

```

from zipfile import ZipFile

from urllib.request import urlopen

from io import BytesIO

import pandas as pd

#Read in 2000 geoheaders file from Census FTP

url = r'https://www2.census.gov/census_2000/datasets/
redistricting_file--pl_94-171/District_of_Columbia/dcgeo.upl.zip'

#The file is fixed width, we're only after SUMLEV, LOGRECNO, and
GEOID00

df =
pd.read_fwf(ZipFile(BytesIO(urlopen(url).read())).open('dcgeo.upl'),
            widths = [8,3,7,7,30,11], header=None, dtype=str)

#Cleanup columns names and filter to SUMLEV 750 (blocks)

df.columns = ['junk','SUMLEV','junk','LOGRECNO','junk','GEOID00']

geoheaders = df[df['SUMLEV']=='750'][['LOGRECNO','GEOID00']]

#Read in 2000 PL Segment 1 file

url = r'https://www2.census.gov/census_2000/datasets/
redistricting_file--pl_94-171/District_of_Columbia/dc00001.upl.zip'

#Read in CSV file

df =
pd.read_csv(ZipFile(BytesIO(urlopen(url).read())).open('dc00001.upl'),
            low_memory=False, dtype=str, header=None)

#Columns names for data dictionary https://www.census.gov/prod/
cen2000/doc/pl-00-1.pdf

df.columns = ["FILEID", "STUSAB", "CHARITER", "CIFSN", "LOGRECNO"] +
(",".join("P00" + str(i) for i in range(10001,10072)) + ',' +
",".join("P00" + str(i) for i in range(20001,20074))).split(",")

#Merge onto filtered geoheaders and filter to needed columns, add a

```

YEAR column

```
df2000 =df.merge(geoheaders, on = 'LOGRECNO')
[['GEOID00', 'P0020001', 'P0020002', 'P0020003', 'P0020004', 'P0020005', 'P0020006', 'P0020007', 'P0020008', 'P0020009', 'P0020010', 'P0020011']]
```

```
df2000['YEAR'] = 2000
```

#Read in 2010 geoheaders file from Census FTP

```
url = r'https://www2.census.gov/census_2010/01-Redistricting_File--PL_94-171/District_of_Columbia/dc2010.pl.zip'
```

#The file is fixed width, we're only after SUMLEV, LOGRECNO, and GEOID00

```
df =
pd.read_fwf(ZipFile(BytesIO(urlopen(url).read())).open('dcgeo2010.pl'),
```

```
widths = [8,3,7,7,30,11], header=None, dtype=str)
```

#Cleanup columns names and filter to SUMLEV 750 (blocks)

```
df.columns = ['junk', 'SUMLEV', 'junk', 'LOGRECNO', 'junk', 'GEOID10']
```

```
geoheaders = df[df['SUMLEV']=='750'][['LOGRECNO', 'GEOID10']]
```

#Read in 2010 PL Segment 1 file from earlier zip file

```
df =
pd.read_csv(ZipFile(BytesIO(urlopen(url).read())).open('dc000012010.pl'),
```

```
low_memory=False, dtype=str, header=None)
```

#Column names from <https://www.census.gov/prod/cen2010/doc/sf1.pdf>

```
df.columns = ["FILEID", "STUSAB", "CHARITER", "CIFSN", "LOGRECNO"] +
(",".join("P00" + str(i) for i in range(10001,10072)) + ',' +
",".join("P00" + str(i) for i in range(20001,20074))).split(",")
```

#Merge onto filtered geoheaders and filter to needed columns, add a YEAR column

```
df2010 =df.merge(geoheaders, on = 'LOGRECNO')
[['GEOID10', 'P0020001', 'P0020002', 'P0020003', 'P0020004', 'P0020005', 'P0
020006', 'P0020007', 'P0020008', 'P0020009', 'P0020010', 'P0020011']]
```

```
df2010['YEAR'] = 2010
```

```
#Read in 2020 geoheaders
```

```
url = r'https://www2.census.gov/programs-surveys/decennial/2020/data/
01-Redistricting_File--PL_94-171/District_of_Columbia/dc2020.pl.zip'
```

```
#Pipe separated file
```

```
df =
pd.read_csv(ZipFile(BytesIO(urlopen(url).read())).open('dcgeo2020.pl')
,
```

```
sep = '|',low_memory=False, dtype=str, header=None)
```

```
#Column names form https://www2.census.gov/programs-surveys/decennial/
2020/technical-documentation/complete-tech-docs/summary-file/
2020Census_PL94_171Redistricting_StatesTechDoc_English.pdf
```

```
df.columns= ["FILEID", "STUSAB", "SUMLEV", "GEOVAR", "GEOCOMP",
"CHARITER", "CIFSN", "LOGRECNO", "GEOID20", "GEOCODE", "REGION",
"DIVISION", "STATE", "STATENS", "COUNTY", "COUNTYCC", "COUNTYNS",
"COUSUB", "COUSUBCC", "COUSUBNS", "SUBMCD", "SUBMCDCC", "SUBMCDNS",
"ESTATE", "ESTATECC", "ESTATENS", "CONCIT", "CONCITCC", "CONCITNS",
"PLACE", "PLACECC", "PLACENS", "TRACTCE20", "BLKGRP", "BLOCKCE20",
"AIANHH", "AIHHTLI", "AIANHHFP", "AIANHHCC", "AIANHHNS", "AITS",
"AITSFP", "AITSCC", "AITSNS", "TTRACT", "TBLKGRP", "ANRC", "ANRCCC",
"ANRCNS", "CBSA", "MEMI", "CSA", "METDIV", "NECTA", "NMEMI", "CNECTA",
"NECTADIV", "CBSAPCI", "NECTAPCI", "UA", "UATYPE", "UR", "CD116",
"CD118", "CD119", "CD120", "CD121", "SLDU18", "SLDU22", "SLDU24",
"SLDU26", "SLDU28", "SLDL18", "SLDL22", "SLDL24", "SLDL26", "SLDL28",
"VTD", "VTDI", "ZCTA", "SDELM", "SDSEC", "SDUNI", "PUMA", "AREALAND",
"AREAWATR", "BASENAME", "NAME", "FUNCSTAT", "GCUNI", "POP100",
"HU100", "INTPTLAT", "INTPTLON", "LSADC", "PARTFLAG", "UGA"]
```

```
#Filter geoheaders
```

```
geoheaders = df[df['SUMLEV']=='750'][['LOGRECNO', 'GEOID20']]
```

```
#Read inSF1 file from earlier zip file
```

```

df =
pd.read_csv(ZipFile(BytesIO(urlopen(url).read())).open('dc000012020.pl
'),

                sep = '|', low_memory=False, header=None, dtype=str)

#Column names form https://www2.census.gov/programs-surveys/decennial/
2020/technical-documentation/complete-tech-docs/summary-file/
2020Census_PL94_171Redistricting_StatesTechDoc_English.pdf

df.columns = ["FILEID", "STUSAB", "CHARITER", "CIFSN", "LOGRECNO"] +
(",".join("P00" + str(i) for i in range(10001,10072)) + ',' +
",".join("P00" + str(i) for i in range(20001,20074))).split(",")

#Merge onto filtered geoheaders and filter to needed columns, add a
YEAR column

df2020 =df.merge(geoheaders, on = 'LOGRECNO')
[['GE0ID20', 'P0020001', 'P0020002', 'P0020003', 'P0020004', 'P0020005', 'P0
020006', 'P0020007', 'P0020008', 'P0020009', 'P0020010', 'P0020011']]

df2020['YEAR'] = 2020

#Filter by GEOIDs to 4 census blocks.

#2000 map: https://opendata.dc.gov/datasets/census-blocks-2000/

#2010 map: https://opendata.dc.gov/datasets/census-blocks-2010/

#The 2020 map isn't available from opendata.dc.gov, but using the
tiger/line shape files you can get the new census blocks

#2020 Tiger/Line: https://www2.census.gov/geo/tiger/TIGER2020/
TABBLOCK20/tl_2020_11_tabblock20.zip

df2000 =
df2000[df2000['GE0ID00'].isin(['00320011000', '00320011001', '0032001100
2', '00320011003'])]

df2010 =
df2010[df2010['GE0ID10'].isin(['0320011004', '0320011001', '0320011002',
'0320011003'])]

df2020 =
df2020[df2020['GE0ID20'].isin(['7500000US110010032001000', '7500000US11
0010032001001', '7500000US110010032001002', '7500000US110010032001003'])
]

```

```

#Convert data columns to interger at last step to ensure
df = df2000.append(df2010).append(df2020)

cols = (",".join("P00" + str(i) for i in
range(20001,20012))).split(',')

df[cols] = df[cols].apply(pd.to_numeric, errors='coerce')

#Group by year
df = df.groupby('YEAR')[cols].sum().reset_index()

df = df.rename(columns = {'P0020001': 'Total Population',
                          'P0020002': 'Hispanic or Latino',
                          'P0020003': 'Non Hispanic',
                          'P0020004': 'Non Hispanic, One Race',
                          'P0020005': 'Non Hispanic, One Race, White',
                          'P0020006': 'Non Hispanic, One Race, Black
of African American',
                          'P0020007': 'Non Hispanic, One Race,
American Indian or Alaska Native',
                          'P0020008': 'Non Hispanic, One Race, Asian',
                          'P0020009': 'Non Hispanic, One Race, Native
Hawaiian or Other Pacific Islander',
                          'P0020010': 'Non Hispanic, One Race, Some
Other Race',
                          'P0020011': 'Non Hispanic, One Race, Two or
More Races'})

df.to_csv('ParkViewRace.csv')

```