**Idea**

The main ideas behind the project can be summarized as follows:

1. The website that has to be converted to Indian currency has to be first downloaded to the local machine with the help of the urllib2 package that contains the function urlopen to send an http request to the desired website.
2. The http response from the web server can be captured by using the read function on the object returned by the urlopen function.
3. We also query the Google finance server to get the current currency conversion rate for USD to INR conversion.
4. To get the conversion rate regular expressions are used to parse the span in which the result of the conversion is present when input=1. (appropriate headers are set at this stage to query the server like from=USD and to=INR and a=1)
5. In order to replace all the currency literals and symbols used inside the document we have to match the following patterns

*$50 million, USD 50 million, 50 million dollar*

To get these formats we again use regular expressions like *r'USD\s([\d.,]+\s\*?\w+)' and r'\$([\d.,]+\s\*?\w+)'*

Combine the both formats to get: **r'(\$|USD\s)([\d.,]+\s\*?\w+)’**

1. Then we use the re package to call the sub function to replace the matched substring in the website string to get the converted text
2. The conversion function works on the following principle:
   1. Remove the undesirable characters like $, USD and ‘,’ from the currency literal.
   2. Construct an absolute value from the word form like ‘million’ ‘billion’ etc. by using the dictionary that maps these words to numbers.
   3. Convert the absolute value to Indian INR by multiplying with the rate.
   4. Now perform successive division till we are left with a number<1000 so as to convert the absolute number to words (like crore lakh and thousand).
   5. This is done by reverse iterating in the order ->crore->lakh->thousand
   6. Any other characters that may have matched are appended after the converted string.
   7. Append INR at the starting of the converted string and then the remaining value after all the iterations.
   8. Return the final converted string.
3. The converted string is then written on to a file named ”page\_content.html” by opening the file in write mode and writing to the file using the write function on the file object.