Timeline

Description automatically generated with medium confidence

Diagram

Description automatically generated

**Q1**:**How does the for-loop work?**

**A**: The for-loop executes the indented lines n-times, where n is the number of items in the list that is being iterated.

**Q2: What is the difference between a while loop and a for-loop?**

**A**: A while loop iterates as long as the while-condition is True. A for-loop iterates that many times as there are items in the list that is being iterated.

**Q3: What are some other examples when we would need to use a for-loop?**

A: A for-loop is generally needed when you want to perform the same manipulation on multiple objects (e.g., capitalizing strings of a list). Another example of when you would need to use a for-loop is if you wanted to convert a video to grayscale. A video is made of hundreds or thousands of image frames. You would need to use a for-loop to iterate over the video frames and apply a grayscale effect to each frame. Another example would be to iterate over the pages of your blog website to extract their user comments, etc.

**Q4: What is the difference between match-case and if-else?**

Match-case is usually to match a string out of a predefined number of strings (e.g., a series of commands, months of the year, etc.).  If-else conditionals check more complex conditions. We will cover if-else conditionals later in the course. There you will understand the exact difference.

Diagram

Description automatically generated

**Q1: When should we use integers and when floats?**

**A**: Integers represent whole numbers, while floats represent decimal numbers. Therefore, if you are getting the number of eggs, the rank of university students, the number of participants in an event, or other whole numbers from the users, then you know you want to work with integers. On the other hand, if you are working with continuous data such as temperature, speed, etc., you want to work with floats.

**Q2: Why do we use parenthesis in functions (e.g., input() and print()) and square brackets when accessing list items (e.g., todos[3]?**

**A**: That is part of the deal/convention/syntax, so the interpreter knows exactly what you are trying to do. In this case, it would help the interpreter understand whether you are calling a function or accessing a list item.

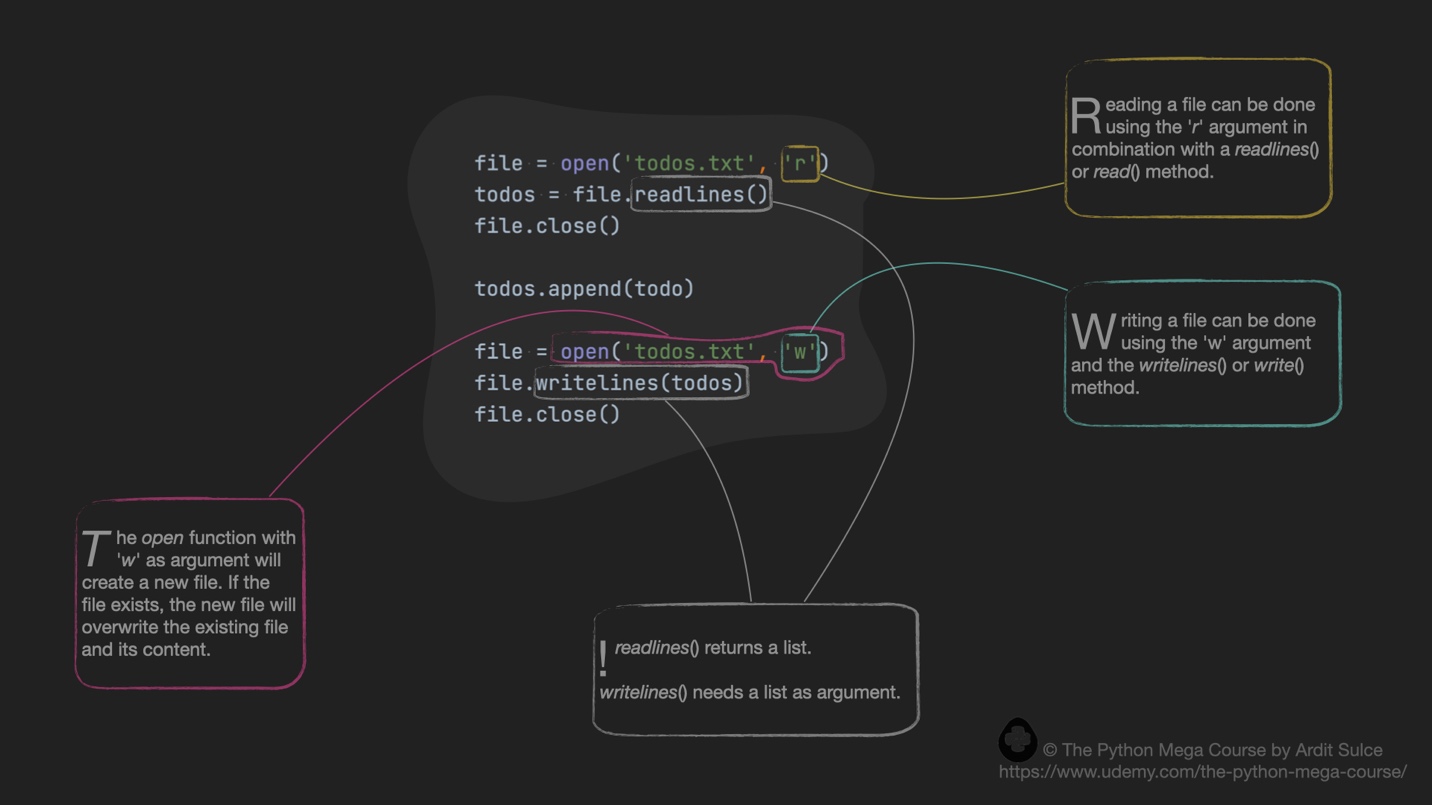
**Q3: Why does indexing start from zero?**

**A**: It is a common design choice across (almost?) all languages. If you have trouble wrapping your mind about the zero-origin system, think of it as a coordinate system that starts from zero and think of the indices as offsets from the origin.

Diagram

Description automatically generated

Enumerate



Files

A picture containing text, screenshot, diagram, font

Description automatically generated

A screenshot of a computer code

Description automatically generated with low confidence

**Q1: We are storing the to-do items in a text file. What would happen if we stored them in an Excel file?**

**A**: If we worked with Excel files instead of text files, we would need to define other functions which would contain code elements that handle Excel files. It is recommended not to delete or change the existing functions but to simply add other functions in the functions.py file. Later in the course, we will cover how to work with Excel files as well.

A screenshot of a computer

Description automatically generated with medium confidence

**Q1: In the app, we imported functions.py from main.py. Is it possible to import main.py from functions.py?**

**A**: Yes, that is technically possible. You can do import main in functions.py. However, that import doesn't make much sense. We usually import the backend to the frontend. The backend is the script where the processing is done (i.e., reading and writing the to-do files), and the frontend is the code that constructs the user interface (i.e., the command line).

**Q2: Can we import more than one .py file from the main.py file?**

**A**: Yes, that is possible, and as the program expands, it is recommended to create more backend files, and you can import all those files from main.py. For example, you might want to create some functions which send out the to-do items by email to your email address. You might also want to create some functions that produce a PDF with the to-do items inside. It is recommended to write the email and the PDF functions in separate modules and then import those modules from the main module. For example, if we had these files:

main.py

functions.py

pdf.py

email.py

In main.py we would have these lines:

1. import functions
2. import pdf
3. import email

A screenshot of a computer

Description automatically generated with medium confidence

<https://docs.python.org/3/py-modindex.html>

A screenshot of a computer program

Description automatically generated with medium confidence