

Forum:[Hack Forums](#)

Thread: [Unsure why this loop doesn't work \[Solved\]](#)

yo yo yo - February 21, 2015, 4:28 am

Hi, so I'm making something in Python which creates XML files. I'm about half way through a codecademy course on Python so I don't have much experience with it.

Here's my issue, my code works when I set it up like this:

```
Code offender = doc.createElement('offender')
offenderName = doc.createElement('offenderName')
offenderDoB = doc.createElement('offenderDoB')
offenderSex = doc.createElement('offenderSex')
offenderAddress = doc.createElement('offenderAddress')
```

But I thought this was pretty ugly and I would have to use these variables elsewhere so I created a list to make things nicer:

```
Code offenderDetails = ['offender', 'offenderName', 'offenderDoB', 'offenderSex', 'offenderAddress']
```

for details in offenderDetails:

```
    details = doc.createElement(details)
```

But this doesn't have the same affect as above, I get the error:

Traceback (most recent call last):

```
File "C:\Python34\Scripts\program 1.py", line 126, in <module>
```

```
    offender.appendChild(offenderName)
```

```
NameError: name 'offenderName' is not defined
```

I'm fairly certain this is how for loops work in Python, although I am used to C/C++/Java.

lockhead - February 21, 2015, 4:39 am

I haven't done any work with python in a while, but this looks correct to me. Can you release the whole code? I think maybe the loop worked and the error occurred later.

yo yo yo - February 21, 2015, 4:51 am

(Feb 21, 2015 05:39 AM)lockhead Wrote: [\[CITE\]](#) I haven't done any work with python in a while, but this looks correct to me. Can you release the whole code? I think maybe the loop worked and the error occurred later.

Here ya go:

-snip-

I can't figure out what the error could be, I've tried moving the location of the for loop/list. I've also tried this kind of for loop:

```
Code for i in range(0, len(x)):
```

```
    print x[i]
```

Same error.

bytecoder - February 21, 2015, 6:50 am

1. Use `raw_input` instead of `input`. (Otherwise you get an error)

Quote:input() actually evaluates the input as Python code. I suggest to never use it. raw_input() returns the verbatim string entered by the user.

2. Code Error in line 142

Code #offenderName.appendChild(setOffenderName) # Wrong
offender.appendChild(setOffenderName) # Right

3. Your while is lacking of code

Wrong:

```
Code offenderDetails = ['offender', 'offenderName', 'offenderDoB', 'offenderSex', 'offenderAddress']
```

for details in offenderDetails:

```
    details = doc.createElement(details)
```

Right:

Code for item in offenderDetails:

```
    print "Create:", item
    details = doc.createElement(item)
    offender.appendChild(details)
```

Otherwise it's quite good.

Your code:

pastebin.com/HnEdVJNN

lockhead - February 21, 2015, 6:52 am

I'm sorry I can't find anything wrong with the code (especially the part that's not working). I don't know any of the xml related content but the while-loop looks correct. Maybe try creating and appending the element in the while loop like this:

Code for x in offenderDetails:

```
    x = doc.createElement(x)
    offender.appendChild(x)
```

yo yo yo - February 21, 2015, 7:01 am

(Feb 21, 2015 07:50 AM)bytecoder Wrote: [\[CITE\]](#) 1.

2.

3.

1. I'm using Python 3, I don't think raw_input is available anymore:

Code NameError: name 'raw_input' is not defined

2. What I have works, the XML file looks correct without making this change.

3. I'm getting errors with this, I'm pretty sure because I'm using Python 3 the syntax is different. It doesn't like this bit:

Code print "Create:", item

When I remove it I still get the same error:

Code NameError: name 'offenderName' is not defined

And thanks :)

(Feb 21, 2015 07:52 AM)lockhead Wrote: [\[CITE\]](#) I'm sorry I can't find anything wrong with the code (especially the part that's not working). I don't know any of the xml related content but the while-loop looks correct. Maybe try creating and appending the element in the while loop like this:

Code for x in offenderDetails:

```
x = doc.createElement(x)
```

```
offender.appendChild(x)
```

Same error with this too. Confusing :/

bytecoder - February 21, 2015, 7:11 am

Ah Python 3 - Sorry, I must have overread it. There should work input() correct.

Can you give me the exact error output?

yo yo yo - February 21, 2015, 7:19 am

(Feb 21, 2015 08:11 AM)bytecoder Wrote: [\[CITE\]](#) Ah Python 3 - Sorry, I must have overread it. There should work input() correct.

Can you give me the exact error output?

Enter the offender's name: df

Traceback (most recent call last):

```
File "C:\Python34\Scripts\program 1.py", line 132, in <module>
```

```
offenderName.appendChild(setOffenderName)
```

NameError: name 'offenderName' is not defined

```
>>>
```

```
Code offenderDetails = ['offenderName', 'offenderDoB', 'offenderSex', 'offenderAddress']
```

```
for x in offenderDetails:
```

```
x = doc.createElement(x)
```

```
offender.appendChild(x)
```

```
.....
```

```
offenderName = doc.createElement('offenderName')
```

```
offenderDoB = doc.createElement('offenderDoB')
```

```
offenderSex = doc.createElement('offenderSex')
```

```
offenderAddress = doc.createElement('offenderAddress')
```

```
offender.appendChild(offenderName)
```

```
offender.appendChild(offenderDoB)
```

```
offender.appendChild(offenderSex)
```

```
offender.appendChild(offenderAddress)
```

```
.....
```

The code which is edited out works, but code at the top gives the error when it should do the same thing.

offenderName should be defined in the for loop.

Misingnoglic - February 21, 2015, 1:24 pm

1. Try not modifying x in the loop (e.g. use another variable)

2. Try creating all the elements first, then doing the append child's

MoonCakeRuler - February 21, 2015, 1:37 pm

You're mutating your loop variable in the loop body. Just change that variable. So it can be like:

Code for detail in offenderDetails:

```
x = doc.createElement(detail)
offender.appendChild(x)
```

That should be it I guess. I'm assuming you don't need the 'x' other than for appending. If you do, just add them to a list/dictionary outside the loop.

Hope that helps.

yo yo yo - February 21, 2015, 2:24 pm

(Feb 21, 2015 02:24 PM) Misingnolic Wrote: [\[CITE\]](#) 1. Try not modifying x in the loop (e.g. use another variable)

2. Try creating all the elements first, then doing the append child's

(Feb 21, 2015 02:37 PM) MoonCakeRuler Wrote: [\[CITE\]](#) You're mutating your loop variable in the loop body. Just change that variable. So it can be like:

Code for detail in offenderDetails:

```
x = doc.createElement(detail)
offender.appendChild(x)
```

That should be it I guess. I'm assuming you don't need the 'x' other than for appending. If you do, just add them to a list/dictionary outside the loop.

Hope that helps.

I get the same error. I even get the error when I append each one on it's own line and only try and create the elements using a loop.

Here's an updated version I'm working on: -snip-

If anyone can get it to run please post how ?

Lines 114-120 (commented out) don't work but the bit below does.

dberube - February 21, 2015, 3:17 pm

In this piece of code, the variable "offenderName" was never declared.

```
Code offenderDetails = ['offenderName', 'offenderDoB', 'offenderSex', 'offenderAddress']
```

for detail in offenderDetails:

```
detail = doc.createElement(detail)
offender.appendChild(detail)
```

```
inputOffenderName = input("Enter the offender's name: ")
```

```
setOffenderName = doc.createTextNode(inputOffenderName)
```

```
offenderName.appendChild(setOffenderName)
```

Because you use variables that use those names later in your code, you have to dynamically define the variable.

That is NOT RECOMMENDED, but if you really (really) want to know:

Spoiler [\(Click to View\)](#)

Last warning! [\(Click to View\)](#)

```
Code offenderDetails = ['offenderName', 'offenderDoB', 'offenderSex', 'offenderAddress']
```

for detail in offenderDetails:

```
element = doc.createElement(detail)
locals()[detail] = element    #This create a new variable in the local scope
offender.appendChild(element)
```

If statements:

Code if inputoffenderSex == "m" or "f" or "Male" or "Female" or "M" or "F" or "male" or "female":

#this gets interpreted to

if inputoffenderSex == "m" or True or True or True or True or True or True or True

Works like that:

Code # Pythonic way

if inputoffenderSex in ("m","f","Male","Female","M","F","male","female"):

#OR

if inputoffenderSex == "m" or inputoffenderSex == "M" #etc...

MoonCakeRuler - February 21, 2015, 4:11 pm

Like dberube said, just changing the if statement would make your script work. And wow man, that locals() kung-fu. xD

Another similar (rather ugly) way would be to store your created elements in a dictionary or similar collection structure. I tried it out because I have nothing better to do but you'll then need to change occurrences of offenderName/Address/Dob/Sex, etc. to something like nameofdict['offenderName']. Given your current script, it isn't exactly as economical as one would expect; could be different if you need to add a larger number of offender details. Might not be the most Pythonic way to do this but here's the script: <http://pastebin.com/scgAVZPJ>

Everything dberube said about the if statements still apply. Good luck with the course. ;D

MR.FREE - February 21, 2015, 4:56 pm

I havent read thru all the post to see if anyone fixed this yet, but try this.

```
Code offenderDetails = ['offender', 'offenderName', 'offenderDoB', 'offenderSex', 'offenderAddress']
```

for details in offenderDetails:

```
    print ("{} = doc.createElement('{}')".format(details,details))which should output
```

Code >>>

```
offender = doc.createElement('offender')
```

```
offenderName = doc.createElement('offenderName')
```

```
offenderDoB = doc.createElement('offenderDoB')
```

```
offenderSex = doc.createElement('offenderSex')
```

```
offenderAddress = doc.createElement('offenderAddress')
```

```
>>>
```

Zer0 Cool - February 21, 2015, 5:55 pm

for future reference, you should split up your code into functions and or modules if it's going to be really long, each function should do 1 thing. makes it much more readable

MR.FREE - February 21, 2015, 6:39 pm

(Feb 21, 2015 06:55 PM)Zer0 Cool Wrote: [\[CITE\]](#) for future reference, you should split up your code into functions and or modules if it's going to be really long, each function should do 1 thing. makes it much more readable

I'm sorry but, where do you see enough code to need a function?

Zer0 Cool - February 21, 2015, 9:56 pm

(Feb 21, 2015 07:39 PM)MR.FREE Wrote: [\[CITE\]](#) I'm sorry but, where do you see enough code to need a function?there's almost 200 lines, that's waaay more than enough to merit a function.

you seem to think adding functions is something you should only do if you have an enormous amount of code, there's enough code here to warrant at least 3 functions, it's pretty much SOP to break your code into functions otherwise it can get really verbose

yo yo yo - February 22, 2015, 12:18 am

(Feb 21, 2015 04:17 PM)dberube Wrote: [\[CITE\]](#) In this piece of code, the variable "offenderName" was never declared.

```
Code offenderDetails = ['offenderName', 'offenderDoB', 'offenderSex', 'offenderAddress']
```

for detail in offenderDetails:

```
    detail = doc.createElement(detail)
    offender.appendChild(detail)
```

```
inputOffenderName = input("Enter the offender's name: ")
```

```
setOffenderName = doc.createTextNode(inputOffenderName)
```

```
offenderName.appendChild(setOffenderName)
```

Because you use variables that use those names later in your code, you have to dynamically define the variable.

That is NOT RECOMMENDED, but if you really (really) want to know:

Spoiler [\(Click to View\)](#)

Last warning! [\(Click to View\)](#)

```
Code offenderDetails = ['offenderName', 'offenderDoB', 'offenderSex', 'offenderAddress']
```

for detail in offenderDetails:

```
    element = doc.createElement(detail)
    locals()[detail] = element    #This create a new variable in the local scope
    offender.appendChild(element)
```

If statements:

```
Code if inputoffenderSex == "m" or "f" or "Male" or "Female" or "M" or "F" or "male" or "female":
```

```
#this gets interpreted to
```

```
if inputoffenderSex == "m" or True or True or True or True or True or True or True
```

Works like that:

```
Code # Pythonic way
```

```
if inputoffenderSex in ("m","f","Male","Female","M","F","male","female"):
```

```
#OR
```

```
if inputoffenderSex == "m" or inputoffenderSex == "M" #etc...
```

```
+3
```

Damn... I was kinda hoping that I missed a colon or something haha. Thanks for the reply, I'll look into this further, and probably stick to the simple way for now.

And thanks for the tip with the multiple conditions, I had changed it to the second way you posted but the first way is much shorter.

(Feb 21, 2015 06:55 PM)Zer0 Cool Wrote: [\[CITE\]](#) for future reference, you should split up your code into functions and or modules if it's going to be really long, each function should do 1 thing. makes it much more readable

This is pretty much the first program in Python I've ever made, and the first time using XML, so I've kept it simple by using more code.