

Build a Time Calculator Project

Write a function named `add_time` that takes in two required parameters and one optional parameter:

- a start time in the 12-hour clock format (ending in AM or PM)
- a duration time that indicates the number of hours and minutes
- (optional) a starting day of the week, case insensitive

The function should add the duration time to the start time and return the result.

If the result will be the next day, it should show (next day) after the time. If the result will be more than one day later, it should show (n days later) after the time, where "n" is the number of days later.

If the function is given the optional starting day of the week parameter, then the output should display the day of the week of the result. The day of the week in the output should appear after the time and before the number of days later.

Below are some examples of different cases the function should handle. Pay close attention to the spacing and punctuation of the results.

```
add_time('3:00 PM', '3:10')  
# Returns: 6:10 PM
```

```
add_time('11:30 AM', '2:32', 'Monday')  
# Returns: 2:02 PM, Monday
```

```
add_time('11:43 AM', '00:20')  
# Returns: 12:03 PM
```

```
add_time('10:10 PM', '3:30')  
# Returns: 1:40 AM (next day)
```

```
add_time('11:43 PM', '24:20', 'tueSday')  
# Returns: 12:03 AM, Thursday (2 days later)
```

```
add_time('6:30 PM', '205:12')  
# Returns: 7:42 AM (9 days later)
```

Do not import any Python libraries. Assume that the start times are valid times. The minutes in the duration time will be a whole number less than 60, but the hour can be any whole number.

Tests

1. Calling `add_time('3:30 PM', '2:12')` should return '5:42 PM'.
2. Calling `add_time('11:55 AM', '3:12')` should return '3:07 PM'.
3. Expected time to end with '(next day)' when it is the next day.
4. Expected period to change from AM to PM at 12:00.
5. Calling `add_time('2:59 AM', '24:00')` should return '2:59 AM (next day)'.
6. Calling `add_time('11:59 PM', '24:05')` should return '12:04 AM (2 days later)'.
7. Calling `add_time('8:16 PM', '466:02')` should return '6:18 AM (20 days later)'.
8. Expected adding 0:00 to return the initial time.
9. Calling `add_time('3:30 PM', '2:12', 'Monday')` should return '5:42 PM, Monday'.
10. Calling `add_time('2:59 AM', '24:00', 'saturday')` should return '2:59 AM, Sunday (next day)'.
11. Calling `add_time('11:59 PM', '24:05', 'Wednesday')` should return '12:04 AM, Friday (2 days later)'.
12. Calling `add_time('8:16 PM', '466:02', 'tuesday')` should return '6:18 AM, Monday (20 days later)'.