

Selection

Python Reference

if / elif / else

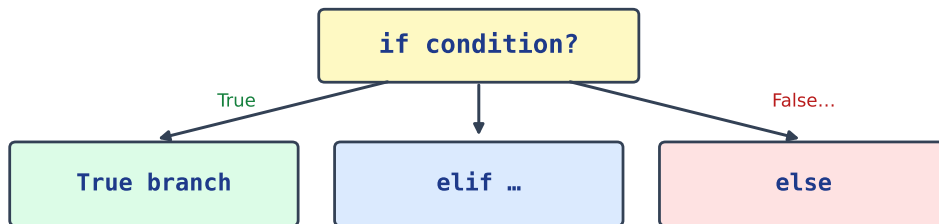
A program chooses what to do with `if`. It runs an indented block only when a condition is true. The `if` line ends with a colon .

```
score = 72
if score >= 60:
    print("pass")
# pass
```

Add `elif` (else-if) for more cases and `else` for "anything else". Python runs the **first** true branch only, then skips the rest.

```
score = 72
if score >= 80:
    print("A")
elif score >= 60:
    print("B")
else:
    print("fail")
# B
```

- Compare values with `==` (equal to), `!=` (not equal to), `<`, `>`, `<=`, `>=`.
- A comparison gives a Boolean — either `True` or `False`.



Exam tip: only one branch runs; conditions are checked top to bottom; indent the body

if / elif / else: only one branch runs

Combining conditions

Join conditions with `and`, `or`, `not`. `and` needs **both** sides true; `or` needs **either** side true; `not` flips a Boolean.

```
age = 16
has_ticket = True
if age >= 18 and has_ticket:
    print("entry allowed")
else:
    print("entry refused")
# entry refused
```

- Use brackets to make the order clear: `(a or b) and c`.

```
temp = 30
if temp > 25 and not temp > 35:
    print("warm but ok")
# warm but ok
```

Common mistakes

- Write `elif`, not `else if`.
- Every `if` / `elif` / `else` line ends with a colon `:`, and its body must be indented.
- `if x = 5:` is an error — use `==` to compare.
- `else` takes no condition; only `if` and `elif` do.