In [6]:
```
#extracting
```

In [ ]:
```
import pandas as pd
```

In [7]:
```
input_file = 'Desktop/om(DSA)/etl/water-and-sanitation.csv'
output_file = 'Desktop/om(DSA)/etl/transformed_water-and-sanitation.csv'
```

In [8]:
```
df = pd.read_csv(input_file)
```

In [9]:
```
df.head()
```

Out[9]:

Usage of limited drinking water services	Usage of unimproved drinking water sources	No usage of drinking water facilities	Usage of safely managed drinking water services	Usage of improved sanitation facilities	Usage of basic sanitation services	...	wat_pip_urban	wat_pip_number_rural	wat_pip_number	wat_pip_number_urban	wat_sm_r
3.299203	43.856777	25.402164	11.093327	26.466162	NaN	...	19.063290	0.00	822523.7	822523.7	
3.299883	43.843445	25.383093	11.105221	26.488068	NaN	...	19.063290	0.00	832069.3	832069.3	
3.607177	42.260395	24.457567	12.007733	28.414984	NaN	...	20.168760	0.00	942862.7	942862.7	
3.914072	40.677280	23.533058	12.909922	30.342781	NaN	...	21.274233	142883.52	1219756.8	1076873.1	
4.220617	39.086002	22.598950	13.818684	32.285492	NaN	...	22.379705	299784.84	1485808.1	1186023.4	

In [10]:
```
#transform
```

In [ ]:
```
df = df.dropna()
```

In [11]:
```
df.info
```

Out[11]:
```
<bound method DataFrame.info of        Country  Year  Usage of improved drinking water sources  \
76     Algeria  2007                                 96.996520
77     Algeria  2008                                 97.208450
78     Algeria  2009                                 97.415550
79     Algeria  2010                                 97.618164
80     Algeria  2011                                 97.816310
...        ...   ...                                       ...
5732  Zimbabwe  2018                                 77.055710
5733  Zimbabwe  2019                                 76.955050
5734  Zimbabwe  2020                                 76.864000
5735  Zimbabwe  2021                                 76.782080
5736  Zimbabwe  2022                                 76.810740

      Usage of basic drinking water services  \
76                                 17.888512
77                                 17.660484
78                                 17.444347
79                                 17.239166
80                                 17.045073
```

In [12]:
```
df.head()
```

Out[12]:

	Country	Year	Usage of improved drinking water sources	Usage of basic drinking water services	Usage of limited drinking water services	Usage of unimproved drinking water sources	No usage of drinking water facilities	Usage of safely managed drinking water services	Usage of improved sanitation facilities	Usage of basic sanitation services	...	wat_pip_urban	wat_pip_number_rural	wat_pip_r
76	Algeria	2007	96.996520	17.888512	5.387287	2.697115	0.306367	73.720720	94.12311	25.189022	...	85.460900	7648207.0	266

In [12]:
```
df.head()
```

Out[12]:

	Country	Year	Usage of improved drinking water sources	Usage of basic drinking water services	Usage of limited drinking water services	Usage of unimproved drinking water sources	No usage of drinking water facilities	Usage of safely managed drinking water services	Usage of improved sanitation facilities	Usage of basic sanitation services	...	wat_pip_urban	wat_pip_number_rural	wat_pip_r
76	Algeria	2007	96.996520	17.888512	5.387287	2.697115	0.306367	73.720720	94.12311	25.189022	...	85.460900	7648207.0	266
77	Algeria	2008	97.208450	17.660484	5.365142	2.506071	0.285478	74.182820	94.34653	25.125036	...	84.741210	7566640.5	269
78	Algeria	2009	97.415550	17.444347	5.343516	2.319164	0.265285	74.627686	94.56176	25.060670	...	84.021520	7493047.0	272
79	Algeria	2010	97.618164	17.239166	5.322189	2.136100	0.245737	75.056810	94.76960	24.995966	...	83.301834	7424406.5	275
80	Algeria	2011	97.816310	17.045073	5.301317	1.956861	0.226832	75.469920	94.96990	24.930914	...	82.582146	7359657.0	279

5 rows × 206 columns

In [13]:
```
df.isnull().sum()
```

Out[13]:
```
Country                                  0
Year                                     0
Usage of improved drinking water sources    0
Usage of basic drinking water services      0
Usage of limited drinking water services    0
                                        ..
wat_sm_number_without                    0
wat_sm_number_without_urban              0
wat_sm_without_rural                     0
wat_sm_without                           0
wat_sm_without_urban                     0
Length: 206, dtype: int64
```

In [14]:
```
# 2. Normalize column names: convert to lower case and replace spaces with underscores
df.columns = [col.strip().lower().replace(' ', '_') for col in df.columns]
```

In [15]:
```
df.head()
```

Out[15]:

	country	year	usage_of_improved_drinking_water_sources	usage_of_basic_drinking_water_services	usage_of_limited_drinking_water_services	usage_of_unim
76	Algeria	2007	96.996520	17.888512	5.387287	
77	Algeria	2008	97.208450	17.660484	5.365142	
78	Algeria	2009	97.415550	17.444347	5.343516	
79	Algeria	2010	97.618164	17.239166	5.322189	
80	Algeria	2011	97.816310	17.045073	5.301317	

5 rows × 206 columns

In [16]:
```
# 3. Convert 'year' column to numeric
df['year'] = pd.to_numeric(df['year'], errors='coerce')
```

In [17]:
```
# 4. Convert specific columns to numeric (e.g., percentage and numeric columns)
numeric_columns = [
    'usage_of_improved_drinking_water_sources',
    'usage_of_basic_drinking_water_services',
    'usage_of_limited_drinking_water_services',
    'usage_of_unimproved_drinking_water_sources',
    'no_usage_of_drinking_water_facilities',
    'usage_of_safely_managed_drinking_water_services',
    'usage_of_improved_sanitation_facilities',
    'usage_of_basic_sanitation_services',
    'usage_of_limited_sanitation_services',
    'usage_of_unimproved_sanitation_facilities',
    'no_usage_of_sanitation_facilities'
```

In [17]:

```
# 4. Convert specific columns to numeric (e.g., percentage and numeric columns)
numeric_columns = [
    'usage_of_improved_drinking_water_sources',
    'usage_of_basic_drinking_water_services',
    'usage_of_limited_drinking_water_services',
    'usage_of_unimproved_drinking_water_sources',
    'no_usage_of_drinking_water_facilities',
    'usage_of_safely_managed_drinking_water_services',
    'usage_of_improved_sanitation_facilities',
    'usage_of_basic_sanitation_services',
    'usage_of_limited_sanitation_services',
    'usage_of_unimproved_sanitation_facilities',
    'no_usage_of_sanitation_facilities',
    'population_total',
    'population_urban',
    'population_rural',
    'pop_den',
    'area',
    'gdp',
    'gdp_per_capita',
    'improved_san_coverage_rural',
    'improved_san_coverage_urban',
    'improved_san_coverage_total',
    'basic_san_coverage_rural',
    'basic_san_coverage_urban',
    'basic_san_coverage_total',
    'limited_san_coverage_rural',
    'limited_san_coverage_urban',
    'limited_san_coverage_total',
    'unimproved_san_coverage_rural',
    'unimproved_san_coverage_urban',
    'unimproved_san_coverage_total',
    'open_defecation_rural',
    'open_defecation_urban',
    'open_defecation_total'
]
```

In [20]:
```
for col in numeric_columns:
    if col in df.columns:
        df[col] = pd.to_numeric(df[col], errors='coerce')
```

In [21]:
```
df.head()
```

Out[21]:

	country	year	usage_of_improved_drinking_water_sources	usage_of_basic_drinking_water_services	usage_of_limited_drinking_water_services	usage_of_unim
76	Algeria	2007	96.996520	17.888512	5.387287	
77	Algeria	2008	97.208450	17.660484	5.365142	
78	Algeria	2009	97.415550	17.444347	5.343516	
79	Algeria	2010	97.618164	17.239166	5.322189	
80	Algeria	2011	97.816310	17.045073	5.301317	

5 rows × 206 columns

In [42]:
```
for col in numeric_columns:
    if col in df.columns:
        df[col] = pd.to_numeric(df[col], errors='coerce')
        # Round off the values to the nearest integer
        df[col] = df[col].round()
```

In [45]:
```
df.head()
```

Out[45]:

	country	year	usage_of_improved_drinking_water_sources	usage_of_basic_drinking_water_services	usage_of_limited_drinking_water_services	usage_of_unim
76	Algeria	2007	97.0	18.0	5.0	
77	Algeria	2008	97.0	18.0	5.0	
78	Algeria	2009	97.0	17.0	5.0	
79	Algeria	2010	98.0	17.0	5.0	
80	Algeria	2011	98.0	17.0	5.0	

5 rows × 206 columns

In [46]:
```
# Load: Write the transformed DataFrame to a new CSV file
df.to_csv(output_file, index=False)
```

In [47]:
```
print("ETL process completed successfully.")
```

```
ETL process completed successfully.
```