```
import os
import sys
from tempfile import NamedTemporaryFile
from urllib.request import urlopen
from urllib.parse import unquote, urlparse
from urllib.error import HTTPError
from zipfile import ZipFile
import tarfile
import shutil

CHUNK_SIZE = 40960
DATA_SOURCE_MAPPING = 'flight-delay-dataset-20182022:https%3A%2F%2Fstorage.googleapis.com%2Fkaggle-data-sets%2F2529204%2F4295427%2Fbundle%2Farchive.zip%3FX-Goog-Algorithm

KAGGLE_INPUT_PATH='/kaggle/input'
KAGGLE_WORKING_PATH='/kaggle/working'
KAGGLE_SYMLINK='kaggle'

!umount /kaggle/input/ 2> /dev/null
shutil.rmtree('/kaggle/input', ignore_errors=True)
os.makedirs(KAGGLE_INPUT_PATH, 0o777, exist_ok=True)
os.makedirs(KAGGLE_WORKING_PATH, 0o777, exist_ok=True)

try:
  os.symlink(KAGGLE_INPUT_PATH, os.path.join("..", 'input'), target_is_directory=True)
except FileExistsError:
  pass
try:
  os.symlink(KAGGLE_WORKING_PATH, os.path.join("..", 'working'), target_is_directory=True)
except FileExistsError:
  pass

for data_source_mapping in DATA_SOURCE_MAPPING.split(','):
    directory, download_url_encoded = data_source_mapping.split(':')
    download_url = unquote(download_url_encoded)
    filename = urlparse(download_url).path
    destination_path = os.path.join(KAGGLE_INPUT_PATH, directory)
    try:
        with urlopen(download_url) as fileres, NamedTemporaryFile() as tfile:
            total_length = fileres.headers['content-length']
            print(f'Downloading {directory}, {total_length} bytes compressed')
            dl = 0
            data = fileres.read(CHUNK_SIZE)
            while len(data) > 0:
                dl += len(data)
                tfile.write(data)
                done = int(50 * dl / int(total_length))
                sys.stdout.write(f"\r[{'=' * done}{' ' * (50-done)}] {dl} bytes downloaded")
                sys.stdout.flush()
                data = fileres.read(CHUNK_SIZE)
            if filename.endswith('.zip'):
              with ZipFile(tfile) as zfile:
                zfile.extractall(destination_path)
            else:
              with tarfile.open(tfile.name) as tarfile:
                tarfile.extractall(destination_path)
            print(f'\nDownloaded and uncompressed: {directory}')
    except HTTPError as e:
        print(f'Failed to load (likely expired) {download_url} to path {destination_path}')
        continue
    except OSError as e:
        print(f'Failed to load {download_url} to path {destination_path}')
        continue

print('Data source import complete.')
```

```
Downloading flight-delay-dataset-20182022, 4006061203 bytes compressed
[==================================================] 4006061203 bytes downloaded
Downloaded and uncompressed: flight-delay-dataset-20182022
Data source import complete.
```

Air Flight Dataset

This dataset encompasses comprehensive flight details, covering cancellations and delays across various airlines, dating back to January 2022.

For streamlined access, you're encouraged to utilize either the Combined_Flights_XXXX.csv or Combined_Flights_XXXX.parquet files, which consolidate data for the entire year. These files have also undergone column filtering, removing those primarily populated with null values from the original dataset.

Should you require access to the raw, month-wise data inclusive of all columns, you can locate it within files labeled Flights_XXXX_X.csv.

Load dependencies packages

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Import the dataset

```
df = pd.read_csv("/kaggle/input/flight-delay-dataset-20182022/Combined_Flights_2022.csv")
```

```
df.head()
```

	FlightDate	Airline	Origin	Dest	Cancelled	Diverted	CRSDepTime	DepTime	DepDelayMinutes	DepDelay
0	2022-04-04	Commutair Aka Champlain Enterprises, Inc.	GJT	DEN	False	False	1133	1123.0	0.0	-10.0
1	2022-04-04	Commutair Aka Champlain Enterprises, Inc.	HRL	IAH	False	False	732	728.0	0.0	-4.0
2	2022-04-04	Commutair Aka Champlain Enterprises, Inc.	DRO	DEN	False	False	1529	1514.0	0.0	-15.0
3	2022-04-04	Commutair Aka Champlain Enterprises, Inc.	IAH	GPT	False	False	1435	1430.0	0.0	-5.0
4	2022-04-04	Commutair Aka Champlain Enterprises, Inc.	DRO	DEN	False	False	1135	1135.0	0.0	0.0

5 rows × 61 columns

Check the columns of dataframe

```
df.columns
```

```
Index(['FlightDate', 'Airline', 'Origin', 'Dest', 'Cancelled', 'Diverted',
       'CRSDepTime', 'DepTime', 'DepDelayMinutes', 'DepDelay', 'ArrTime',
       'ArrDelayMinutes', 'AirTime', 'CRSElapsedTime', 'ActualElapsedTime',
       'Distance', 'Year', 'Quarter', 'Month', 'DayofMonth', 'DayOfWeek',
       'Marketing_Airline_Network', 'Operated_or_Branded_Code_Share_Partners',
       'DOT_ID_Marketing_Airline', 'IATA_Code_Marketing_Airline',
       'Flight_Number_Marketing_Airline', 'Operating_Airline',
       'DOT_ID_Operating_Airline', 'IATA_Code_Operating_Airline',
       'Tail_Number', 'Flight_Number_Operating_Airline', 'OriginAirportID',
       'OriginAirportSeqID', 'OriginCityMarketID', 'OriginCityName',
       'OriginState', 'OriginStateFips', 'OriginStateName', 'OriginWac',
       'DestAirportID', 'DestAirportSeqID', 'DestCityMarketID', 'DestCityName',
       'DestState', 'DestStateFips', 'DestStateName', 'DestWac', 'DepDel15',
       'DepartureDelayGroups', 'DepTimeBlk', 'TaxiOut', 'WheelsOff',
       'WheelsOn', 'TaxiIn', 'CRSArrTime', 'ArrDelay', 'ArrDel15',
       'ArrivalDelayGroups', 'ArrTimeBlk', 'DistanceGroup',
       'DivAirportLandings'],
      dtype='object')
```

```
df.info()
```

```
 5   Diverted                                bool
 6   CRSDepTime                              int64
 7   DepTime                                 float64
 8   DepDelayMinutes                         float64
 9   DepDelay                                float64
 10  ArrTime                                 float64
 11  ArrDelayMinutes                         float64
 12  AirTime                                 float64
 13  CRSElapsedTime                          float64
 14  ActualElapsedTime                       float64
 15  Distance                                float64
 16  Year                                    int64
 17  Quarter                                 int64
 18  Month                                   int64
 19  DayofMonth                              int64
 20  DayOfWeek                               int64
 21  Marketing_Airline_Network               object
 22  Operated_or_Branded_Code_Share_Partners object
 23  DOT_ID_Marketing_Airline                int64
 24  IATA_Code_Marketing_Airline             object
 25  Flight_Number_Marketing_Airline         int64
 26  Operating_Airline                       object
 27  DOT_ID_Operating_Airline                int64
 28  IATA_Code_Operating_Airline             object
 29  Tail_Number                             object
 30  Flight_Number_Operating_Airline         int64
 31  OriginAirportID                         int64
 32  OriginAirportSeqID                      int64
 33  OriginCityMarketID                      int64
 34  OriginCityName                          object
 35  OriginState                             object
 36  OriginStateFips                         int64
 37  OriginStateName                         object
 38  OriginWac                               int64
 39  DestAirportID                           int64
 40  DestAirportSeqID                        int64
 51  WheelsOff                               float64
 52  WheelsOn                                float64
 53  TaxiIn                                  float64
 54  CRSArrTime                              int64
 55  ArrDelay                                float64
 56  ArrDel15                                float64
 57  ArrivalDelayGroups                      float64
 58  ArrTimeBlk                              object
 59  DistanceGroup                           int64
 60  DivAirportLandings                      int64
dtypes: bool(2), float64(18), int64(23), object(18)
memory usage: 1.8+ GB
```

About the dataset

As depicted above, this dataset comprises over 4 million records and encompasses 64 variables or features. It has been meticulously recorded, with each column assigned appropriate data types.

```
df.describe()
```

	CRSDepTime	DepTime	DepDelayMinutes	DepDelay	ArrTime	ArrDelayMinutes	AirTime
count	4.078318e+06	3.957885e+06	3.957823e+06	3.957823e+06	3.954079e+06	3.944916e+06	3.944916e+06
mean	1.329587e+03	1.334374e+03	1.601494e+01	1.309049e+01	1.457886e+03	1.578307e+01	1.110075e+02
std	4.904801e+02	5.056219e+02	5.231498e+01	5.332016e+01	5.431841e+02	5.198424e+01	6.996246e+01
min	1.000000e+00	1.000000e+00	0.000000e+00	-7.800000e+01	1.000000e+00	0.000000e+00	8.000000e+00
25%	9.140000e+02	9.170000e+02	0.000000e+00	-5.000000e+00	1.046000e+03	0.000000e+00	6.000000e+01
50%	1.320000e+03	1.325000e+03	0.000000e+00	-2.000000e+00	1.500000e+03	0.000000e+00	9.400000e+01
75%	1.735000e+03	1.744000e+03	1.100000e+01	1.100000e+01	1.914000e+03	1.000000e+01	1.410000e+02
max	2.359000e+03	2.400000e+03	7.223000e+03	7.223000e+03	2.400000e+03	7.232000e+03	7.270000e+02

8 rows × 41 columns