CO_2 Box Model

We are going to make a box model to try and explain CO2 concentrations.

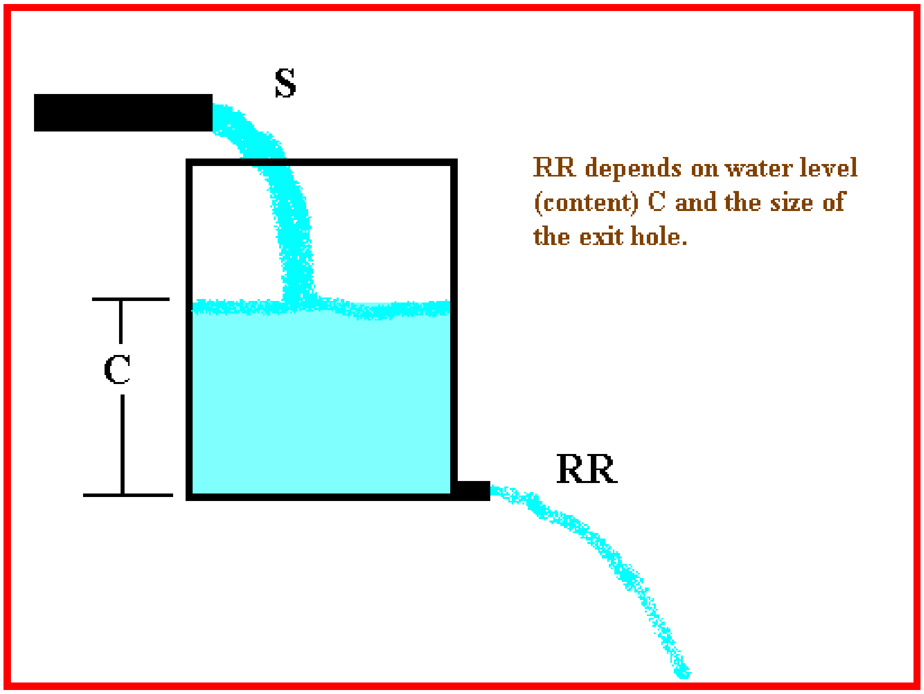
I am freely taking from this website http://serc.carleton.edu/quantskills/activities/co2_global.html and this website. http://serc.carleton.edu/quantskills/activities/co2_global.html Robert Mackay is the person behind this. http://www.atmosedu.com/

I have double checked the equations against the website already. But please make sure I have no typos.

We are going to treat the atmosphere as a simple box model. Here is a figure describing it. It is simple. Think about that we are adding CO2, it is building up in the atmosphere, and then it is leaving. The rate we are adding is increasing but it is leaving at a constant rate.

In [3]:
```
from IPython.display import Image
Image(url='http://www.atmosedu.com/physlets/GlobalPollution/waterbucket.gif')
```

Out[3]:


S
RR depends on water level (content) C and the size of the exit hole.
C
RR

this makes the equation

$$\frac{dC}{dt} = S - \frac{C}{\tau}$$

C is the Concentration of CO_2 in the atmospher in ppm.

S is the amount of CO_2 we are adding every year in ppm.

tau (τ) is is the residence time of CO_2 in years.

So basically we are saying the changing the CO_2 over time is equal to the amount we put in and how quickly it leaves.

Now we just need to solve it!

The answeres are from here. http://www.atmosedu.com/physlets/GlobalPollution/instructorNotes.htm

$$C = C_o e^{-t/\tau} + e^{-t/\tau} \int_0^t St' e^{-t'/\tau} dt'$$

I am not sure about the t-prime and I am looking into it. I think t is total time and t prime is at any time step. But we know the source term is changing and increasing over time. But this is the solution over time telling us the concentration C at any time in the atmosphere.

But we also need to know how much CO2 we are adding

$$S(t) = S_o e^{Rt}$$

$$S_o$$

is the initial addition to the atmosphere at time zero

R

is the rate of growth of emission

Before we solve the whole equation lets just solve for S(t).

So read in your mauna lao data

We will need datetime today

Final equation

As you will see below. The final solution we will solve is

$$C = C_o e^{-t/\tau} + \frac{S_o}{R+\frac{1}{\tau}} \left[e^{Rt} - e^{-t/\tau} \right]$$

$$\tau$$

tau is the lifetime of CO_2 in the atmosphere in years. We use 120.

In [2]:

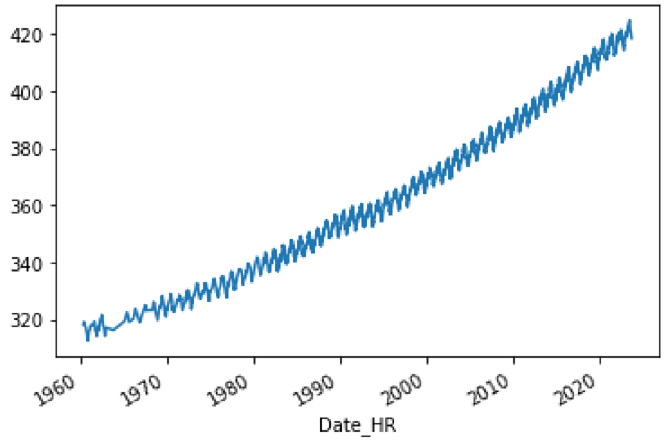
```
%matplotlib inline
import pandas as pd
import numpy as np
import matplotlib.pylab as plt
from scipy import stats
from matplotlib.backends.backend_pdf import PdfPages

import datetime
```

In [16]:

```
url='https://scrippsco2.ucsd.edu/assets/data/atmospheric/stations/flask_co2/daily/daily_fl
df_mlo=pd.read_csv(url,skiprows=69,usecols=[0,1,5,6],names=['Date','HR','Flag','CO2']\
                        ,parse_dates=[['Date','HR']]\
                        ,index_col='Date_HR',header=None)
df_mlo=df_mlo[df_mlo['Flag']==0].copy()
df_mlo['CO2'].plot()
```

Out[16]:

```
<AxesSubplot:xlabel='Date_HR'>
```



to make this work you will need elapsed time. This is a little tricky to make and takes a few tricks. lets run through them.

First you need a start time which is the first index. But since it is an index you need to use index to get the values out. to get the values out of an index you type this

In [17]:

```
df_mlo.index
```

Out[17]:

```
DatetimeIndex(['1960-03-30 12:00:00', '1960-03-31 16:00:00',
               '1960-04-24 08:00:00', '1960-08-18 10:00:00',
               '1960-10-06 14:48:00', '1960-12-01 10:50:00',
               '1961-01-04 12:50:00', '1961-01-17 14:00:00',
               '1961-03-01 15:23:00', '1961-06-01 14:20:00',
               ...
               '2023-05-23 08:16:00', '2023-05-30 08:25:00',
               '2023-06-06 07:54:00', '2023-06-20 10:46:00',
               '2023-06-27 08:13:00', '2023-07-11 08:08:00',
               '2023-07-25 08:02:00', '2023-08-24 08:20:00',
               '2023-08-29 09:18:00', '2023-09-05 09:10:00'],
              dtype='datetime64[ns]', name='Date_HR', length=2323, freq=None)
```

to get the 0 row which is the start time you use your brackets

In [18]:

```
df_mlo.index[0]
```

Out[18]:

```
Timestamp('1960-03-30 12:00:00')
```

to get the elapsed you subtract the index from the start time. This works but gives funny results so THIS WONT WORK YET!

In [19]:

```
df_mlo['elapsed'] = (df_mlo.index.to_series() - df_mlo.index.to_series()[0])
```

In [20]:

```
df_mlo.head()
```

Out[20]:

	Flag	CO2	elapsed
Date_HR			
1960-03-30 12:00:00	0	318.27	0 days 00:00:00
1960-03-31 16:00:00	0	318.29	1 days 04:00:00

	Flag	CO2	elapsed
Date_HR			
1960-04-24 08:00:00	0	319.33	24 days 20:00:00
1960-08-18 10:00:00	0	315.56	140 days 22:00:00
1960-10-06 14:48:00	0	312.62	190 days 02:48:00

In [21]:

```
df_mlo.elapsed.dtype
```

Out[21]:

```
dtype('<m8[ns]')
```

So we have no idea what that elapsed time type means and we can't do math on it. So we google stackoverflow and ask for help. This is not the best post but it helps. http://stackoverflow.com/questions/24588437/convert-date-to-float-for-linear-regression-on-pandas-data-frame If you divide a datetime by a timedelta it converts it for you. It is wonky. And if we divide by 365 days it will put it into years.

In [22]:

```
df_mlo['elapsed'] = (df_mlo.index.to_series() - df_mlo.index.to_series()[0])\
                    /datetime.timedelta(days=365)
```

In [23]:

```
df_mlo.head()
```

Out[23]:

	Flag	CO2	elapsed
Date_HR			
1960-03-30 12:00:00	0	318.27	0.000000
1960-03-31 16:00:00	0	318.29	0.003196
1960-04-24 08:00:00	0	319.33	0.068037
1960-08-18 10:00:00	0	315.56	0.386073
1960-10-06 14:48:00	0	312.62	0.520868

so we have an elapsed time. so now lets compute an S at every time step.

Set your S_o to 3 ppm. Set R to 1% which is 1/100 which is .01

then plot the S over time.

Flashback

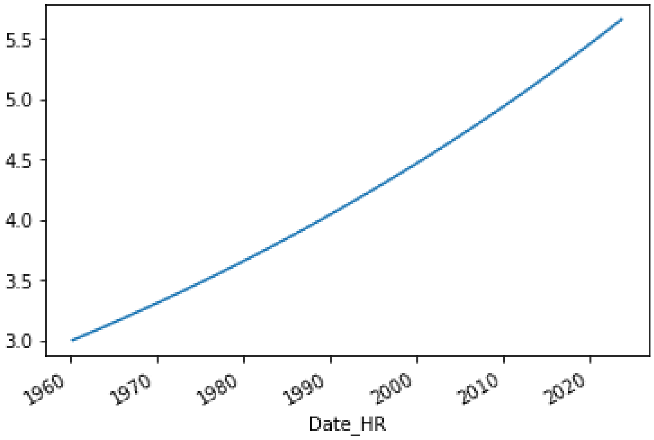
Make sure to put in decimal places also exponent is np.exp

In [24]:

```
So=3.
R=.01
df_mlo['S']=So*np.exp(R*df_mlo.elapsed)
df_mlo.S.plot()
```

Out[24]:

```
<AxesSubplot:xlabel='Date_HR'>
```



Now here is the solution for the whole CO2 equation.
http://www.atmosedu.com/physlets/GlobalPollution/solution.jpg

$$C = C_o e^{-t/\tau} + \frac{S_o}{R+\frac{1}{\tau}}\left[e^{Rt} - e^{-t/\tau}\right]$$

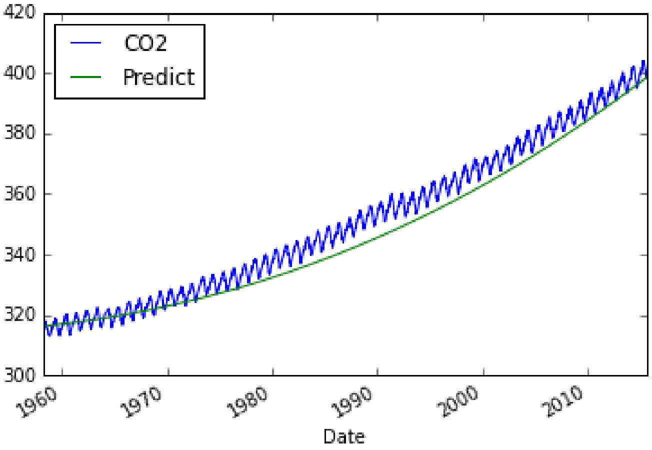
Solve the equation and figure out the best fit parameters for CO2. How much has CO2 output been growing over the years?
set tau to 120.0. Remember this is a simple box model so real life might be slightly different.

What would CO_2 look like now if the growth rate R was halved starting at the beginning of the dataset?

here are my answers. Good luck

```
In [36]:
```

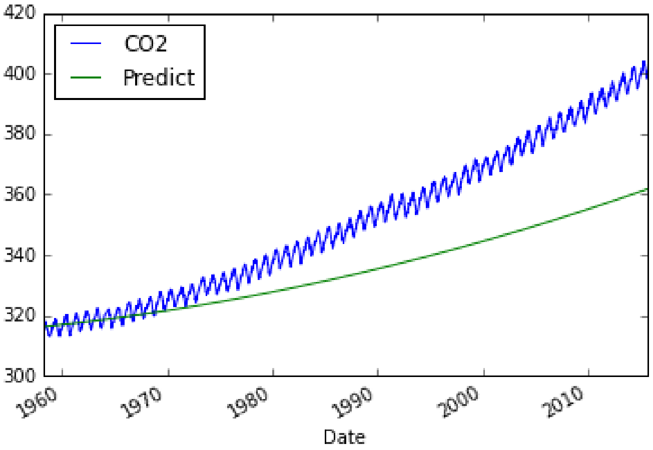
```
Out[36]: <matplotlib.axes._subplots.AxesSubplot at 0x1fb12748>
```



My answer with the R halved.

```
In [37]:
```

```
Out[37]: <matplotlib.axes._subplots.AxesSubplot at 0x1fd2e240>
```


CO2
Predict
420
400
380
360
340
320
300
1960
1970
1980
1990
2000
2010
Date

```
In [ ]:
```