In [4]:

```
# Here are all the installs and imports you will need for your word cloud script and uploa

!pip install wordcloud
!pip install fileupload
!pip install ipywidgets
!jupyter nbextension install --py --user fileupload
!jupyter nbextension enable --py fileupload

import wordcloud
import numpy as np
from matplotlib import pyplot as plt
from IPython.display import display
import fileupload
import io
import sys
```

```
Requirement already satisfied: webencodings in /opt/conda/lib/python3.6/si
te-packages (from bleach->nbconvert->notebook>=4.4.1->widgetsnbextension~=
3.4.0->ipywidgets) (0.5.1)
Installing /opt/conda/lib/python3.6/site-packages/fileupload/static -> fil
eupload
Up to date: /home/jovyan/.local/share/jupyter/nbextensions/fileupload/exte
nsion.js
Up to date: /home/jovyan/.local/share/jupyter/nbextensions/fileupload/widg
et.js
Up to date: /home/jovyan/.local/share/jupyter/nbextensions/fileupload/file
upload/widget.js
- Validating: OK

    To initialize this nbextension in the browser every time the notebook
 (or other app) loads:

          jupyter nbextension enable fileupload --user --py

Enabling notebook extension fileupload/extension...
      - Validating: OK
```

Whew! That was a lot. All of the installs and imports for your word cloud script and uploader widget have been completed.

IMPORTANT! If this was your first time running the above cell containing the installs and imports, you will need save this notebook now. Then under the File menu above, select Close and Halt. When the notebook has completely shut down, reopen it. This is the only way the necessary changes will take affect.

To upload your text file, run the following cell that contains all the code for a custom uploader widget. Once you run this cell, a "Browse" button should appear below it. Click this button and navigate the window to locate your saved text file.

In [6]:

```
# This is the uploader widget

def _upload():

    _upload_widget = fileupload.FileUploadWidget()

    def _cb(change):
        global file_contents
        decoded = io.StringIO(change['owner'].data.decode('utf-8'))
        filename = change['owner'].filename
        print('Uploaded `{}` ({:.2f} kB)'.format(
            filename, len(decoded.read()) / 2 **10))
        file_contents = decoded.getvalue()

    _upload_widget.observe(_cb, names='data')
    display(_upload_widget)

_upload()
```

```
FileUploadWidget(label='Browse', _dom_classes=('widget_item', 'btn-group'))
```

```
Uploaded `ex.txt` (5.28 kB)
```

The uploader widget saved the contents of your uploaded file into a string object named file_contents that your word cloud script can process. This was a lot of preliminary work, but you are now ready to begin your script.

Write a function in the cell below that iterates through the words in file_contents, removes punctuation, and counts the frequency of each word. Oh, and be sure to make it ignore word case, words that do not contain all alphabets and boring words like "and" or "the". Then use it in the generate_from_frequencies function to generate your very own word cloud!

Hint: Try storing the results of your iteration in a dictionary before passing them into wordcloud via the generate_from_frequencies function.

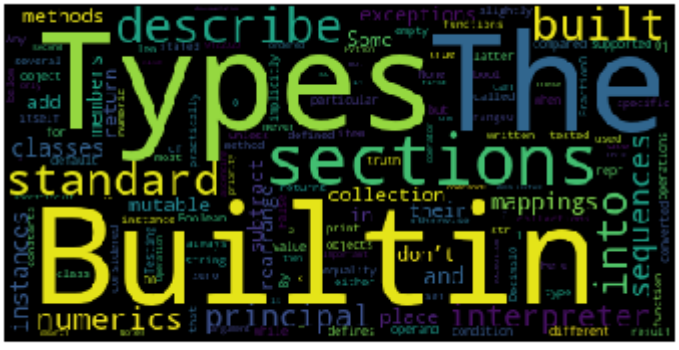
In [7]:

```
def calculate_frequencies(file_contents):
    # Here is a list of punctuations and uninteresting words you can use to process your te
    punctuations = '''!()-[]{};:'"\,<>./?@#$%^&*_~'''
    uninteresting_words = ["the", "a", "to", "if", "is", "in" "it", "of", "and", "or","on",
    "we", "our", "ours", "you", "your", "yours", "he", "she", "him", "his", "her", "hers",
    "their", "what", "which", "who", "whom", "this", "that", "am", "are", "was", "were", "b
    "have", "has", "had", "do", "does", "did", "but", "at", "by", "with", "from", "here", "
    "all", "any", "both", "each", "few", "more", "some", "such", "no", "nor", "too", "very"

    # LEARNER CODE START HERE
    frequencies =  {}
    taken = []
    for letter in punctuations:
        file_contents =  file_contents.replace(letter,'')
    for word in uninteresting_words:
        w = ' '+word+' '
        file_contents = file_contents.replace(w,' ')
    for word in file_contents.split():
        if word.lower() not in taken:
            taken.append(word.lower())
            if word not in frequencies:
                frequencies[word] = 1
            else:
                frequencies[word] += 1


    #wordcloud
    cloud = wordcloud.WordCloud()
    cloud.generate_from_frequencies(frequencies)
    return cloud.to_array()
```

If you have done everything correctly, your word cloud image should appear after running the cell below. Fingers crossed!

In [8]:

```
# Display your wordcloud image

myimage = calculate_frequencies(file_contents)
plt.imshow(myimage, interpolation = 'nearest')
plt.axis('off')
plt.show()
```



In [ ]: