Declaring variables and variable annotations

Don't mind that word "Real" in the flowchart in exercise 4! 😉

You're supposed to declare the variables, before assigning new values. So, as an example, you would declare the variable `distance8Hours` like this:

```
distance8Hours = 0.0
```

Which declares `distance8Hours` variable as a float variable (note the decimals) and not an integer. We would then know that this variable is intended to be of type float.

Please note that skipping declaring variables beforehand is certainly possible, but the programmer will keep better track of which type a certain variable is supposed to be. Not declaring variables may lead to variables being coded as different types, unintentionally, during the course of writing a script, for instance.

I would put it like this in the beginning of the program with a comment before:

```
# Variables for distances. The data type will be float for each one.

distance5Hours = 0.0
distance8Hours = 0.0
distance12Hours = 0.0
```

What's been done on these lines above is that each variable is **declared** as well as **initialized**, as a value is assigned to them. This is the way it's normally done in Python when it comes to declare variables. A **name** of the variable and a **value** assigned to it which also gives the data type (by its value) at the same time.

From Python version 3.6+ it's possible to annotate a variable like below, which is like a hint about the desired data type later on in the program.

```
# Below syntax means that the variable distance12Hours

# is expected to be a float

distance12Hours : float

# Later on, when it will be assigned a value.

# Even if it is still possible to assign

# another data type to it along the way, it is not recommended!

# Stick to the data type it is expected to have!

# This data type is, again, given by the syntax above.
```

Below follows an example of the use of declaration of variables.

```
# declare a variable named "distance8Hours"
# and annotate it to have the desired
# data type = float
# ( note the colon needed )
distance8Hours : float

# assigning a value to distance8Hours
distance8Hours = 4.56;

# print the value of distance8Hours
print(distance8Hours);
```

Another exmple, a variable named `fname` for a String value (String represents text 👍):

```
fname : str
```

Using these annotations, example above – `distance12Hours` and `fname`, means that the variables will not be initialized (no values assigned) but accepted by the Python interpreter at runtime and we will have the variables declared before values are assigned.

You can read some more about these annotations here:

PEP 526

PEP = Python Enhancement Proposals