How long until the weekend?

# Syntax checklist

If you are faced with an **error message**, read it carefully and try to fix the problem.

Use the list below to check for common errors (and tick ✓ if you find yours).

|  | misspelled if, elif, or else (this includes using capitals) |
| --- | --- |
|  | forgot the colon : at the end of a line containing if, elif, or else |
|  | neglected to **indent** statements in the if–block, elif–block, or else–block |
|  | indented if, elif, or else by mistake |
|  | used = instead of == in a condition, to check if two values are equal |
|  | used quotes around the name of a variable |
|  | forgot to use quotes around a string literal (like "Monday") |

# Task

## Step 1

**Open** this **incomplete** program (the-cc.io/py-week-2) in your development environment:

| 1  2  3  4  5  6  7 | print("What day is it today?")  day = int(input())  if day <= 4:  print("It’s a weekday")  remaining = .  else:  print("It’s the weekend!") |
| --- | --- |

Remember: The program uses an integer for each day of the week, ranging from 0 for Monday to 6 for Sunday. Here’s a handy reference:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| --- | --- | --- | --- | --- | --- | --- |
| Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Sunday |

## Step 2

**Complete** line 5, so that the value of remaining is the number of days left until the weekend (including the current day).

Tip: The current day is an integer, ranging from 0 to 4 in the case of weekdays. Saturday is day 5. Do the maths!

## Step 3

**Insert** the line below in your program, wherever you believe it is appropriate to display the value of remaining days to the user.

|  | print(remaining, "days until the weekend") |
| --- | --- |

Tip: Make sure this message is displayed only in the case that the day is a weekday.

| **Example** |  |
| --- | --- |
| Note: Use this example to check your program. Given the input you see in this sample interaction, this is the output your program should produce. | |
| The program displays a prompt and waits for keyboard input. | What day is it today? |
| The user types in a reply. | 1 |
| The program identifies the type of day and displays the days remaining until the weekend. | It’s a weekday  4 days until the weekend |

## Step 4

**Extend** the program so that Friday (day 4) is treated differently, with a different message displayed than for the rest of the weekdays.

Tip: You will need to use if-elif-else, since there will now be three branches in your program.

| **Example** |  |
| --- | --- |
| Note: Use this example to check your program. Given the input you see in this sample interaction, this is the output your program should produce. | |
| The program displays a prompt and waits for keyboard input. | What day is it today? |
| The user types in a reply. | 4 |
| The program identifies the particular day and displays a custom message. | It’s Friday  Just a day left until the weekend |

# Explorer task - The actual day of the week

Instead of asking the user to type the current day of the week, the program can retrieve the actual day, using the datetime module.

**Replace** the lines that receive user input (marked with -) with the lines that retrieve the current day of the week (marked with +).

| -  - | print("What day is it today?")  day = int(input()) |
| --- | --- |
| +  + | from datetime import datetime  day = datetime.now().weekday() |

Resources are updated regularly - the latest version is available at: [the-cc.io/curriculum](http://the-cc.io/curriculum).



This resource is licensed by the [Raspberry Pi Foundation](https://www.raspberrypi.org/) under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International licence. To view a copy of this license, visit, see [creativecommons.org/licenses/by-nc-sa/4.0/](https://creativecommons.org/licenses/by-nc-sa/4.0/).