```python
# Syed Abidi, DS-530, Chapter-1, Exercise 1-2
# This program performs the solution for Chapter -1, Exercise 1.2 on page no.
111
# a): Create a file name "chap01.ex.py", and write a code that reads file
"2002FemResp.dat.gz"
# b): Print the value counts for a variable "Pregnum" and compare them to the
published results in "NSFG codebook"


from __future__ import print_function, division

import sys

import nsfg

import thinkstats2

# Create a file name "chap01.ex.py", and write a code that reads file
"2002FemResp.dat.gz"
def ReadFemResp(dct_file='2002FemResp.dct', dat_file='2002FemResp.dat.gz',
nrows=None):
    # Reads the NSFG Respondent file and defines the string file dct
    dct = thinkstats2.ReadStataDct(dct_file)
    # Define the data frame "df" variable to read
dat_file='2002FemResp.dat.gz'
    df = dct.ReadFixedWidth(dat_file, compression='gzip', nrows=nrows)
    # Call the function CleanFemResp to recode variables from the respondent
frame
    CleanFemResp(df)
    return df
    # retruns the DataFrame


# Define the function CleanFemResp to recode the variables from the
respondent frame on the dataFrame df
def CleanFemResp(df):
    pass
    # used as a placeholder for future code



# Define a function to validate pregnum in the respondent file.
def ValidatePregnum(resp):
    # Read the pregnancy frame
    preg = nsfg.ReadFemPreg()

    # make the map from caseid to list of pregnancy indices
    preg_map = nsfg.MakePregMap(preg)

    # iterate through the respondent pregnum series
    for index, pregnum in resp.pregnum.items():
        caseid = resp.caseid[index]
        indices = preg_map[caseid]

        # Check that pregnum from the respondent file equals the number of
records in the pregnancy file
        if len(indices) != pregnum:
```

```python
            print(caseid, len(indices), pregnum)
            return False

    return True


# Tests the functions in this module.
def main(script):

    # Read and validate the respondent file
    resp = ReadFemResp()

    assert(len(resp) == 7643)
    assert(resp.pregnum.value_counts()[1] == 1267)
    assert(ValidatePregnum(resp))

    print('%s: All tests passed.' % script)

if __name__ == '__main__':
    main(*sys.argv)
```