AL NAFI,
A company with a focus on education,
wellbeing and renewable energy.

Python Alpha 500

Python Comprehensions

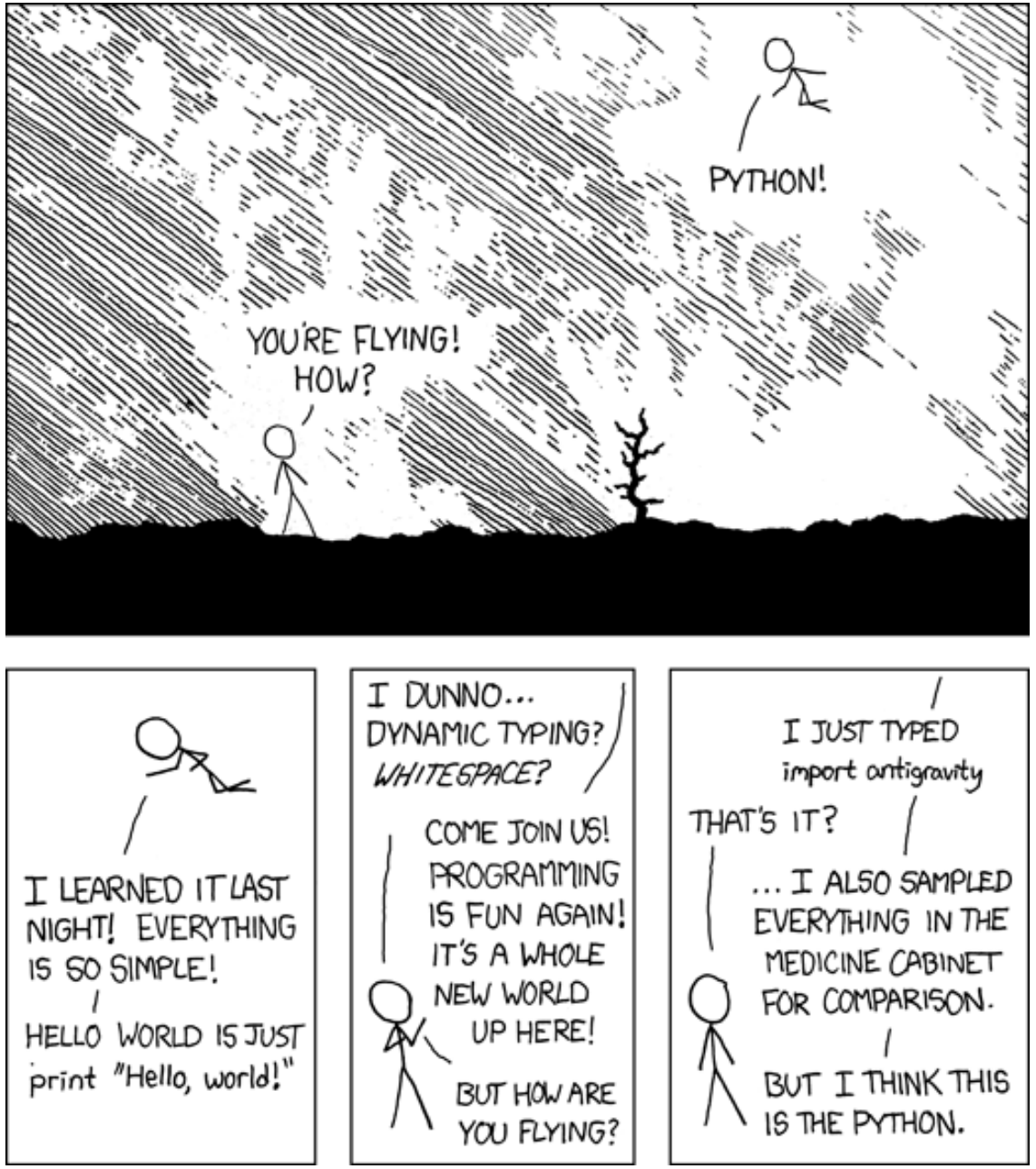
Supplication at the start of the prayer after takbeer

اللّهُمَّ رَبَّ جِبْرائيل ، وَميكائيل ، وَإِسْرافيل، فاطِرَ السَّمواتِ وَالأَرْض ، عالِمَ الغَيْبِ وَالشَّهادَةِ أَنْتَ تَحْكمُ بَيْنَ عِبادِكَ فيما كانوا فيهِ يَخْتَلِفون. اهدِني لِما اخْتُلِفَ فيهِ مِنَ الْحَقِّ بِإِذْنِك ، إِنَّكَ تَهْدي مَنْ تَشاءُ إِلى صِراطٍ مُسْتَقيم

O Allah, Lord of Jibra-eel, Meeka-eel and Israfeel (great angles), Creator of the heavens and the Earth, Knower of the seen and the unseen. You are the arbitrator between Your servants in that which they have disputed. Guide me to the truth by Your leave, in that which they have differed, for verily You guide whom You will to a straight path. Reference: Muslim 1/534

Study, Rinse and Repeat

- Please subscribe to our YouTube Channel to be on top of your studies.
- Please logon to our website https://alnafi.com/login/
- Use your username and password to logon
- Please keep an eye on zone@alnafi.com emails
- Please review the videos of 30 minutes daily
- Please review the notes daily.
- Please take time for clearing up your mind and reflect on how things are proceeding in your studies.



Comprehensions

The Python language has a couple of methods for creating lists and dictionaries that are known as comprehensions. There is also a third type of comprehension for creating a Python known as comprehension.

You will find that the comprehension constructs build on the knowledge you have acquired from the previous classes as they contain loops and conditionals themselves.

Python's list comprehension is an example of the language's support for *functional programming concepts*. There's plenty of debate about the **best way** to develop program code: either procedurally, using functional programming techniques, or using object orientation.

List comprehensions

List comprehensions in Python are very handy. They can also be a little hard to understand when and why you would use them.

List comprehensions tend to be harder to read than just using a simple for loop as well.

You may want to review the looping class before you continue. If you are ready, then we'll spend some time looking at how to construct list comprehensions and learn how they can be used.

List comprehensions in one word

A list comprehension is basically a one line for loop that produces a Python list data structure. Here's a simple example:

```
x = [i for i in range(5)]
x
```

Breaking the code down:

1. The code returns integers starting at 0 and ending at 4.
2. This can be useful if you need to create a list very quickly.

Another list comprehensions examples

```
if [i for i in line if "SOME TERM" in i]:
```

Breaking the code down:

1. This code is parsing a file and looking for something in particular.

Casting strings into integers ☺

```
x = ['1', '2', '3', '4', '5']
y = [int(i) for i in x]
y
```

Breaking the code down:

1. Mixing functions into the mix in this example.
2. In this code we are applying a function to every element in a list.
3. Such as when you need to cast a bunch of strings into integers.

Strip the list

This sort of thing comes up more often than you'd think. I have also had to loop over a list of strings and call a string method, such as strip on them because they had all kinds of leading or ending white space:

```
myStrings = [s.strip() for s in myStringList]
```

Nested list comprehensions

There are occasions where you need to create a nested list comprehension. Its like flatten multiple lists into one.

list = [[1,2,3], [4,5,6], [7,8,9]]

[num for elem in list for num in elem]

Dictionary Comprehensions

Dictionary comprehension is a method for transforming one dictionary into another dictionary. During this transformation, items within the original dictionary can be conditionally included in the new dictionary and each item can be transformed as needed.

Dictionary Comprehensions examples

```
print( {i: str(i) for i in range(5)} )
```

Breaking the code down:

1. Basically it is creating an integer key and string value for each item in the range.

Dictionary Comprehensions example continued…

```
my_dict = {1:"dog", 2:"cat", 3:"horse"}
print( {value:key for key, value in my_dict.items()} )
```

Parking lot item as when we learn about class we will know more inshAllah.

More examples, assignments, code snippets, and projects will be shared in Python Beta so that what you learned in Python Primer and Python Alpha you are able to retain it inshAllah.

So don’t confuse yourself with list comprehensions.

Set Comprehensions

Set comprehensions are created in much the same way as dictionary comprehensions. Now a Python set is much like a mathematical set in that it doesn't have any repeated elements. You can create a normal set like this:

```
my_list = [1, 2, 2, 3, 4, 5, 5, 7, 8]
my_set = set(my_list)
my_set
```

Breaking the code down:

1. As you can see in this code a call has been set to remove the duplicates using set.

Another set example

```
my_list = [1, 2, 2, 3, 4, 5, 5, 7, 8]
my_set = {x for x in my_list}
my_set
```

Breaking the code down:

1. Basically we changed the square brackets that a list comprehension uses to curly braces that the dictionary comprehensions utilizes. So there are multiple ways of using a set function.

Wrapping up

Parking lot item as when we learn about class we will know more inshAllah.

More examples, assignments, code snippets, and projects will be shared in Python Beta so that what you learned in Python Primer and Python Alpha you are able to retain it inshAllah.

So don’t confuse yourself with list comprehensions. Just try to use your imagination and fiddle around as more is on its way inshAllah.

جزاك اللهُ

To ask questions, please logon to the portal https://alnafi.com/login/ and use your username and password. From within the portal you can ask questions. We will only answer questions if they are coming through the portal and not on email.

For any other queries please reach out on info@alnafi.com