

## Week 9 Exercises to be reviewed on November 3

Please use the i206 style guidelines found here:

<http://blogs.ischool.berkeley.edu/i206f10/assignment-style-guidelines/>

Make sure to add the header to your files like so:

```
#!/usr/bin/env python

__author__ = 'Ariel Chait'
__email__ = 'ariel@ischool.berkeley.edu'
__python_version__ = '3.2.1'
__can_anonymously_use_as_example__ = True

# Code starts here
```

Use i90 as the class number the week as the assignment name and leave out the document type so this assignment would be saved as **i90\_week9\_ariel.py**

1. Create a new class called **Time**

Time will have three attributes:

Hour

Minute

Second

You should be able to instantiate a time with:

```
>>> current_time = Time()
```

Creates an instance of the time class and sets all of the attributes to 00

You should also be able to instantiate it with specific values:

```
>>> current_time = Time(11,45,05)
```

Printing a time object should look like:

```
>>> print(current_time)
```

```
>>> 11:45:05
```

Adding two time objects should add the hours, minutes, and seconds

```
>>> t1 = Time(1,45,03)
```

```
>>> t2 = Time(5,10,40)
```

```
>>> t3 = t1 + t2
```

```
>>> print(t3)
```

```
>>> 6:55:43
```

The Time class should have a method to turn 24 hour format into 12 hour format and print am or pm depending on the value of the hour.

Guiding comments for the method are as follows:

```
def amPm(self):
    # check if the hour is greater than 12
        # if it is, subtract 12 from the hour
        # print the time and the string 'pm' after it
    # else
        # print the time and the string 'am' after it
```

Remember this a method of the class time so it should be defined inside of the class definition.

Output should look like:

```
>>> t1 = Time(20,10,50)
>>> t1.amPm
>>> 10:10:50 pm
```

```
>>> t2 = Time(8,15,40)
>>> t2.amPm
>>> 8:15:40 am
```

To complete this assignment, define the Time class and write a few lines of code to test the functionality of instantiation, printing, addition, and the amPm method.