6. Dictionaries

❖ Introduction to dictionaries: key-value pairs.

What is a dictionary?

A dictionary (*dict*) is a **mutable**, **ordered** (since Python 3.7), and **indexed** collection that maps **unique, immutable keys** to **values of any data type**

- **Keys** must be immutable (e.g., strings, numbers, tuples) and **unique**—duplicates overwrite existing entries .
- **Values** can be any type—even other dicts or lists—and need not be unique.

Example :

```
d = {“name”:”ajay”}

for k, v in d.items():
   print(f"{k}: {v}")
```

Iterating through dictionaries

Loop through:

- **Keys**: for k in d:
- **Values**: for v in d.values():

- **Pairs**: for k, v in d.items():

❖Accessing, adding, updating, and deleting dictionary elements

1. Accessing Elements :

- Bracket notation: value = d[key]
  - Direct lookup; fast (O(1)) due to hashing
  - Raises KeyError if the key doesn't exist
- get() method: value = d.get(key, default)
  - Returns default (or None) if the key is absent—safer access

2. Adding & Updating Elements:

- Direct assignment: d[key] = value

    - Creates a new entry or updates an existing key; average-case O(1)
- update() method: d.update(other_dict) or d.update(k=3, v=4)
    - Merges key-value pairs from another dict or via keyword args
- Merge operators (Python 3.9+):

3. Deleting Elements:

- del d[key]
    - Removes the key; raises KeyError if key not present
- d.pop(key[, default])
    - Removes and returns the value; returns default if key absent (optional)
- d.popitem()
    - Removes and returns the *last-inserted* key-value pair ((key, value)) in Python 3.7+
- d.clear()
        - Empties the entire dictionary

❖Dictionary methods like keys(), values(), and items().

1. dict.keys()

- Returns a dict_keys view of all keys.
- Supports iteration and in checks.
- Converts to a list for snapshot or indexing.

2. dict.values()

- Returns a dict_values view of all values.
- Useful for iteration and summarizing values.
- Also dynamic; converts to list when needed.

3. dict.items()

- Returns a dict_items view of (key, value) tuples.
- Great for tuple unpacking in loops.
- Updates with dictionary; can be converted to list.