Tasks – Comparison Operators

Practice comparison operators: `==`, `!=`, `<`, `>`, `<=`, `>=`. They return `True` or `False`. Create each file, run it, and check the output.

Run scripts with: `python3 script_name.py`

Part 1 – Equal (==) and not equal (!=)

Task 1.1 – == with numbers (`compare_equal.py`)

- Create `compare_equal.py`.
- Print the result of `5 == 5`. Print the result of `5 == 3`.

Expected output:

```
True
False
```

Task 1.2 – != with numbers (`compare_not_equal.py`)

- Create `compare_not_equal.py`.
- Print the result of `10 != 3`. Print the result of `7 != 7`.

Expected output:

```
True
False
```

Task 1.3 – == and != with variables (`compare_eq_var.py`)

- Create `compare_eq_var.py`.
- Assign `100` to `a` and `100` to `b`. Print `a == b`. Assign `50` to `c`. Print `a != c`.

Expected output:

```
True
True
```

Part 2 – Less than (<) and greater than (>)

Task 2.1 – < and > (`compare_less_greater.py`)

- Create `compare_less_greater.py`.
- Print `3 < 5`. Print `3 > 5`. Print `5 > 5`.

Expected output:

```
True
False
False
```

Task 2.2 – Variables and < (`compare_less_var.py`)

- Create `compare_less_var.py`.
- Assign `10` to `x` and `20` to `y`. Print `x < y`. Print `y < x`.

Expected output:

```
True
False
```

Part 3 – Less than or equal (<=) and greater than or equal (>=)

Task 3.1 – <= and >= (`compare_le_ge.py`)

- Create `compare_le_ge.py`.
- Print `4 <= 4`. Print `4 <= 5`. Print `4 >= 4`. Print `3 >= 5`.

Expected output:

```
True
True
True
False
```

Task 3.2 – <= with variables (`compare_le_var.py`)

- Create `compare_le_var.py`.
- Assign `7` to `n`. Print `n <= 10`. Print `n <= 7`. Print `n <= 5`.

Expected output:

```
True
True
False
```

Part 4 – Mix and strings

Task 4.1 – Several comparisons in one script (`compare_mixed.py`)

- Create `compare_mixed.py`.
- In one script, print the result of: `10 == 10`, `10 != 9`, `10 < 20`, `10 >= 10`. Use one print per expression (or combine with commas).

Expected output:

```
True
True
```

```
True
True
```

Task 4.2 – Strings (lexicographic order) (`compare_strings.py`)

- Create `compare_strings.py` .
- Comparisons work on strings (alphabetical/lexicographic order). Print `"apple" < "banana"` . Print `"abc" == "abc"` . Print `"abc" != "ABC"` (case matters).

Expected output:

```
True
True
True
```

Task 4.3 – Store result in variable (`compare_store.py`)

- Create `compare_store.py` .
- Assign to a variable the result of a comparison (e.g. `result = 5 >= 3`). Print the variable. Then assign another comparison (e.g. `result = 1 == 0`) and print again.

Expected output:

```
True
False
```

Done

You've used: `==` , `!=` , `<` , `>` , `<=` , `>=` with numbers and variables, and seen that strings can be compared (lexicographic order). Comparison results are booleans (`True` / `False`).