# If Statements: Making Decisions

Python for Absolute Beginners · Lesson 9

## What you'll learn

- How if runs code only when a condition is true
- How else covers the "otherwise" case
- How elif chains several choices
- Why indentation is part of the code, not decoration
- The two most common if mistakes

# The first decision

- Everything so far has run straight from top to bottom
- An if statement breaks that: it runs some code only when a condition is true, and skips it otherwise
- The condition is just a comparison, like the ones from Lesson 8
- The if line ends with a colon (:) — “here comes the code for this case”
- The lines underneath are indented (usually four spaces) — that is how Python knows they belong to the if



# A first if statement

```
                                age = 12
if age >= 10:
    print("You are 10 or older.")
print("Done.")

Output:
You are 10 or older.
Done.
```

The final print is not indented, so it runs no matter what.

# The same program, age 8

```
                                age = 8
if age >= 10:
    print("You are 10 or older.")
print("Done.")

Output:
Done.
```

The indented line is skipped entirely, because the condition is False.

# Otherwise: else

- else covers the case where the condition is false
- else needs no condition of its own — it means “in every other case”
- Like if, it ends with a colon and its code is indented



# Else in code

```
                              score = 45
if score >= 50:
    print("You passed!")
else:
    print("Keep practicing — you will get there.")

Output:
Keep practicing — you will get there.
```

The score is below 50, so the else block runs.

# Several choices: elif

- elif (short for “else if”) chains more than two cases in order
- Python checks each condition from the top and runs the first block whose condition is true — and only that one
- Order matters: always put the strictest condition first, or a looser one above it will swallow every case below

# Elif in code

```
score = 85

if score >= 90:
    print("Excellent!")
elif score >= 70:
    print("Good job!")
elif score >= 50:
    print("You passed.")
else:
    print("Keep practicing.")

Output:
Good job!
```

85 is not >= 90, so Python moves on; it is >= 70, so it prints “Good job!” and stops checking.

# Indentation is not decoration

- In most languages, indentation is just tidiness. In Python, it is meaning
- Same lines, different indentation, different program
- With age 8, a program where both greeting lines are indented prints only “Goodbye.”
- With age 8, a program where only the first line is indented still prints “Welcome in!”
- Indent every line that belongs to the if, elif, or else above it — four spaces is the standard

# The two most common if mistakes

- Forgetting the colon: an if line without : gives a syntax error — every if, elif, and else line ends with a colon
- Mixing indentation: some lines two spaces and others four, or tabs mixed with spaces, gives an indentation error
- Pick four spaces and use them everywhere — in IDLE, pressing Tab inserts spaces for you



# Hands-on exercise

- Write a program that stores your age and prints “You are a teen” if it is 13 or older, and “You are not a teen yet” otherwise
- Extend it with elif: under 13 → “Child”, 13 to 19 → “Teen”, 20 or older → “Adult” — test it with three different ages
- Deliberately remove the colon from an if line, read the error, then put the colon back and confirm the program works

# Key takeaways

- if runs its indented block only when its condition is true
- else covers every other case, with no condition of its own
- elif chains choices in order; the first true condition wins, and only it runs
- Every if / elif / else line ends with a colon; its block is indented four spaces
- Indentation is meaning in Python — same lines, different indent, different program



# Take the quiz

Take the Lesson 9 quiz — 5 quick questions to check what you learned.

# Next lesson

Loops: Repeating Without Retyping