Hands-on Activity 10.1 Data Analysis using Python

Name: John Rome A. Belocora

Section: CPE22S3

Transforming Waste Management: Learning from the Best to Improve the Worst

Solid waste data from 330 cities

What A Waste Global Dataset

```
import pandas as pd
import numpy as np

cd = pd.read_csv('country_level_data_0.csv')
cd
```

	iso3c	region_id	country_name	income_id	gdp	composition_food_organic_w
0	ABW	LCN	Aruba	HIC	NaN	
1	AFG	SAS	Afghanistan	LIC	2.141361e+10	
2	AGO	SSF	Angola	LMC	1.030423e+11	
3	ALB	ECS	Albania	UMC	1.347108e+10	
4	AND	ECS	Andorra	HIC	3.319880e+09	
...	...	...	...	...	...	
212	XKX	ECS	Kosovo	LMC	7.129272e+09	
213	YEM	MEA	Yemen, Rep.	LIC	1.192703e+10	
214	ZAF	SSF	South Africa	UMC	4.212087e+11	
215	ZMB	SSF	Zambia	LMC	2.703717e+10	
216	ZWE	SSF	Zimbabwe	LIC	1.481899e+10	

217 rows × 51 columns

```
ctd = pd.read_csv('city_level_data_0_0.csv')
ctd
```

	iso3c	region_id	country_name	income_id	city_name	additional_data_annual_budget
0	AFG	SAS	Afghanistan	LIC	Jalalabad	
1	AFG	SAS	Afghanistan	LIC	Kandahar	
2	AFG	SAS	Afghanistan	LIC	Mazar-E-Sharif	
3	AFG	SAS	Afghanistan	LIC	Kabul	
4	AFG	SAS	Afghanistan	LIC	HiratÂ	
...	...	...	...	...	...	
362	ZWE	SSF	Zimbabwe	LIC	HarareÂ	
363	ZWE	SSF	Zimbabwe	LIC	Gweru	
364	ZWE	SSF	Zimbabwe	LIC	Kariba	
365	ZWE	SSF	Zimbabwe	LIC	Masvingo City	
366	ZWE	SSF	Zimbabwe	LIC	SakubvaÂ	

367 rows × 113 columns

```
cd.columns
```

```
Index(['iso3c', 'region_id', 'country_name', 'income_id', 'gdp',
       'composition_food_organic_waste_percent', 'composition_glass_percent',
       'composition_metal_percent', 'composition_other_percent',
       'composition_paper_cardboard_percent', 'composition_plastic_percent',
       'composition_rubber_leather_percent', 'composition_wood_percent',
       'composition_yard_garden_green_waste_percent',
       'other_information_information_system_for_solid_waste_management',
       'other_information_national_agency_to_enforce_solid_waste_laws_and_regulations',
       'other_information_national_law_governing_solid_waste_management_in_the_country',
       'other_information_ppp_rules_and_regulations',
       'other_information_summary_of_key_solid_waste_information_made_available_to_the_public',
       'population_population_number_of_people',
       'special_waste_agricultural_waste_tons_year',
       'special_waste_construction_and_demolition_waste_tons_year',
       'special_waste_e_waste_tons_year',
       'special_waste_hazardous_waste_tons_year',
       'special_waste_industrial_waste_tons_year',
       'special_waste_medical_waste_tons_year',
       'total_msw_total_msw_generated_tons_year',
       'waste_collection_coverage_rural_percent_of_geographic_area',
       'waste_collection_coverage_rural_percent_of_households',
       'waste_collection_coverage_rural_percent_of_population',
       'waste_collection_coverage_rural_percent_of_waste',
       'waste_collection_coverage_total_percent_of_geographic_area',
       'waste_collection_coverage_total_percent_of_households',
       'waste_collection_coverage_total_percent_of_population',
       'waste_collection_coverage_total_percent_of_waste',
       'waste_collection_coverage_urban_percent_of_geographic_area',
       'waste_collection_coverage_urban_percent_of_households',
       'waste_collection_coverage_urban_percent_of_population',
       'waste_collection_coverage_urban_percent_of_waste',
       'waste_treatment_anaerobic_digestion_percent',
       'waste_treatment_compost_percent',
       'waste_treatment_controlled_landfill_percent',
       'waste_treatment_incineration_percent',
       'waste_treatment_landfill_unspecified_percent',
       'waste_treatment_open_dump_percent', 'waste_treatment_other_percent',
       'waste_treatment_recycling_percent',
       'waste_treatment_sanitary_landfill_landfill_gas_system_percent',
       'waste_treatment_unaccounted_for_percent',
       'waste_treatment_waterways_marine_percent',
       'where_where_is_this_data_measured'],
      dtype='object')
```

We will only focus on getting the basic data of Wastes

```
# Country with the Highest Organic Food Waste
food_waste = cd[['country_name','composition_food_organic_waste_percent']].sort_values(by='composition_food_organic_waste_percent', ascendir
food_waste
```

	country_name	composition_food_organic_waste_percent
61	Ethiopia	87.60
38	Cameroon	83.40
13	Burundi	81.00
17	Bangladesh	80.58
200	Uganda	74.50
...	...	...
182	Eswatini	NaN
190	Tajikistan	NaN
199	Tanzania	NaN
206	Venezuela, RB	NaN
215	Zambia	NaN

217 rows × 2 columns

Next steps: View recommended plots

```
# Country with the Highest Composition Glass
compo_glass = cd[['country_name','composition_glass_percent']].sort_values(by='composition_glass_percent', ascending=False)
compo_glass
```

	country_name	composition_glass_percent
117	Latvia	21.40
186	Turks and Caicos Islands	20.50
179	Slovak Republic	19.00
207	British Virgin Islands	18.10
164	Russian Federation	16.82
...	...	...
182	Eswatini	NaN
190	Tajikistan	NaN
199	Tanzania	NaN
206	Venezuela, RB	NaN
215	Zambia	NaN

217 rows × 2 columns

Next steps: View recommended plots

```
# Country with the Highest composition metal
compo_metal = cd[['country_name','composition_metal_percent']].sort_values(by='composition_metal_percent', ascending=False)
compo_metal
```

	country_name	composition_metal_percent
10	Australia	19.380000
169	Singapore	19.120000
214	South Africa	16.910461
66	Micronesia, Fed. Sts.	16.730000
170	Solomon Islands	13.150000
...	...	...
182	Eswatini	NaN
190	Tajikistan	NaN
199	Tanzania	NaN
206	Venezuela, RB	NaN
215	Zambia	NaN

217 rows × 2 columns

Next steps: View recommended plots

```
compo_other = cd[['country_name','composition_other_percent']].sort_values(by='composition_other_percent', ascending=False)
compo_other
```

	country_name	composition_other_percent
21	Bosnia and Herzegovina	92.70
30	Botswana	91.90
128	Mali	72.50
168	Senegal	68.00
173	San Marino	65.56
...	...	...
182	Eswatini	NaN
190	Tajikistan	NaN
199	Tanzania	NaN
206	Venezuela, RB	NaN
215	Zambia	NaN

217 rows × 2 columns

Next steps: View recommended plots

```
#Merging the country data and the city data
merged_data = pd.merge(cd, ctd, on='country_name', suffixes=('cd', 'ctd'))
merged_data
```

	iso3ccd	region_idcd	country_name	income_idcd	gdp	composition_food_org
0	AFG	SAS	Afghanistan	LIC	2.141361e+10	
1	AFG	SAS	Afghanistan	LIC	2.141361e+10	
2	AFG	SAS	Afghanistan	LIC	2.141361e+10	
3	AFG	SAS	Afghanistan	LIC	2.141361e+10	
4	AFG	SAS	Afghanistan	LIC	2.141361e+10	
...	...	...	...	...	...	
362	ZWE	SSF	Zimbabwe	LIC	1.481899e+10	
363	ZWE	SSF	Zimbabwe	LIC	1.481899e+10	
364	ZWE	SSF	Zimbabwe	LIC	1.481899e+10	
365	ZWE	SSF	Zimbabwe	LIC	1.481899e+10	
366	ZWE	SSF	Zimbabwe	LIC	1.481899e+10	

367 rows × 163 columns

```
merged_data.columns
```

```
Index(['iso3ccd', 'region_idcd', 'country_name', 'income_idcd', 'gdp',
       'composition_food_organic_waste_percentcd',
       'composition_glass_percentcd', 'composition_metal_percentcd',
       'composition_other_percentcd', 'composition_paper_cardboard_percentcd',
       ...
       'waste_treatment_compost_percentctd',
       'waste_treatment_controlled_landfill_percentctd',
       'waste_treatment_incineration_percentctd',
       'waste_treatment_landfill_unspecified_percentctd',
       'waste_treatment_open_dump_percentctd',
       'waste_treatment_other_percentctd',
       'waste_treatment_recycling_percentctd',
       'waste_treatment_sanitary_landfill_landfill_gas_system_percentctd',
       'waste_treatment_unaccounted_for_percentctd',
       'waste_treatment_waterways_marine_percentctd'],
      dtype='object', length=163)
```

```
merged_data['city_name']
```

```
0           Jalalabad
1            Kandahar
2      Mazar-E-Sharif
```

```
3              Kabul
4             HiratÂ
           ...
362          HarareÂ
363            Gweru
364           Kariba
365    Masvingo City
366         SakubvaÂ
Name: city_name, Length: 367, dtype: object
```

```
# Creating another data frame called 'basic_data' and adding only basic infos
basic_data = merged_data[['iso3ccd', 'region_idcd', 'country_name', 'city_name',
    'composition_food_organic_waste_percentcd', 'composition_glass_percentcd',
    'composition_metal_percentcd', 'composition_other_percentcd']]

basic_data
```

	iso3ccd	region_idcd	country_name	city_name	composition_food_organic_waste_perce
0	AFG	SAS	Afghanistan	Jalalabad	
1	AFG	SAS	Afghanistan	Kandahar	
2	AFG	SAS	Afghanistan	Mazar-E-Sharif	
3	AFG	SAS	Afghanistan	Kabul	
4	AFG	SAS	Afghanistan	HiratÂ	
...	...	...	...	...	
362	ZWE	SSF	Zimbabwe	HarareÂ	
363	ZWE	SSF	Zimbabwe	Gweru	
364	ZWE	SSF	Zimbabwe	Kariba	
365	ZWE	SSF	Zimbabwe	Masvingo City	
366	ZWE	SSF	Zimbabwe	SakubvaÂ	

Next steps: View recommended plots

```
basic_data.country_name.unique()
```

```
array(['Afghanistan', 'Angola', 'Albania', 'United Arab Emirates',
       'Argentina', 'Armenia', 'American Samoa', 'Australia', 'Austria',
       'Azerbaijan', 'Burundi', 'Belgium', 'Benin', 'Burkina Faso',
       'Bangladesh', 'Bulgaria', 'Bahrain', 'Bosnia and Herzegovina',
       'Belarus', 'Belize', 'Bolivia', 'Brazil', 'Bhutan', 'Botswana',
       'Canada', 'Switzerland', 'Chile', 'China', 'Côte d'Ivoire',
       'Cameroon', 'Congo, Dem. Rep.', 'Congo, Rep.', 'Colombia',
       'Comoros', 'Costa Rica', 'Cuba', 'Cyprus', 'Czech Republic',
       'Germany', 'Djibouti', 'Denmark', 'Dominican Republic', 'Algeria',
       'Ecuador', 'Egypt, Arab Rep.', 'Spain', 'Estonia', 'Ethiopia',
       'Finland', 'Fiji', 'France', 'Micronesia, Fed. Sts.', 'Gabon',
       'United Kingdom', 'Georgia', 'Ghana', 'Guinea', 'Gambia, The',
       'Equatorial Guinea', 'Greece', 'Guatemala', 'Honduras', 'Croatia',
       'Haiti', 'Hungary', 'Indonesia', 'Isle of Man', 'India', 'Ireland',
       'Iran, Islamic Rep.', 'Iraq', 'Israel', 'Italy', 'Jordan', 'Japan',
       'Kazakhstan', 'Kenya', 'Kyrgyz Republic', 'Cambodia', 'Kiribati',
       'Korea, Rep.', 'Kuwait', 'Lao PDR', 'Lebanon', 'Liberia', 'Libya',
       'Sri Lanka', 'Lithuania', 'Latvia', 'Morocco', 'Moldova',
       'Madagascar', 'Maldives', 'Mexico', 'Marshall Islands',
       'Macedonia, FYR', 'Mali', 'Myanmar', 'Montenegro', 'Mongolia',
       'Northern Mariana Islands', 'Mozambique', 'Mauritania', 'Malawi',
       'Malaysia', 'Namibia', 'Niger', 'Nigeria', 'Nicaragua',
       'Netherlands', 'Norway', 'Nepal', 'New Zealand', 'Oman',
       'Pakistan', 'Panama', 'Peru', 'Philippines', 'Palau',
       'Papua New Guinea', 'Poland', 'Portugal', 'Paraguay',
       'West Bank and Gaza', 'Qatar', 'Romania', 'Russian Federation',
       'Rwanda', 'Saudi Arabia', 'Senegal', 'Solomon Islands',
       'Sierra Leone', 'El Salvador', 'Somalia', 'Serbia', 'South Sudan',
       'Slovak Republic', 'Slovenia', 'Sweden', 'Syrian Arab Republic',
       'Togo', 'Thailand', 'Tajikistan', 'Turkmenistan', 'Timor-Leste',
       'Tonga', 'Tunisia', 'Turkey', 'Tuvalu', 'Tanzania', 'Uganda',
       'Ukraine', 'Uruguay', 'United States', 'Uzbekistan',
       'Venezuela, RB', 'Vietnam', 'Vanuatu', 'Samoa', 'Kosovo',
       'Yemen, Rep.', 'South Africa', 'Zambia', 'Zimbabwe'], dtype=object)
```

```
basic_data.region_idcd.unique()
```

```
array(['SAS', 'SSF', 'ECS', 'MEA', 'LCN', 'EAS', 'NAC'], dtype=object)
```

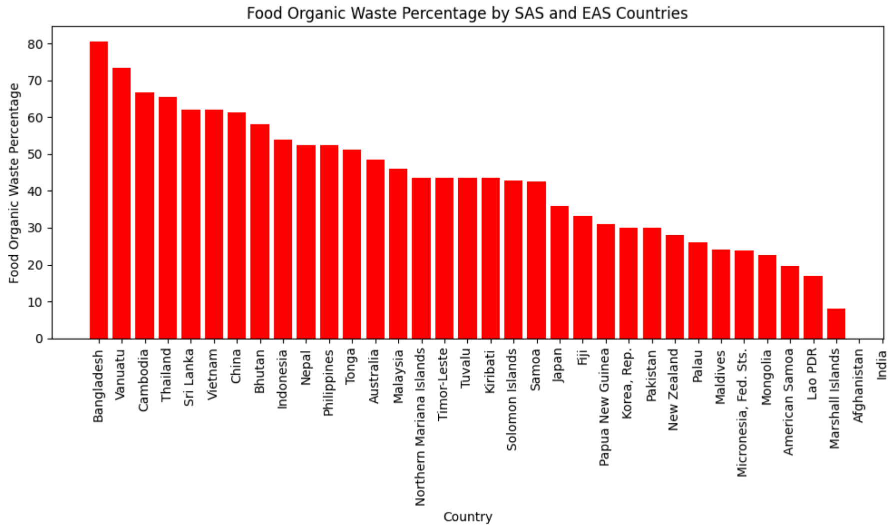
```
Asia = basic_data.query('region_idcd == "SAS" or region_idcd == "EAS"')
Asia
```

	iso3ccd	region_idcd	country_name	city_name	composition_food_organic_waste_perce
0	AFG	SAS	Afghanistan	Jalalabad	
1	AFG	SAS	Afghanistan	Kandahar	
2	AFG	SAS	Afghanistan	Mazar-E-Sharif	
3	AFG	SAS	Afghanistan	Kabul	
4	AFG	SAS	Afghanistan	HiratÂ	
...	...	...	...	...	
330	TUV	EAS	Tuvalu	Funafuti	
344	VNM	EAS	Vietnam	Hanoi	
345	VNM	EAS	Vietnam	Ho Chi Minh City	
346	VUT	EAS	Vanuatu	Port Vila	
[illegible]	[illegible]	[illegible]	[illegible]	Upolu	

Next steps: View recommended plots

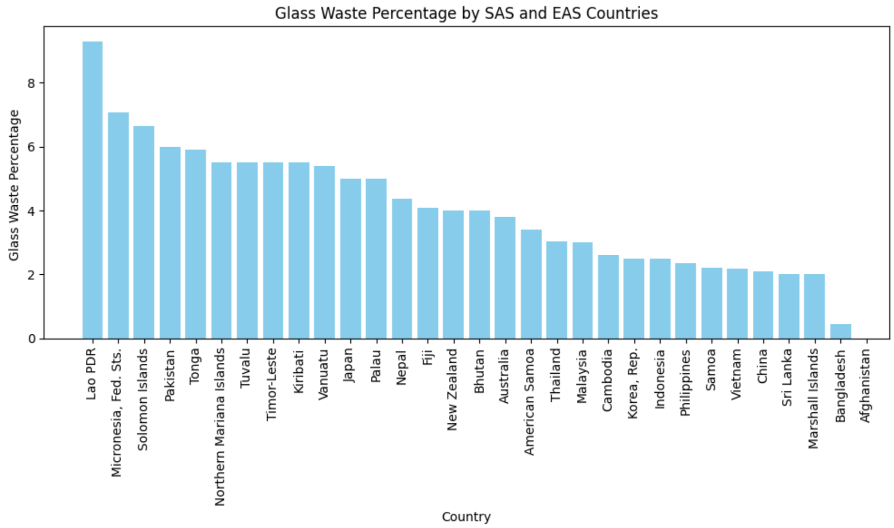
Plotting

```
# Food Waste for Asia Countries in Data
import matplotlib.pyplot as plt

# Sort the DataFrame by 'composition_food_organic_waste_percentcd' in descending order
food_waste = Asia.sort_values(by='composition_food_organic_waste_percentcd', ascending=False)

# Extract country names and their corresponding food organic waste percentages
country = food_waste['country_name']
organic_waste_percentages = food_waste['composition_food_organic_waste_percentcd']

# Create a bar plot
plt.figure(figsize=(10, 6))
plt.bar(country, organic_waste_percentages, color='red')
plt.xlabel('Country')
plt.ylabel('Food Organic Waste Percentage')
plt.title('Food Organic Waste Percentage by SAS and EAS Countries')
plt.xticks(rotation=90)  # Rotate x-axis labels for better readability
plt.tight_layout()
plt.show()
```

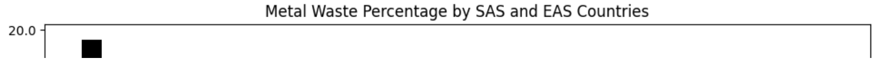


We can observe that Bangladesh has the highest Food Organic Waste in Asia in our Data set

```
# Glass Waste for Asia Countries in Data
compo_glass = Asia.sort_values(by='composition_glass_percentcd', ascending=False)

country2 = compo_glass['country_name']
glass_waste_percentages = compo_glass['composition_glass_percentcd']

plt.figure(figsize=(10, 6))
plt.bar(country2, glass_waste_percentages, color='skyblue')
plt.xlabel('Country')
plt.ylabel('Glass Waste Percentage ')
plt.title('Glass Waste Percentage by SAS and EAS Countries')
plt.xticks(rotation=90)  # Rotate x-axis labels for better readability
plt.tight_layout()
plt.show()
```




```
# Metal Waste for Asia Countries in Data
compo_metal = Asia.sort_values(by='composition_metal_percentcd', ascending=False)

country3 = compo_metal['country_name']
metal_waste_percentages = compo_metal['composition_metal_percentcd']

plt.figure(figsize=(10, 6))
plt.bar(country3, metal_waste_percentages, color='black')
plt.xlabel('Country')
plt.ylabel('Metal Waste Percentage ')
plt.title('Metal Waste Percentage by SAS and EAS Countries')
plt.xticks(rotation=90)
plt.tight_layout()
plt.show()
```




```
# Other Waste for Asia Countries in Data
compo_other = Asia.sort_values(by='composition_other_percentcd', ascending=False)
```