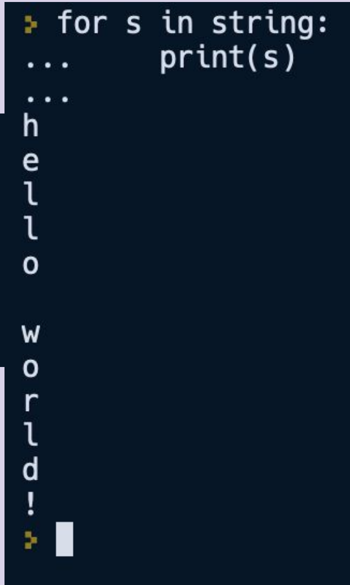
we can also loop through strings like lists!

write a function which has no arguments, asks for user input of a phone number, and decides whether the phone number is valid in the united states (has exactly 10 digits)

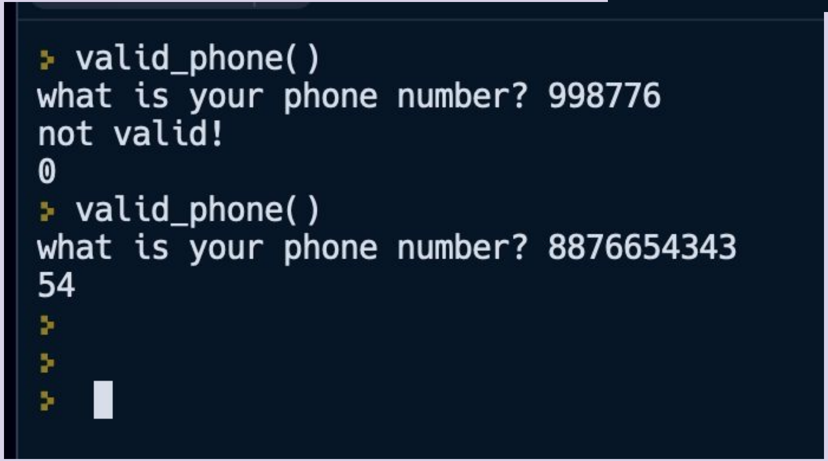
if it is valid, the function returns the **sum** of the digits in the phone number. if it is not valid, the function prints “not valid!” and returns 0



```
>
> string = 'hello world!'
> len(string)
12
> 
```



```
> for s in string:
...     print(s)
...
h
e
l
l
o
 
w
o
r
l
d
!
> 
```



```
> valid_phone()
what is your phone number? 998776
not valid!
0
> valid_phone()
what is your phone number? 8876654343
54
>
>
> 
```

Tournament Time

Your goal – to make the best possible character that will **win** battles against other class characters.

Challenge: you don’t know what battle types the character will be competing in ahead of time! So you need to be strategic in how you train and not neglect any skills.

However, you are friends with the battle master and she has allowed you to help create the rules for some of the competitions.

vs

Step 1: Character Creation

Your character can train with up to **100** points, across 10 features:

They also each have a Name, Story, and Special Power, which are strings.

You will define a dictionary containing these features (and give values from 0-100 for each), as well as their name, story, and special power. We will use these dictionaries for the battle.

Please also add ‘student’ which is your first name in all lowercase letters so we can track submissions.

your dictionary must be named [you]_d

example:

strength: 0

stealth: 16

speed: 7

smarts: 24

swimming: 40

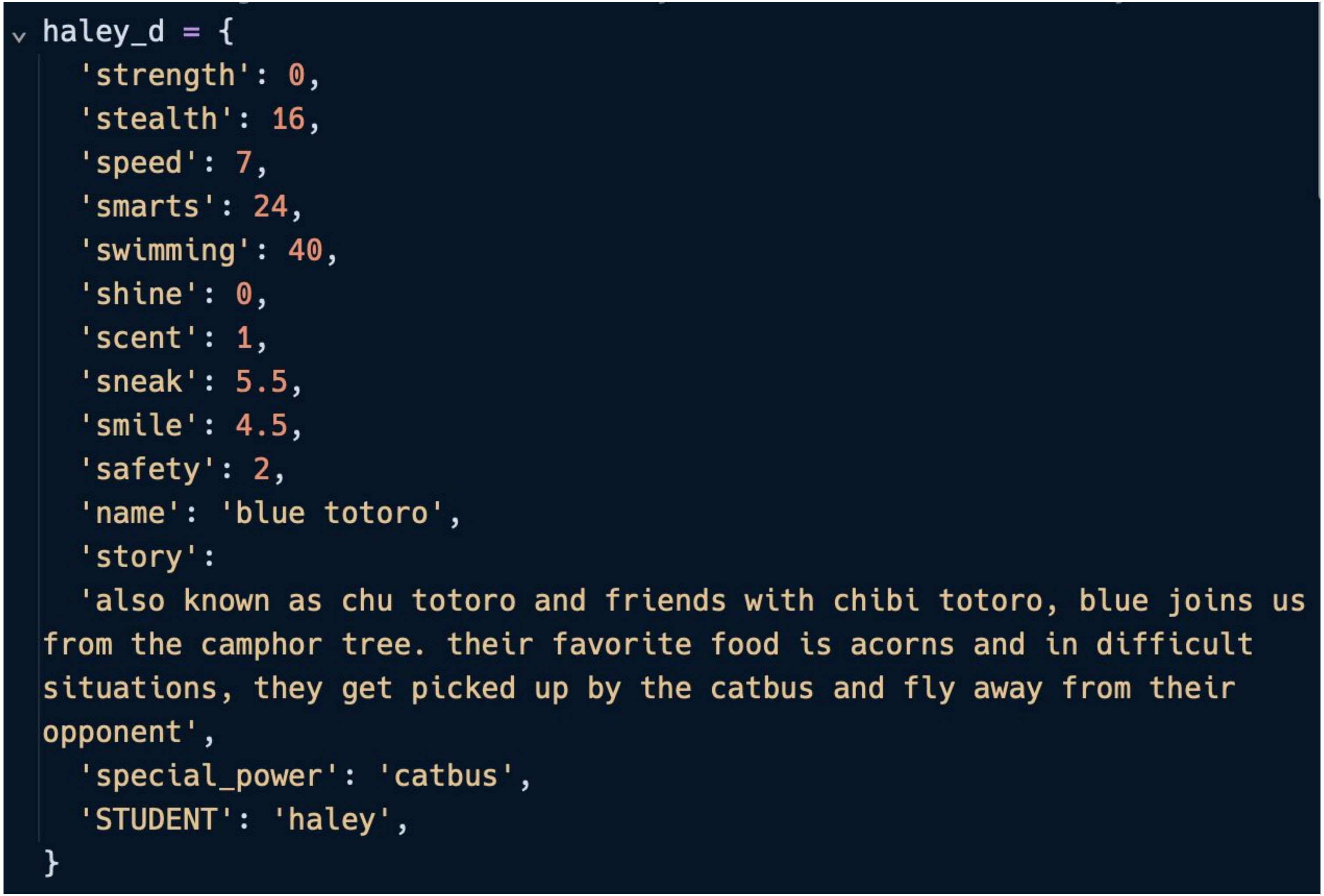
shine: 0

scent: 1

sneak: 5.5

smile: 4.5

safety: 2



```
haley_d = {
  'strength': 0,
  'stealth': 16,
  'speed': 7,
  'smarts': 24,
  'swimming': 40,
  'shine': 0,
  'scent': 1,
  'sneak': 5.5,
  'smile': 4.5,
  'safety': 2,
  'name': 'blue totoro',
  'story':
  'also known as chu totoro and friends with chibi totoro, blue joins us
from the camphor tree. their favorite food is acorns and in difficult
situations, they get picked up by the catbus and fly away from their
opponent',
  'special_power': 'catbus',
  'STUDENT': 'haley',
}
```

Group Dictionary Practice – Checking if legal players

Let’s write a function that all characters will have to go through before entering their arena, which decides if they are trying to sneak in any extra points.

In words, we want to know if the sum of their skill points is **less than or equal to** 100.

We should return True if the player is legal, and False if not. We don’t care if the player has less than 100 points – if they didn’t train enough, that’s their fault!

Step 2: Battle Preparation

you should now create **three** functions which will be used to battle characters!

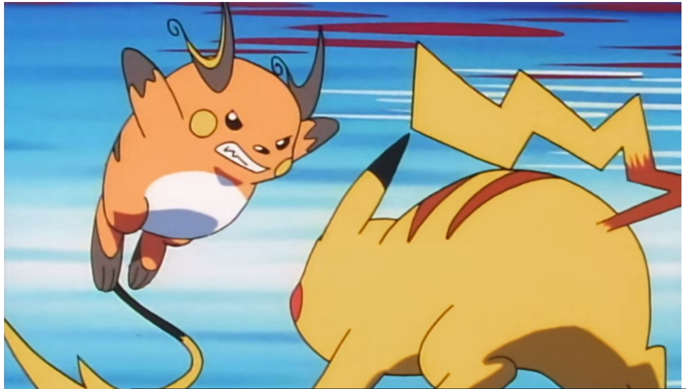
your first two functions should battle **two** characters – the arguments will be two character dictionaries

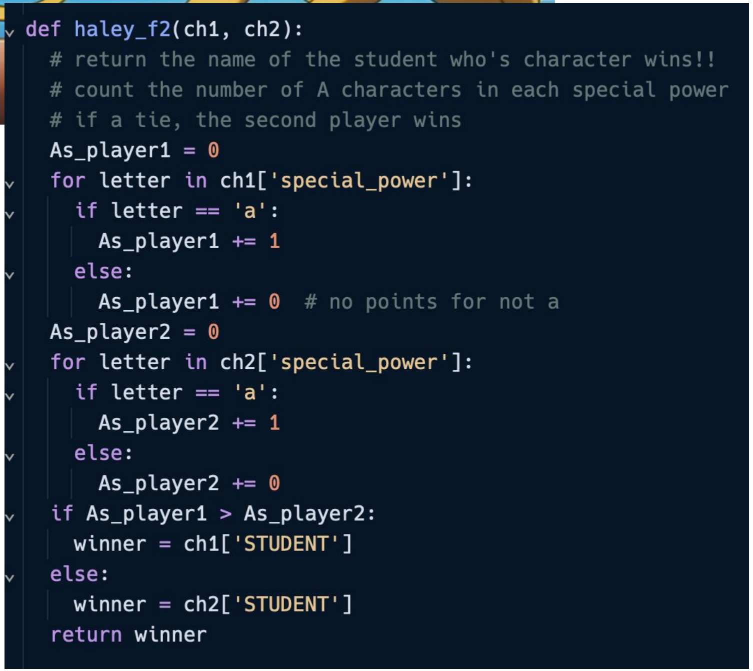
one function should be a battle royale and take three characters as arguments!

- at least one function should compare numeric properties, and at least one function should compare text properties

be creative! you don't know what arena you will end up in

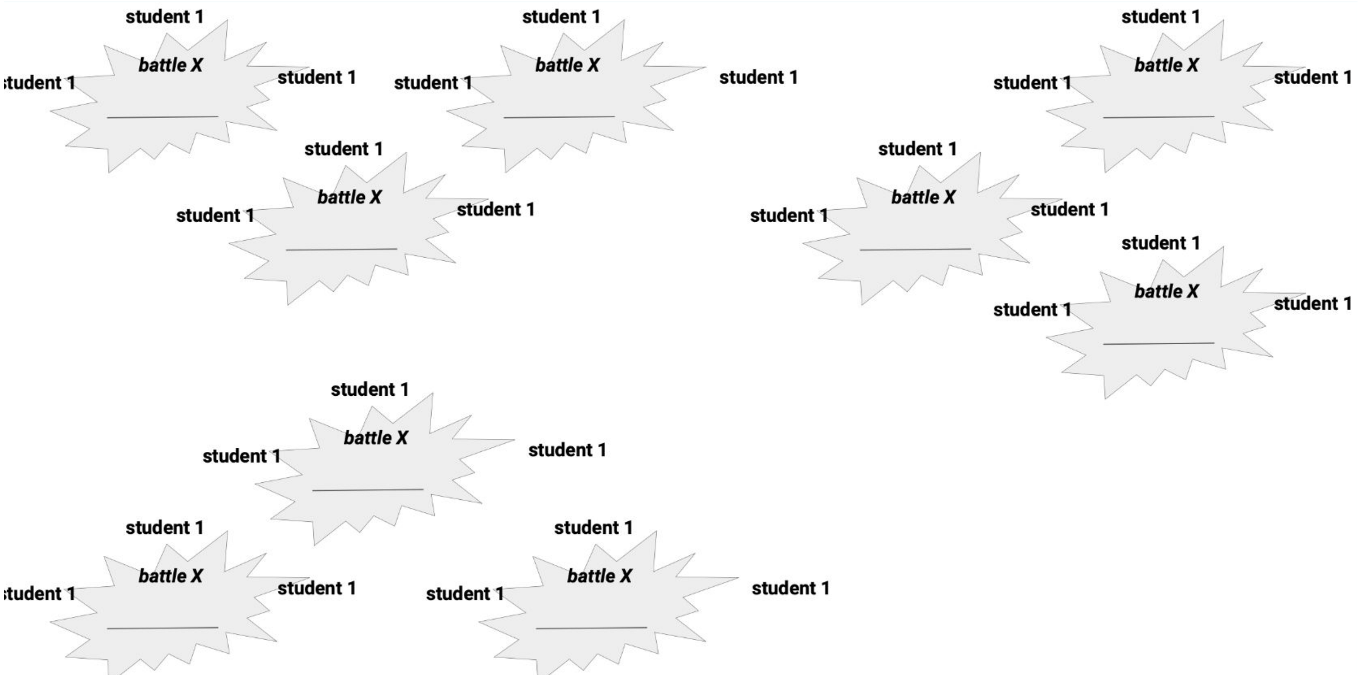
important: functions should be named [you]_f1, [you]_f2, and [you]_f3, and should return the **name** of the **student** with the winning character





```python
def haley_f2(ch1, ch2):
  # return the name of the student who's character wins!!
  # count the number of A characters in each special power
  # if a tie, the second player wins
  As_player1 = 0
  for letter in ch1['special_power']:
    if letter == 'a':
      As_player1 += 1
    else:
      As_player1 += 0  # no points for not a
  As_player2 = 0
  for letter in ch2['special_power']:
    if letter == 'a':
      As_player2 += 1
    else:
      As_player2 += 0
  if As_player1 > As_player2:
    winner = ch1['STUDENT']
  else:
    winner = ch2['STUDENT']
  return winner
```

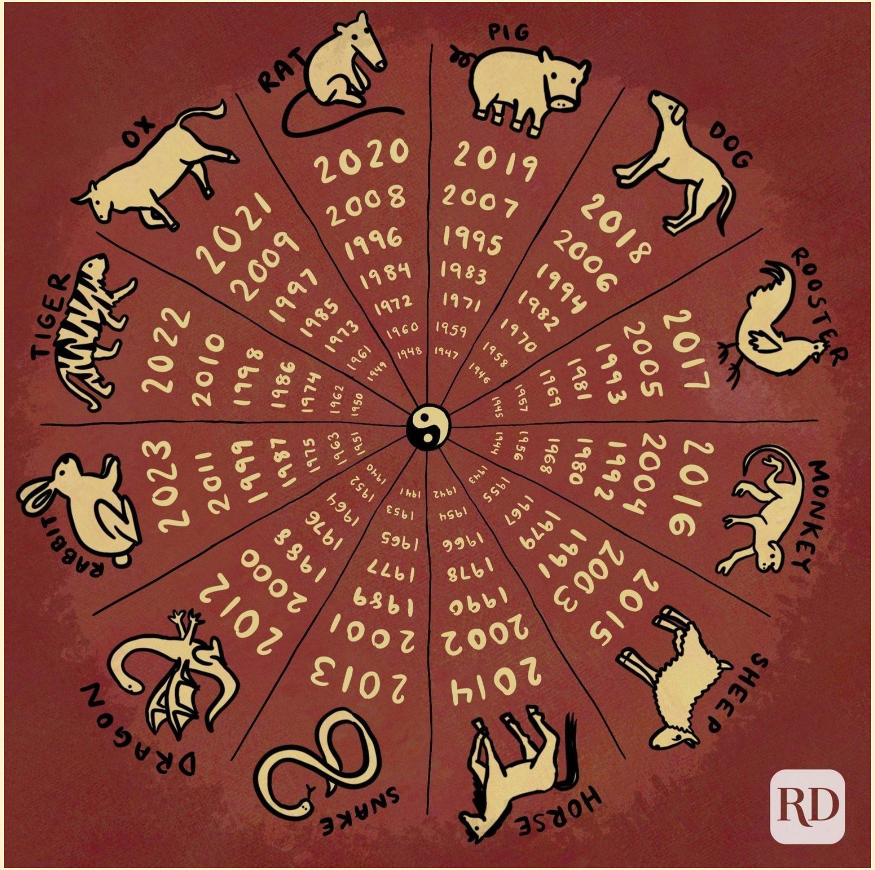
Step 3: Battle!!!



Homework Problem 1A: Zodiac Years

Write a function that takes in a year and prints the person’s zodiac animal, using this calendar.

HINT: Look at what the modulo value of each year is, and make a dictionary using those possibilities!



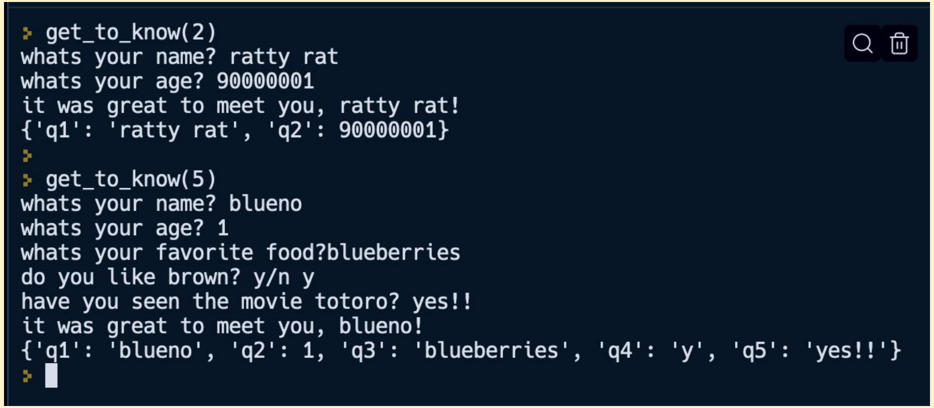
Homework Problem 1B: Making new dictionary friends

In this problem, you will write a function **get_to_know** which takes one argument **N**, which should be a number between 2 and 5. Then, you will ask your user **N** many questions to get to know them better!

The function should ask the user questions of your choice (for example, their name, age, and favorite food in the dining halls). At least one answer should be a **string** and at least one should be a **float or int**. As you learn about them, add their answers to a dictionary as **values**, with keys you choose.

After asking all of these questions, you should print “nice to meet you!” and a message **including at least one answer to a question you asked**. The function should return the dictionary with your information.

Hint1: you can make an empty dictionary by just setting empty curly brackets! d = {}. Hint2: you will need to use a for loop! Hint3: You can put your questions into a dictionary, with keys as the numbers of the questions.



```
› get_to_know(2)
whats your name? ratty rat
whats your age? 90000001
it was great to meet you, ratty rat!
{'q1': 'ratty rat', 'q2': 90000001}
›
› get_to_know(5)
whats your name? blueno
whats your age? 1
whats your favorite food?blueberries
do you like brown? y/n y
have you seen the movie totoro? yes!!
it was great to meet you, blueno!
{'q1': 'blueno', 'q2': 1, 'q3': 'blueberries', 'q4': 'y', 'q5': 'yes!!'}
›
```

Homework Problem 2: Reflection

Think back through this week on all the work you have done and problems you have solved (scroll through your homeworks if you need a refresher!). How did you go about solving problems? When you saw an error in replit (red text or an output you didn't expect), what did you do? Some examples might include working with classmates or instructors, walking through common bugs slides, looking back to previous problems, googling, etc. Describe your process for doing your assignments. Also, what has been your favorite activity so far? What was the hardest thing to learn?

Please write between 200 and 250 words.

Homework Problem 3: Looking forward

Next week we will be working with datasets to make graphs! We have a few options for data files we have found which you may be interested in working with. From the following options, please rank your top 3 choices for which data sets sound the most interesting to you. You will be working with this data for your final project, which will be in groups of 2 or 3 students. If there is anyone who you would like to work with, please list them here, and we will try to respect your wishes.

Options:

- Traffic Violations
- Movies on IMDB
- Air Quality
- Arrests in Rhode Island
- College Admissions
- Data Science Salaries
- Spotify Top Songs
- Tiktok Song Types