Class Clock ✔✔✔ ☺ 🎓

This assignment **closed** December 1, 2024 at 23:15.

The *time of day* can be represented by the hours ($0 \leq h < 24$) and minutes ($0 \leq m < 60$).

Assignments

1. Write a class `Clock`, that can be used to create, add and print time objects. The class should at least contain:

   a. a constructor to create an instance by giving the hours and minutes (in that order). Both values might exceed `60`, but should be normalized to an actual time. Both values should be stored as object properties.

   E.g.: `06:62` should become `07:02`, `25:00` should become `01:00`.

   b. the method `__eq__(self, other: object) -> bool` to check if `self` and `clock` represent the same time. We provided the base implementation, since `other` in signature must have the type `object`. Complete the method!

   Hint: This allows you to use `==` on two clock objects. Python Docs

   c. the method `__add__(self, other: "Clock") -> "Clock"` to add the `clock` object and the `self` Clock object.

   Hint: This allows you to use `+` on two clock objects. Python Docs

   d. the method `add_minutes(self, minutes: int) -> "Clock"` to add the `minutes` to the current Clock object.

   e. a method `__str__(self) -> str` to transform the object into a proper `String` representation.

2. Test your `Clock` using the script. Generate random `Clock` objects and test your methods' implementations.

Hints

- The methods for adding should always return a **new** clock instance and not modify the `self` instance.
- The type-hint `"Clock"` has to be in quotes, since it is not defined by the time the type checker checks the functions. The quotes delay the type checking till the end of the script, at which point `Clock` is defined.

Template files

Get all files in an archive templates.zip or templates.tgz.

`clock.py`

Miit Dholakia | | | |