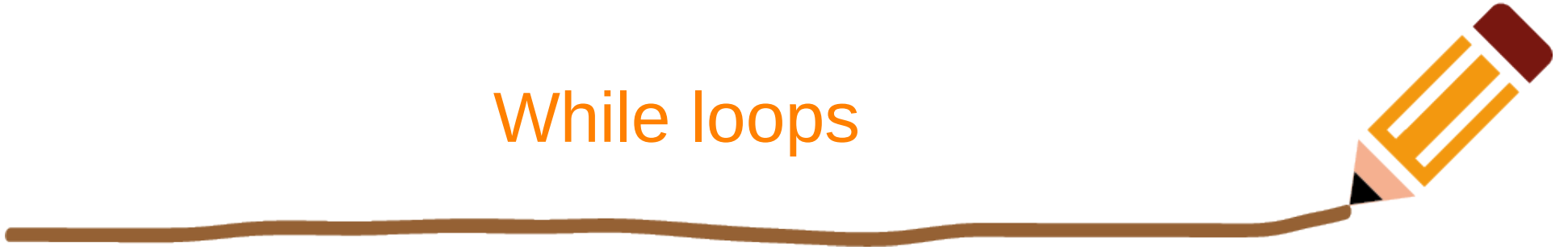


While loops



Admin



- Questions?
- Project?
- And a reminder →
-



© Sarah Andersen

Two kinds of loops



- Two kinds of loops in programming
 - We do something for each element of a collection
 - Which we use what keyword?

Two kinds of loops



- Two kinds of loops in programming
 - We do something for each element of a collection
 - Which we use what keyword?
 - for
 - We do something until the world situation meets some condition
 - Real life
 - Shovel snow till it is gone
 - Programming
 - Ask user for input till they give the right answer

While



- Keep doing it until the condition is met loop in python: While

- Syntax

- while <condition>:
 - Do codeblock
- Condition can be any boolean condition as with if statements

```
number_of_questions = get_number_of_students()
while number_of_questions > 0:
    print("your question is ....")
    number_of_questions = number_of_questions - 1
```

While : infinite loops



- Be careful of **infinite loops**
 - You need to update something about the condition inside of the loop or you will loop forever
 - Lets try it and see.

```
number_of_questions = get_number_of_students()
while number_of_questions > 0:
    print("your question is ....")
    #oops I forgot to update number_of_questions
```

While with a 'flag'



- A **flag** variable is a term for using a boolean type variable to determine if we should do something or not
- Set the flag to false till the world changes to be done the loop

```
done = False
while not done:
    answer = float(input("Enter gpa, enter number from 0.0 to 4.0:"))
    if 0.0 <= answer <= 4.0:
        done = True
```

#then we continue here

*#note that older versions of python we would have to use and
if 0.0 <= answer and answer <= 4.0*

Loop forever



- If your program should do something 'forever' then we have a special loop
 - Where here forever means till the program ends
- Usually used in games, operating systems etc
-
- while True:
 - Do stuff here

Loop Forever II



- If loop forever means till the program ends, we need to exit loop when program ends
 - Python keyword **break** will exit current loop (either while or for)
- while True:
 - Do stuff here
 - if end:
 - break

Class Exercise



- Stalker 'friend'
- Or
- Insistent administrator
- Ask the user to join them again and again till they say any answer like 'yes' (let's accept 'yes', 'yup', 'yeah' or 'ok')
 - Hint: string startswith() method
 - Hint use list of strings and check with if input in list
- Then celebrate
 - First in text, then using audio

To say the text out loud



- In order to have the program say the text, we could write a parsing engine and code to create audio
 - But that would be a lot of work and
 - ‘programmers are lazy’
 - So what should we do?

To say the text out loud



- In order to have the program say the text, we could write a parsing engine and code to create audio
 - But that would be a lot of work and
 - ‘programmers are lazy’
 - So what should we do?
 - Use an existing library!! (or three in this case)
- MAC users: <pycharm><preferences>
- Windows/Linux users: <file><settings>
- Then choose the python interpreter from the project tab

Add basic library



- pyttsx3
- A simple use is to the right.
- It is lightweight, installs quickly, and sounds robotic.
- Lets adjust this to our program
- Highlighted is optional to change sound

- `import pyttsx3`

```
engine = pyttsx3.init()
engine.setProperty('voice', 'gmw/en-us')
rate = engine.getProperty('rate')
engine.setProperty('rate', rate - 50)
engine.say("Hello Comp 151, let's talk.")
engine.runAndWait()
```

More convincing Text to Speech



- Modern TTS is **much** better
- But it isn't as trivial
 - Either use someone else's fancy computer ('the cloud')
 - Or have an nvidia GPU
 - Often complex setup/install
 - Especially for CS1 students
- Let's try one
 - Much harder than pyttsx3
 - But better quality.

More convincing Text to Speech



- Install two new packages
 - Using the usual settings
 - kokoro_onnx
 - sounddevice
- Download
 - <https://github.com/thewh1teagle/kokoro-onnx/releases/download/model-files-v1.0/kokoro-v1.0.onnx>
 - And
 - <https://github.com/thewh1teagle/kokoro-onnx/releases/download/model-files-v1.0/voices-v1.0.bin>



- Example

```
import sounddevice as sd
```

```
from kokoro_onnx import Kokoro
```

```
kokoro = Kokoro("kokoro-v1.0.onnx", "voices-v1.0.bin")
```

```
samples, sample_rate = kokoro.create(
```

```
    "Hello. Comp 151! We are speaking now", voice="af_sarah", speed=1.0, lang="en-us"
```

```
)
```

```
print("Playing audio...")
```

```
sd.play(samples, sample_rate)
```

```
sd.wait()
```


Now let's update our program



- To say rather than print.

Removing values from a list



- The list remove function does what?

Removing values from a list



- The list remove function does what?
 - Removes the *first* item that matches from the list
 - What if you want to remove them all?
 - You have to be careful with for loop (especially with range)
 - But can use while loop and remove together
- ```
books = ["Python Crash Course", "Game Programming", "Code Complete", "Clean Code", "The Pragmatic Programmer", "Python Crash Course"]
```
- ```
while "Python Crash Course" in books:
  - books.remove("Python Crash Course")
```

Another in class Exercise



- Lets grab the CountryPop text file from the class resources page;
 - get it and put it in your pycharm project folder
 - Data source: <https://www.worldometers.info/world-population/population-by-country/>
- Write a small program to open this file
 - Read in all the lines,
 - Create a list
 - For each line in the file create a dictionary with three keys
 - “country” “population” and “pop_change”
 - Add the dictionary to the list.
 - Just to make sure it worked, print each item in the list
- I did this for us for project 5 – now lets try it together.

One more use for while loops



- Let's program like its 1993
 - Or deep inside of any modern program
- Let's extend that last program to add a text based menu
- We need a while loop in that loop
 - Print the menu
 - Read input from the user
 - In an if-elif...else statement pick the right activity based on the users request and then do it
 - Find largest
 - Find smallest
 - Add new
 - quit

Reading



- Read Chapter 7 with these slides
-