merge_data

March 19, 2023

[ ]:
```
# import libraries
import pandas as pd
import numpy as np
```

[ ]:
```
# carbon dioxide emissions per person per year per country
co2 = pd.read_csv ('co2.csv')
```

[ ]:
```
co2.shape
```

[ ]:
```
co2.columns
```

[ ]:
```
co2.describe()
```

[ ]:
```
co2.info()
```

[ ]:
```
co2.head()
```

[ ]:
```
co2.tail()
```

[ ]:
```
# GDP (gross domestic product) per year per country
gm = pd.read_csv('gapminder.csv')
```

[ ]:
```
gm.shape
```

[ ]:
```
gm.columns
```

[ ]:
```
gm.describe()
```

[ ]:
```
gm.info()
```

[ ]:
```
gm.head()
```

[ ]:
```
gm.tail()
```

[ ]:
```
# drop duplicates
df_gm = gm[['Country', 'region']].drop_duplicates()
```

[ ]:
```
# inner join or merge two data frames
df_w_regions = pd.merge (co2, df_gm,
                         left_on = 'country',
                         right_on = 'Country',
                         how = 'inner')
```

[ ]:
```
# drop one of the country columns
df_w_regions = df_w_regions.drop('Country', axis = 'columns')
```

[ ]:
```
# change identifier variables to our choice
new_co2 = pd.melt (df_w_regions, id_vars = ['country', 'region'])
```

[ ]:
```
# rename columns
new_cols = ['country', 'region', 'year', 'co2']
new_co2.columns = new_cols
```

[ ]:
```
new_co2.info()
```

[ ]:
```
df_co2 = new_co2[new_co2['year'] > '1963']
```

[ ]:
```
df_co2 = df_co2.sort_values (by = ['country', 'year'])
```

[ ]:
```
df_co2.head()
```

[ ]:
```
df_gdp = gm [['Country', 'Year', 'gdp']]
```

[ ]:
```
df_gdp.columns = ['country', 'year', 'gdp']
```

[ ]:
```
df_gdp.head()
```

[ ]:
```
df_gdp.info()
```

[ ]:
```
# df_gdp['year'] = df_gdp['year'].astype(str)
```

[ ]:
```
df_gdp = df_gdp.astype({'year':'str'})
```

[ ]:
```
df_gdp.info()
```

[ ]:
```
# merge the two data frames
data = pd.merge(df_co2, df_gdp, on = ['country', 'year'], how = 'left')
```

[ ]:
```
data = data.dropna()
```

[ ]:
```
data['year'] = data['year'].astype(int)
```

[ ]:
```
data.info()
```

[ ]:
```
# create numpy arrays
np_co2 = np.array(data['co2'])
np_gdp = np.array(data['gdp'])
```

[ ]:
```
# get correlation coefficient
np.corrcoef (np_co2, np_gdp)
```