ordered vs unordered Sequence

- **Ordered Sequence**: elements accessed using an index that represents the location
  - List, Tuple, String
- **Unordered Sequence**: elements accessed using a key
  - Dictionary

\[
\text{dic} = \{ \text{k0: v1, k1: v2, ..., k10: v11} \}
\]
\[
\text{dic}[\text{k5}] = \text{value}
\]
Announcements: Project Overview

Hi everyone,

I'm sure a lot of people are wondering, "how do we get started with the project? What should we do now?" To help address these questions, I made a project intro video where I go over the project page, different components of the project, what you'll be graded on, and how to code with your teammates.

If you aren't sure how to get started, please give it a watch!

[VIDEO]: https://drive.google.com/file/d/18FnQQlZfDKBF9z71WBej_rntVzat0iIF/view?usp=sharing

Project Report 1.1 (due on Friday 9/30 10:00 pm)
Announcements

- HW02 (due on Monday 9/26 10:00 pm)
- Lesson06, Lesson07 (due on Wednesday 9/28 10:00 pm)
- HW03 (due on Monday 10/3 10:00 pm)
- Lesson08, Lesson09 (due on Wednesday 10/05 10:00 pm)
# Updated Office Hours

<table>
<thead>
<tr>
<th>Day</th>
<th>Time</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monday</td>
<td>9:00</td>
<td>Discord</td>
</tr>
<tr>
<td>Monday</td>
<td>13:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Monday</td>
<td>15:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Monday</td>
<td>16:00</td>
<td>Discord</td>
</tr>
<tr>
<td>Tuesday</td>
<td>10:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Tuesday</td>
<td>11:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Tuesday</td>
<td>14:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Tuesday</td>
<td>15:00</td>
<td>Discord</td>
</tr>
<tr>
<td>Wednesday</td>
<td>9:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Wednesday</td>
<td>13:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Wednesday</td>
<td>14:00</td>
<td>Discord</td>
</tr>
<tr>
<td>Wednesday</td>
<td>15:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Wednesday</td>
<td>17:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Thursday</td>
<td>11:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Thursday</td>
<td>12:00</td>
<td>Discord</td>
</tr>
<tr>
<td>Thursday</td>
<td>13:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Thursday</td>
<td>14:00</td>
<td>Discord</td>
</tr>
<tr>
<td>Friday</td>
<td>10:00</td>
<td>Discord</td>
</tr>
<tr>
<td>Friday</td>
<td>13:00</td>
<td>0224 Siebel</td>
</tr>
<tr>
<td>Friday</td>
<td>14:00</td>
<td>Discord</td>
</tr>
</tbody>
</table>
Topics Covered in Week 5 (Lesson 06, 07)

- Files
  - read
  - readlines

- List Operations

- Dictionaries
Today's Plan Week 6 (Lesson 08)

- Mutable vs Immutable
- Aliasing
- Copying Lists:
  - Shallow copy
- Loop Aids:
  - continue, break, zip, and enumerate
Create a group of two or three colleagues and work on the following exercise:

- Search for an element in the list, return the search outcome (True/False) and the number of iterations. Your solution should use a break function.

```python
# function definition
def search(myList, element):
    # add your solution

# Testing
output1 = search(testList, 20)
output2 = search(testList, -5)
print(output1)  # Output: (True, 5)
print(output2)  # Output: (False, 6)
```

Wrap-up

- Today’s Lecture:
  - Mutable vs immutable
  - Aliasing and shallow copying
  - break, continue, zip, enumerate
- Next Lecture:
  - How to handle Exceptions (runtime errors that we anticipate)