

Types

CAC 180

Amber Wagner

Birmingham-Southern College

Class Today

- Odds and Ends/Review
- Strings
- Practice

Gladys West

1931 -

- American Mathematician
- Studied Mathematics at Virginia State College
- Taught for two years
- 1956 - began working at Naval Surface Warfare Center Dahlgren Division
- Analyzed data from satellites to model Earth's shape
- Seasat radar altimetry project - first satellite that could remotely sense oceans



Gladys West

1931 -

- Programmer in the Dahlgren Division for large-scale computers and a project manager for data-processing systems used in the analysis of satellite data
- 1986 - West published “Data Processing System Specifications for the Geosat Satellite Radar Altimeter” for the Naval Surface Weapons Center (NSWC)
 - Major contribution to the accuracy of GPS and the measurement of satellite data
- Completed a PhD with Virginia Tech in 2018
- Inducted into the United States Air Force Hall of Fame in 2018

What is a variable?

Rules for Naming Variables

- One word
- Must start with a letter
- Cannot have special characters other than underscore
- Cannot be a keyword
- Should be meaningful
- Should start with a lowercase letter

Which ones are valid variable names?

1. value

6. 2x

2. my value

7. input

3. VALUE

8. total sum

4. total!

9. 4quarter

5. my_total

10. y

Printing Tricks

- What does the comma do in the print statement?
 - `print("The total is", total, ".")`
- What does the plus sign do?
 - `print("The total is " + str(total) + ".")`

Printing Tricks

- Assuming total is a float, the following line causes an error, why?
 - `print("The total is " + total + ".")`
- The plus sign is *overloaded*. It is used to add numbers (mathematical addition) and strings (concatenation). How should it handle adding a number to a string?
- How do we correct the line?

Printing Tricks

- Assuming total is a float, the following line causes an error, why?
 - `print("The total is " + total + ".")`
- The plus sign is *overloaded*. It is used to add numbers (mathematical addition) and strings (concatenation). How should it handle adding a number to a string?
- How do we correct the line? Cast the total as a string.
 - `print("The total is " + str(total) + ".")`

What is a string?

Example

H	E	L	L	O
0	1	2	3	4

- The word 'HELLO' has how many letters? This is referred to as a string's length.
- Notice that the positions go from 0 to 4.
- The positions always start with 0.
- Because strings have individually named positions, programmers can access each individual letter.

Finding the Length

- To find the length of a string, use *len()*

```
myString = 'pumpkin'  
print('The length of the string is', len(myString))
```

- Why do we care about length?
 - Password verification
 - Field validity: 10 digit phone number
 - When traversing through a string, it sets the boundaries of the loop

Accessing Elements

- When accessing the individual elements of a string, use square brackets after the name

fName = 'Susan'

mName = 'Rachel'

lName = 'Smith'

initials = fName[0] + mName[0] + lName[0]

String
Concatenation

Let's Think...

- If I have the integer 15, how can I break this down into the tens value and the ones value?
- In other words, I want to be able to find that I have a 1 followed by a 5.
- If it is an integer, I would have to use math to manipulate it.

Let's Think...

- Assume I have the following:
`num = int(input('Enter a two digit number: '))`
- Determine individual numbers using math:
`tens = num // 10`
`ones = num % 10`
`print('Your number has a', tens, 'followed by a', ones)`
- Determine individual numbers using string:
`num = input('Enter a two digit number: ')`
`print('Your number has a', num[0], 'followed by a', num[1])`

Immutable

- Strings are immutable.
- Meaning that you can't change the value of a string once you set it.

Immutable

name = 'Joe Smith'

name

Joe Smith

name = 'Joseph Smith'

name

Joe Smith

Joseph Smith

Repetition

- In Python, you can multiply strings.
- What do you think would be the outcome of the following?
word = 'go'
newWord = word*3

Slicing

- This is often referred to as substring
- You use the string followed by [:]
- Example:
name = 'Joe Smith'
fname = name[0:3]
- The first number represents the starting index, and the second number represents one past the upper bound of where we want to stop.

Let's Code

1. Print the first character of a string
2. Print the last character of a string
3. Ask the user to enter separate words and build a string (first name, middle name, last name)
4. Using string repetition, print 'Mississippi'

Lists

Lists

- Give me a list of colors...
- A list is a type of container or data structure that stores a group of items. Very similar to a string.
- They do not have to be the same type (unlike in other languages)
 - `colorList = ['red','blue','green','purple']`
 - `randomList = [5,'blue',7,num]`

Example

```
colorList = ['red','blue','green','purple']
```

'red'	'blue'	'green'	'purple'
0	1	2	3

Common List Methods

- `append()`
This will append items to the end of a list
Ex: `myList.append('abc')`
- `pop()`
Removes the item from the list at a specified position
Ex: `myList.pop(3)`
 - If no value is provided in the parentheses, it removes the last item in the list
- `remove()`
Removes the first occurrence of a specified item
Ex: `myList.remove('apple')`

More Methods

Operation	Description
<code>len(list)</code>	Find the length of the list.
<code>list1 + list2</code>	Produce a new list by concatenating list2 to the end of list1.
<code>min(list)</code>	Find the element in list with the smallest value.
<code>max(list)</code>	Find the element in list with the largest value.
<code>sum(list)</code>	Find the sum of all elements of a list (numbers only).
<code>list.index(val)</code>	Find the index of the first element in list whose value matches val.
<code>list.count(val)</code>	Count the number of occurrences of the value val in list.

Tuples

- Similar to a list but immutable (what does that mean again?)
- *variable_name = (val, val, val)*
- Examples:
 - Coordinates
 - RGB values of pixels

Why do these
make sense
as tuples?

Dictionaries

- How does the Oxford dictionary work?
- A dictionary in programming works the same way, there is a key and a value
- *dictionary_name = {key:value, key:value...}*


Example

- I have a list of contacts: names and numbers
- Why would it be a bad idea to store the contacts in two lists...one for names and one for numbers?
- What should the key be for a list of contacts?
- What should the value be?

Names		Numbers
Suzy	→	2055551234
Jack	→	2055552345
Kim	→	2055553456
John	→	2055554567
David	→	2055555678

Example Answers

- I have a list of contacts: names and numbers
- Why would it be a bad idea to store the contacts in two lists...one for names and one for numbers?
 - If you were to delete a name from the name list but forget to delete the associated number, the lists now do not match up.
- What should the key be for a list of contacts?
 - Name
- What should the value be?
 - Number

Names		Numbers
Suzy	→	2055551234
 Mark	↗	2055552345
Kim	↗	2055553456
John	↗	2055554567
David	↗	2055555678

Creating the Dictionary

```
contacts={  
'Suzy':'2055551234',  
'Jack':'2055552345',  
'Kim':'2055553456',  
'John':'2055554567',  
'David':'2055555678'  
}
```

Name	Number
Suzy	2055551234
Jack	2055552345
Kim	2055553456
John	2055554567
David	2055555678

Alternative Method - Helpful for Adding Entries

```
contacts = {}
```

```
contacts['Suzy'] = '2055551234'
```

```
contacts['Jack'] = '2055552345'
```

```
contacts['Kim'] = '2055553456'
```

```
contacts['John'] = '2055554567'
```

```
contacts['David'] = '2055555678'
```

Name	Number
Suzy	2055551234
Jack	2055552345
Kim	2055553456
John	2055554567
David	2055555678

Getting an Item from a Dictionary

- How do we get the phone number for one of our contacts?
- `print(contacts['Suzy'])`
- Dictionaries don't work by position, they work by the key
- Dictionaries are actually hashmaps, can't guarantee order

Updating an Entry

- `contacts['Suzy'] = '2055559876'`

Practice #1

- **Populate a dictionary with the following information (you're welcome to use your own information if you prefer).**
- **Print it to the screen.**
- **Allow the user to edit one player's total yards.**
- **Allow the user to ask for the total yards for one particular player and print as a whole number.**
- **Challenge: For the player above, print the difference between that player's total yards and the average total yards among the five quarterbacks.**
- **Total career passing yards:**
 - **Peyton Manning - 71940**
 - **Brett Favre - 71838**
 - **Drew Brees - 80358**
 - **Tom Brady - 79204**
 - **Dan Marino - 61361**

For Next Class

- Lab Friday - focus on strings, lists, and dictionaries
- Start working on Assignment #1 - you now have all of the information you need...NO GOOGLE!
- Have a wonderful day!