# World Class Machine Learning

Machine Learning Hub

Menu

ABOUT US

Menu

## Python XML Parsing Example

Posted on June 9, 2018 by Jonathan Steele

Hello Everyone!

Today, I want to share how to parse through XML files in Python. We will use a simple example and then use a real-life example from my experience.

### Resources

1. General Information about XML (W3 Schools)
2. Tutorial I used for Python XML DOM parsing (TutorialsPoint)
3. Setting up a Python Setup (ML Hello World)
4. Example files (W3 Schools)
    1. Direct Note Link (Alt-Click link to Download)
    2. Direct CD Catalog Link (Alt-Click link to Download)

### Trivial Example

I'm assuming you have some things figured out already:

1. You know about XML generally (If not, follow Resource Link 1)

2. You already have Python working on your Computer (If not, follow Resource Link 3)

Now, let's start.

- Download the sample XML files from Resource Link 4.1 and 4.2.
- Get those sample XML files in a folder by themselves named "xml_example" or something like that.
- View the XML file to get a feel for what it's like.

note.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

- In that same folder, create a file named "parseNotes.py" and open it in a text editor that you prefer.
- Enter in this code: This will be a little rote your first time, but we will use it a little differently in a second and you can always examine the documentation and other tutorials to get a deeper feel.

y-journal-1.xml | missionary-journal-2.xml | parseNotes.py | parseCatalog.py

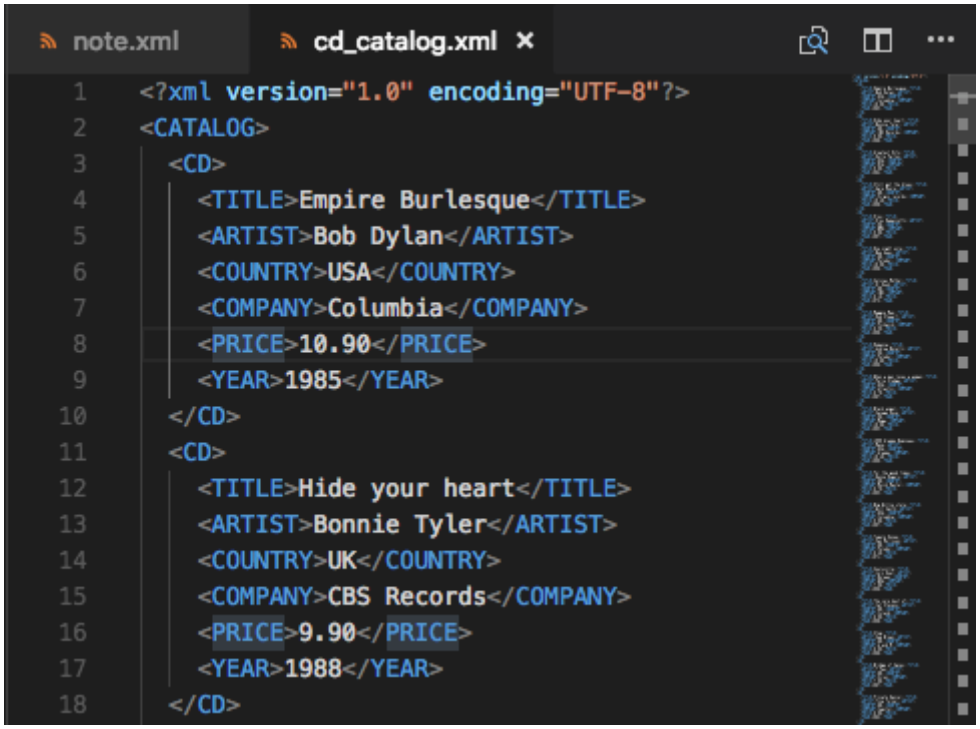
```python
from xml.dom.minidom import parse
import xml.dom.minidom

# Open XML document using minidom parser
DOMTree = xml.dom.minidom.parse("note.xml");
noteRootNode = DOMTree.documentElement

# get the elements
to = noteRootNode.getElementsByTagName("to")[0].childNodes[0].data
fromText = noteRootNode.getElementsByTagName("from")[0].childNodes[0].data
heading = noteRootNode.getElementsByTagName("heading")[0].childNodes[0].data
body = noteRootNode.getElementsByTagName("body")[0].childNodes[0].data

# display
print("============= Note(s) ============")
print("To: " + to)
print("From: " + fromText)
print("Heading: " + heading)
print("Body: " + body)
print("==================================")
```

## Intermediate Example

If you've skipped the Trivial Example and jumped to the Intermediate, make sure to grab the "cd_catalog.xml" file from Resource Link 4.2 and get your stuff in a project folder.

Let's go!

- Create a file name "parseCatalog.py"
- Add the necessary imports, as seen above.
- Read in the file "cd_catalog.xml" using MiniDom, as seen above.
- View the XML file to get a feel for how we will need to parse it.




- As you can see, there is a root node "CATALOG" that has a list of "CD"s
  - and each "CD" has a
    - TITLE
    - ARTIST
    - COUNTRY
    - COMPANY
    - PRICE
    - YEAR
- So let's make sure to parse our stuff in that way. But let's add one thing. Let's use a for loop to access the list of "CD"s so that we can read in as many or as few CDs as there are in the Catalog.
- Test out a For Loop on your CDs list by using code kind of like this:

parseCatalog.py × missionary-journal-1.xml missionary-journal-2.xml parseNotes.py

```python
from xml.dom.minidom import parse
import xml.dom.minidom

# Open XML document using minidom parser
DOMTree = xml.dom.minidom.parse("cd_catalog.xml")
catalogRootNode = DOMTree.documentElement

# get the list
cds = catalogRootNode.getElementsByTagName("CD")
numOfRows = 0
for cd in cds:
    numOfRows += 1

print("The Catalog has " + str(numOfRows) + " CDs in it. Counted via a FOR Loop.")
print("The Catalog has " + str(len(cds)) + " CDs in it. Counted via a LEN command.")
```

- Once you've figured out how to use the For Loops for parsing the XML lists, then you're ready to look at the whole list.

parseCatalog.py × missionary-journal-1.xml missionary-journal-2.xml parseNotes.py

```python
from xml.dom.minidom import parse
import xml.dom.minidom

# Open XML document using minidom parser
DOMTree = xml.dom.minidom.parse("cd_catalog.xml")
catalogRootNode = DOMTree.documentElement

print("============ CD(s) ============")
# get the list
cds = catalogRootNode.getElementsByTagName("CD")
for cd in cds:
    # collect data
    title = cd.getElementsByTagName("TITLE")[0].childNodes[0].data
    artist = cd.getElementsByTagName("ARTIST")[0].childNodes[0].data
    country = cd.getElementsByTagName("COUNTRY")[0].childNodes[0].data
    company = cd.getElementsByTagName("COMPANY")[0].childNodes[0].data
    price = cd.getElementsByTagName("PRICE")[0].childNodes[0].data
    year = cd.getElementsByTagName("YEAR")[0].childNodes[0].data

    # display
    print("================================")
    print("Title: " + title)
    print("Artist: " + artist)
    print("Country: " + country)
    print("Company: " + company)
    print("Price: " + price)
    print("Year: " + year)

print("================================")
```

- And that's all there is to it! You've parse the whole XML file!

## Conclusion

So far, we've done a couple trivial examples of how to parse XML files.

Next time, we can have a real-life example for parsing and sorting a journal file that was kept in XML. Until then!

## LANGUAGE

English

## RECENT POSTS



## CATEGORIES



## PAGES

English

© 2019 World Class Machine Learning | WordPress Theme by Superbthemes