Please create a `hw3` directory inside the private GitHub repo where you have submitted your previous homework (`<github-username>-hw`). All your code should appear in a file called `hw3.py` that lives inside the `hw3` directory. Make sure to add `hw3.py` to the repo and commit your changes with `$ git commit -a -m "replace this with your own log message"`. If you are working from a previously cloned repo, remember to execute `$ git pull` to retrieve any changes from `github.com` *before* committing.

Question 1. *We have provided a function called `create_index(*filenames)`. It takes a variable number of filenames as input and returns a dictionary that maps a filename (str) to the unique words the file contains (set of str). In this question, we want you to create an* inverted index*—an inverted index maps content to the location of the content.*

Please write the function `invert_index(d)` that takes a dictionary d as input and returns a new dictionary d_{inv}, where:

- *d contains mappings from a filename (str) to the words contained by the file (set of str), and*
- *d_{inv} contains mappings from a word (str) to the filenames that contain the word (set of str).*

We have provided the function `create_index(d)` below, as well as the content of the text files that were used in this example (`sample1.txt` and `sample2.txt`). You can also find these resources on the course website.

```
1  def create_index(*filenames):
2      """
3      takes a variable number of filenames, and returns a dictionary
4      that maps each filename to the set of all words in the file
5      """
6
7      def read_words(filename):
8          """returns all of the words in filename as a list"""
9          with open(filename, 'rt') as fin:
10             return fin.read().split()
11
12     return {fname : set(read_words(fname)) for fname in filenames}
```

sample1.txt contents:

```
this file contains some words
```

sample2.txt contents:

```
this file contains some other words too
```

Example usage:

```
>>> from hw3 import create_index, invert_index
>>> d = create_index('sample1.txt', 'sample2.txt')
>>> d
{'sample1.txt': {'contains', 'this', 'file', 'some', 'words'},
'sample2.txt': {'this', 'contains', 'too', 'words', 'file', 'other', 'some'}}
>>> d_inv = invert_index(d)
>>> d_inv
{'this': ['sample1.txt', 'sample2.txt'], 'file': ['sample1.txt', 'sample2.txt'],
'other': ['sample2.txt'], 'contains': ['sample1.txt', 'sample2.txt'],
'too': ['sample2.txt'], 'some': ['sample1.txt', 'sample2.txt'],
'words': ['sample1.txt', 'sample2.txt']}
```