**"""**

**Your task is to check the "productionStartYear" of the DBPedia autos datafile for valid values.**

**The following things should be done:**

**- check if the field "productionStartYear" contains a year**

**- check if the year is in range 1886-2014**

**- convert the value of the field to be just a year (not full datetime)**

**- the rest of the fields and values should stay the same**

**- if the value of the field is a valid year in the range as described above,**

**write that line to the output\_good file**

**- if the value of the field is not a valid year as described above,**

**write that line to the output\_bad file**

**- discard rows (neither write to good nor bad) if the URI is not from dbpedia.org**

**- you should use the provided way of reading and writing data (DictReader and DictWriter)**

**They will take care of dealing with the header.**

**You can write helper functions for checking the data and writing the files, but we will call only the**

**'process\_file' with 3 arguments (inputfile, output\_good, output\_bad).**

**"""**

**import csv**

**import pprint**

**import re**

**import datetime**

**INPUT\_FILE = 'autos.csv'**

**OUTPUT\_GOOD = 'autos-valid.csv'**

**OUTPUT\_BAD = 'FIXME-autos.csv'**

**def process\_file(input\_file, output\_good, output\_bad):**

**with open(input\_file, "r") as f:**

**reader = csv.DictReader(f)**

**header = reader.fieldnames**

**p = re.compile('\d\d\d\d')**

**p1 = re.compile('dbpedia')**

**f1 = open(output\_good, "w")**

**f2 = open(output\_bad, "w")**

**i = 0**

**j = 0**

**for row in reader:**

**m1 = p1.search(row["URI"])**

**if m1:**

**m = p.match(row["productionStartYear"])**

**if m:**

**a,b = m.span()**

**year = row["productionStartYear"][a:b]**

**year = int(year)**

**row["productionStartYear"] = year**

**if (year>1886) and (year<2014):**

**f1w = csv.DictWriter(f1, delimiter=",", fieldnames = header)**

**f1w.writeheader()**

**f1w.writerow(row)**

**else:**

**f2w = csv.DictWriter(f2, delimiter="," ,fieldnames = header)**

**f2w.writeheader()**

**f2w.writerow(row)**

**else:**

**f2w = csv.DictWriter(f2, delimiter="," ,fieldnames = header)**

**f2w.writeheader()**

**f2w.writerow(row)**

**f1.close()**

**f2.close()**

**#COMPLETE THIS FUNCTION**

**# This is just an example on how you can use csv.DictWriter**

**# Remember that you have to output 2 files**

**# with open(output\_good, "w") as g:**

**# writer = csv.DictWriter(g, delimiter=",", fieldnames= header)**

**# writer.writeheader()**

**# for row in YOURDATA:**

**# writer.writerow(row)**

**def test():**

**process\_file(INPUT\_FILE, OUTPUT\_GOOD, OUTPUT\_BAD)**

**if \_\_name\_\_ == "\_\_main\_\_":**

**test()**