**Topic 75: How to Retrieve a Python List or Dictionary from a JSON File**

**What**  
This topic explains how to retrieve a Python list or dictionary from a JSON file using the json module in Python. This allows you to load the data that was saved in the previous step back into your Python program for further use.

**Why**

* **Persistence of data**: Storing data in a JSON file allows you to persist complex data structures across sessions. You can retrieve the data later, even after closing and reopening the program.
* **Data exchange**: JSON files can be used to exchange data between different systems and applications, making retrieval from a JSON file a common task.

**How**

1. **Import the JSON module**:  
   If you haven’t already imported the json module, you need to do so:

python

Copy code

import json

1. **Open the file to read the data**:  
   Open the JSON file using the open() function in read mode ("r") (Python defaults to reading if mode is not specified):

python

Copy code

with open("customer\_29876.json") as f:

1. **Retrieve the data using json.load()**:  
   Call json.load() to deserialize the JSON data from the file and load it back into a Python object (in this case, a dictionary or list):

python

Copy code

customer\_29876 = json.load(f)

1. **Use the retrieved data**:  
   You can now use the retrieved data as you would with any other Python dictionary or list. For example, to print the dictionary:

python

Copy code

print(customer\_29876)

Or to access a specific value:

python

Copy code

print(customer\_29876["last name"])

**Things to Remember**

* **json.load()**: This function reads the JSON data from the file and converts it back into a Python object (e.g., a dictionary or list). The syntax is json.load(file\_handle).
* **Data types**: The data must be in a JSON-compatible format (lists, dictionaries, strings, numbers, etc.) in the file for json.load() to work properly.
* **File handling**: Always ensure that the file you're trying to read exists, otherwise Python will raise a FileNotFoundError.
* **Accessing data**: After loading the data, you can access individual elements (in a list) or values (in a dictionary) using the standard indexing or key access methods.