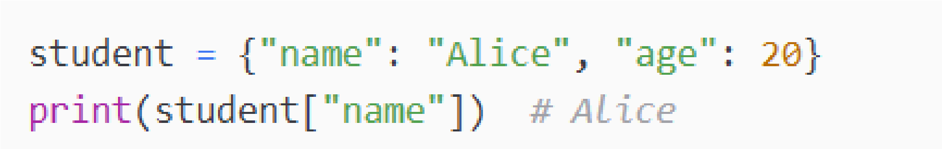
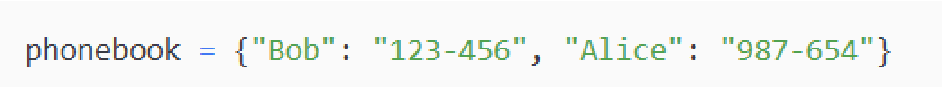
What is a Dictionary?

- What is a Dictionary?
  - **Formal:** An unordered, mutable collection mapping immutable keys to arbitrary values.
  - **Analogy:** Think of a real-world dictionary — words (keys) mapped to definitions (values).
  - **Collection type:** Stores data as **key → value** pairs.
  - **Keys:** Must be unique, immutable (string, number, tuple).
  - **Values:** Can be of any type.



```
student = {"name": "Alice", "age": 20}
print(student["name"])  # Alice
```

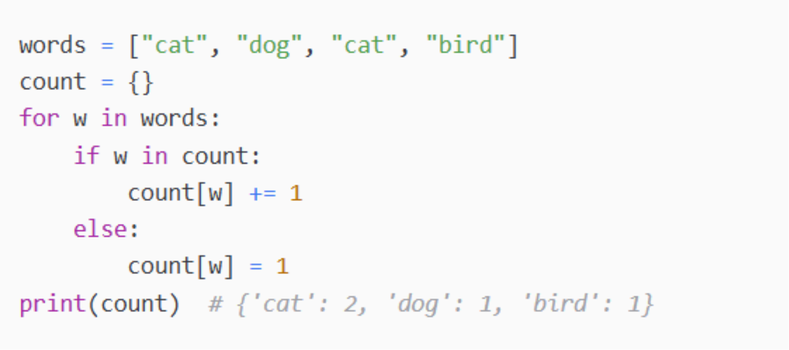


```
phonebook = {"Bob": "123-456", "Alice": "987-654"}
```

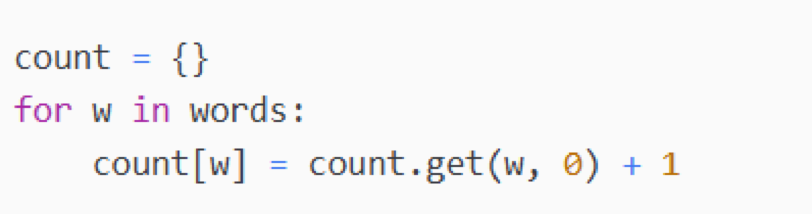
Creating Dictionaries

- Dictionary literal: {key: value, key2: value2}
- Empty dictionary: {} or dict()
- **Example:** fruit_colors = {"apple": "red", "banana": "yellow"}

- Counting with a Dictionary



```
words = ["cat", "dog", "cat", "bird"]
count = {}
for w in words:
    if w in count:
        count[w] += 1
    else:
        count[w] = 1
print(count)  # {'cat': 2, 'dog': 1, 'bird': 1}
```



```
count = {}
for w in words:
    count[w] = count.get(w, 0) + 1
```

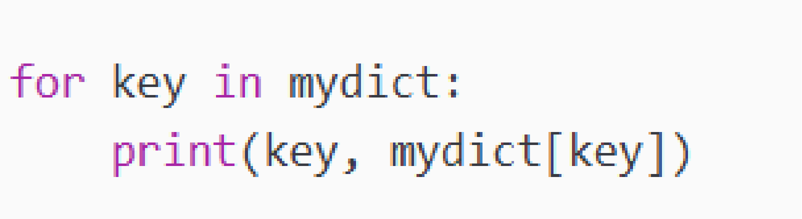
dict.get(key, default) returns value for key if exists, else default.

Retrieving Lists of Keys and Values

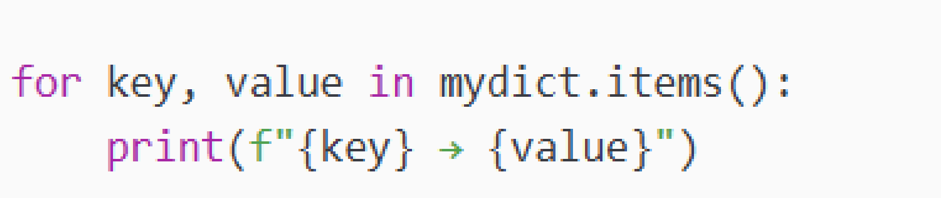
- Getting Keys and Values
  - mydict.keys() → view of keys
  - mydict.values() → view of values
  - mydict.items() → view of (key, value) pairs

- list(mydict.keys()) → all keys in a list
- list(mydict.values()) → all values in a list
- list(mydict.items()) → list of (key, value) tuples

- **Looping Through a Dictionary**

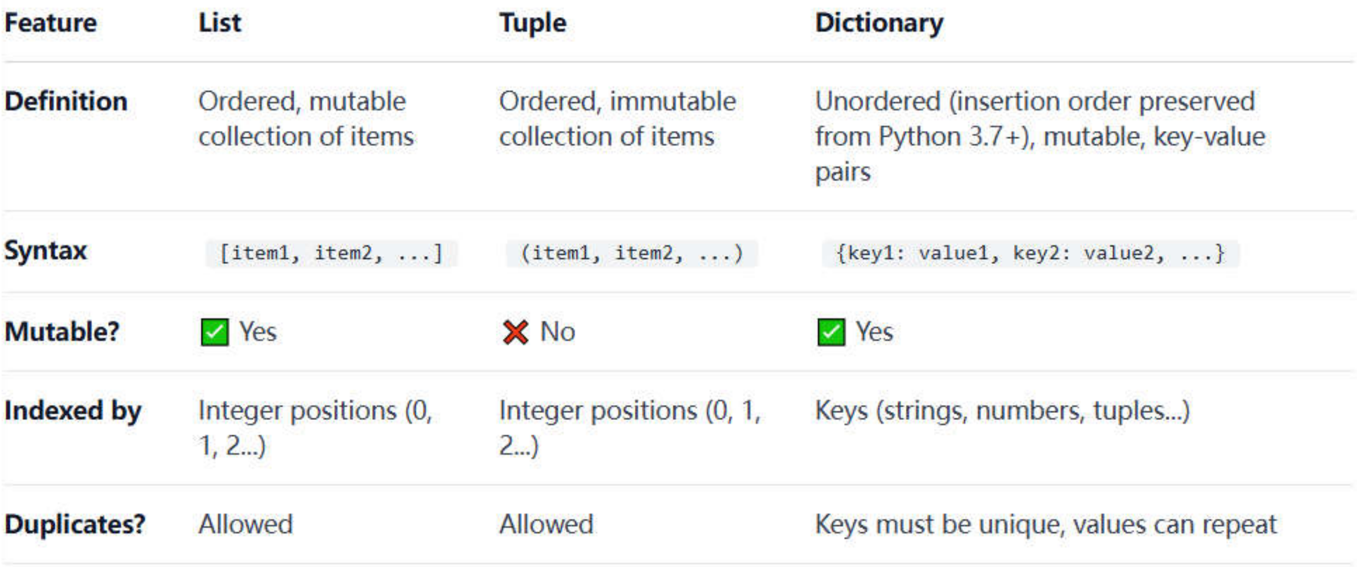


```python
for key in mydict:
    print(key, mydict[key])
```

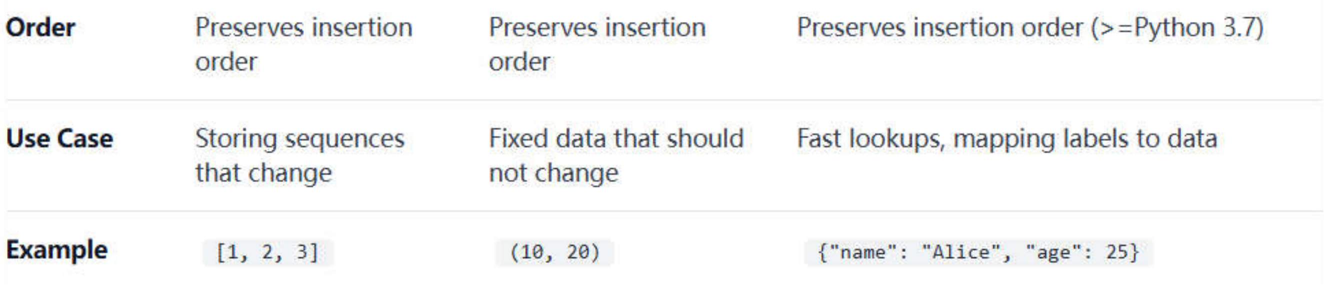


```python
for key, value in mydict.items():
    print(f"{key} → {value}")
```

List vs Tuple vs Dictionary

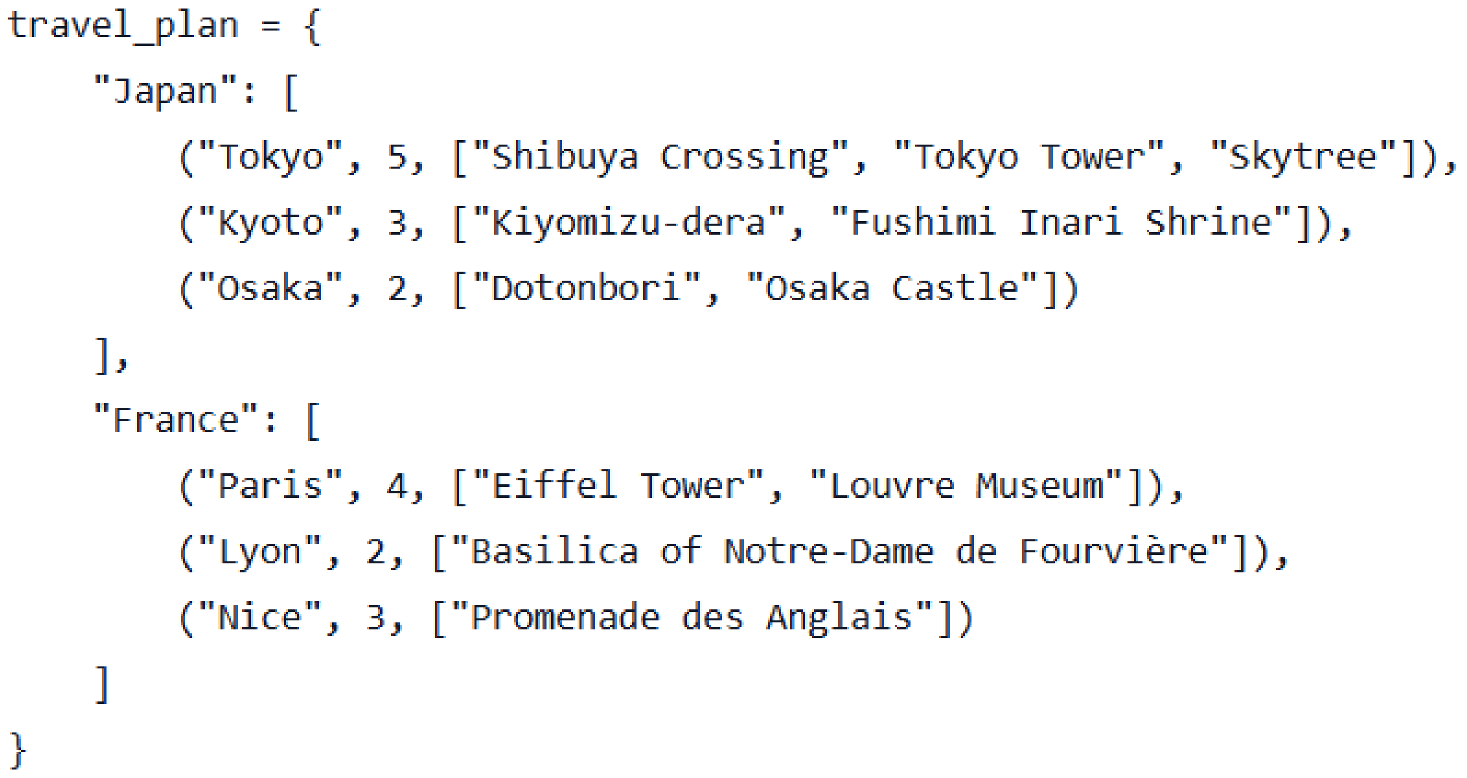


Feature	List	Tuple	Dictionary
Definition	Ordered, mutable collection of items	Ordered, immutable collection of items	Unordered (insertion order preserved from Python 3.7+), mutable, key-value pairs
Syntax	`[item1, item2, ...]`	`(item1, item2, ...)`	`{key1: value1, key2: value2, ...}`
Mutable?	✅ Yes	❌ No	✅ Yes
Indexed by	Integer positions (0, 1, 2...)	Integer positions (0, 1, 2...)	Keys (strings, numbers, tuples...)
Duplicates?	Allowed	Allowed	Keys must be unique, values can repeat



Order	Preserves insertion order	Preserves insertion order	Preserves insertion order (>=Python 3.7)
Use Case	Storing sequences that change	Fixed data that should not change	Fast lookups, mapping labels to data
Example	`[1, 2, 3]`	`(10, 20)`	`{"name": "Alice", "age": 25}`

Example



```
travel_plan = {
    "Japan": [
        ("Tokyo", 5, ["Shibuya Crossing", "Tokyo Tower", "Skytree"]),
        ("Kyoto", 3, ["Kiyomizu-dera", "Fushimi Inari Shrine"]),
        ("Osaka", 2, ["Dotonbori", "Osaka Castle"])
    ],
    "France": [
        ("Paris", 4, ["Eiffel Tower", "Louvre Museum"]),
        ("Lyon", 2, ["Basilica of Notre-Dame de Fourvière"]),
        ("Nice", 3, ["Promenade des Anglais"])
    ]
}
```

Output of travel_plan["Japan"][0][2][1]?
Code for look up “Eiffel Tower”?