As seasons roll on by

# Worked example

This program accesses a list called days, which contains the names of days, to display two *specific* items (those with index 5 and 6).

| 1  2  3  4  5  6 | days = ["Monday", "Tuesday",  "Wednesday", "Thursday",  "Friday", "Saturday", "Sunday"]  print("These are the weekend days:")  print(days[5])  print(days[6]) |
| --- | --- |

# Worked example

This program accesses a list called days, which contains the names of days, to display the name of the current day and the day *before* that.

| 1  2  3  4  5  6  7  8  9  10  11  12 | days = ["Monday", "Tuesday",  "Wednesday", "Thursday",  "Friday", "Saturday", "Sunday"]   | print("What day is it today?")  day = int(input()) | Today as integer . | | --- | --- |  | today = days[day] | Today . | | --- | --- |  | if day > 0:  yesterday = days[**day-1**]  else:  yesterday = days[6] | Yesterday . | | --- | --- |   print("Today it is", today)  print("Yesterday it was", yesterday) |
| --- | --- | --- | --- | --- | --- | --- | --- |

# Task 1 - Summer

## Step 1

**Open** this **incomplete** program (the-cc.io/py-months-1) in your development environment:

| 1  2  3  4  5  6  7  8  9 | months = ["January", "February",  "March", "April", "May",  "June", "July", "August",  "September", "October", "November",  "December"]  print("These are the summer months:")  print(months[ ])  print(months[ ])  print(months[ ]) |
| --- | --- |

## Step 2

**Complete** lines 7, 8, and 9 with an integer, so that the program displays the names of the summer months.

Tip: You are used to numbering months starting from 1 for January. However, list items in Python are numbered starting from 0. You need to take that into account.

| **Example** |  |
| --- | --- |
| Note: Use this example to check your program. This is the output your program should produce. | |
| The program displays the names of the summer months. | These are the summer months:  June  July  August |

# Task 2 - A name for a number

## Step 1

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

| 1  2  3  4  5  6  7  10 | months = ["January", "February",  "March", "April", "May",  "June", "July", "August",  "September", "October", "November",  "December"]  print("What month is it? (1-12)")  month = int(input())  print("It is", months[ ]) |
| --- | --- |

## Step 2

**Complete** line 10 with an expression, so that the program displays the name of the current month.

Tip: You are used to numbering months starting from 1 for January. However, list items in Python are numbered starting from 0. You need to take that into account.

| **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 month is it? (1-12) |
| The user types a reply. | 1 |
| The program displays the name of the month. | It is January |

| **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 month is it? (1-12) |
| The user types a reply. | 12 |
| The program displays the name of the month. | It is December |

# Task 3 - Seasons

## Step 1

**Open** this **incomplete** program (the-cc.io/py-months-3) in your development environment:

| 1  2  3  4  5  6  7  8  9  10  11  12  13 | seasons = ["Winter", "Spring",  "Summer", "Autumn"]  print("What month is it? (1-12)")  month = int(input())  if :  season = 0  elif :  season = 1  elif :  season = 2  else:  season = 3  print("It is", seasons[season]) |
| --- | --- |

## Step 2

**Complete** lines 5, 7, and 9 with a Boolean expression (a condition) that checks the value of the month variable and assigns an appropriate value to season.

Tip: The numbering of seasons is arbitrary. Looking at the seasons list, numbering ranges from 0 for winter to 3 for autumn.

Tip: The condition required in line 5 should check the month variable, to determine if it corresponds to a winter month (season 0). The numbers for winter months are not consecutive (1, 2, and 12), so you will need to use or in the condition.

# Explorer task - The actual month

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

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

| -  - | print("What month is it? (1-12)")  month = int(input()) |
| --- | --- |
| +  + | from datetime import datetime  day = datetime.now().month |

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/).