Python and SQL for Extracting, Transforming, and Loading (ETL), and Visualization with Power BI for Non-Profit Organizations

Project Context:

Hi Yedou,
I remember you mentioned that you had some programming knowledge with Python and SQL. I have a small project in preparation for our Nonprofit Organization and would appreciate your help.
Could you use your Python and SQL skills to create a database for managing and analyzing our activities?
Best, Sydney

Conversation:

Yedou (Developer): "Hello Sydney! Have you collected or saved your activity data into a CSV file or database that I can use to extract the data needed for your project?"
Sydney (Client): "No, we don't have data stored this way."
Yedou (Developer): "Okay, no problem. I can pull the data I need from the API to complete the project for you. What are the objectives of the project?
"Sydney (Client): "Currently, our budgets are limited, so we want to identify which countries should be our priorities for humanitarian aid and determine the priority sectors within these countries."

Data-Driven Decision-Making:

Before initiating a humanitarian response, an organization needs to gather comprehensive information about the country or continent to ensure an effective and appropriate intervention. Key aspects to consider include:

1. **Geopolitical Context**
   - **Government and Political Stability**: Understand the current political situation, government structure, stability, and any ongoing conflicts or tensions.
   - **Legal Framework**: Familiarize with local laws and regulations that might affect the operation, including humanitarian laws, import/export restrictions, and NGO regulations.
2. **Demographics and Population**
   - **Population Size and Distribution**: Data on population density, urban vs. rural distribution, age distribution, and key demographic groups.
   - **Vulnerable Populations**: Identify groups particularly vulnerable to crises, such as children, elderly, disabled, and marginalized communities.
3. **Economic Conditions**
   - **Economic Indicators**: GDP, unemployment rates, inflation, poverty levels, and overall economic stability.
   - **Infrastructure**: Condition of transportation, communication, and utilities infrastructure.
4. **Health and Nutrition**
   - **Health Infrastructure**: Availability and quality of healthcare facilities, medical supplies, and personnel.
   - **Disease Burden**: Prevalence of communicable and non-communicable diseases, vaccination coverage, and existing public health threats.
   - **Nutritional Status**: Rates of malnutrition, food security status, and availability of essential food items.

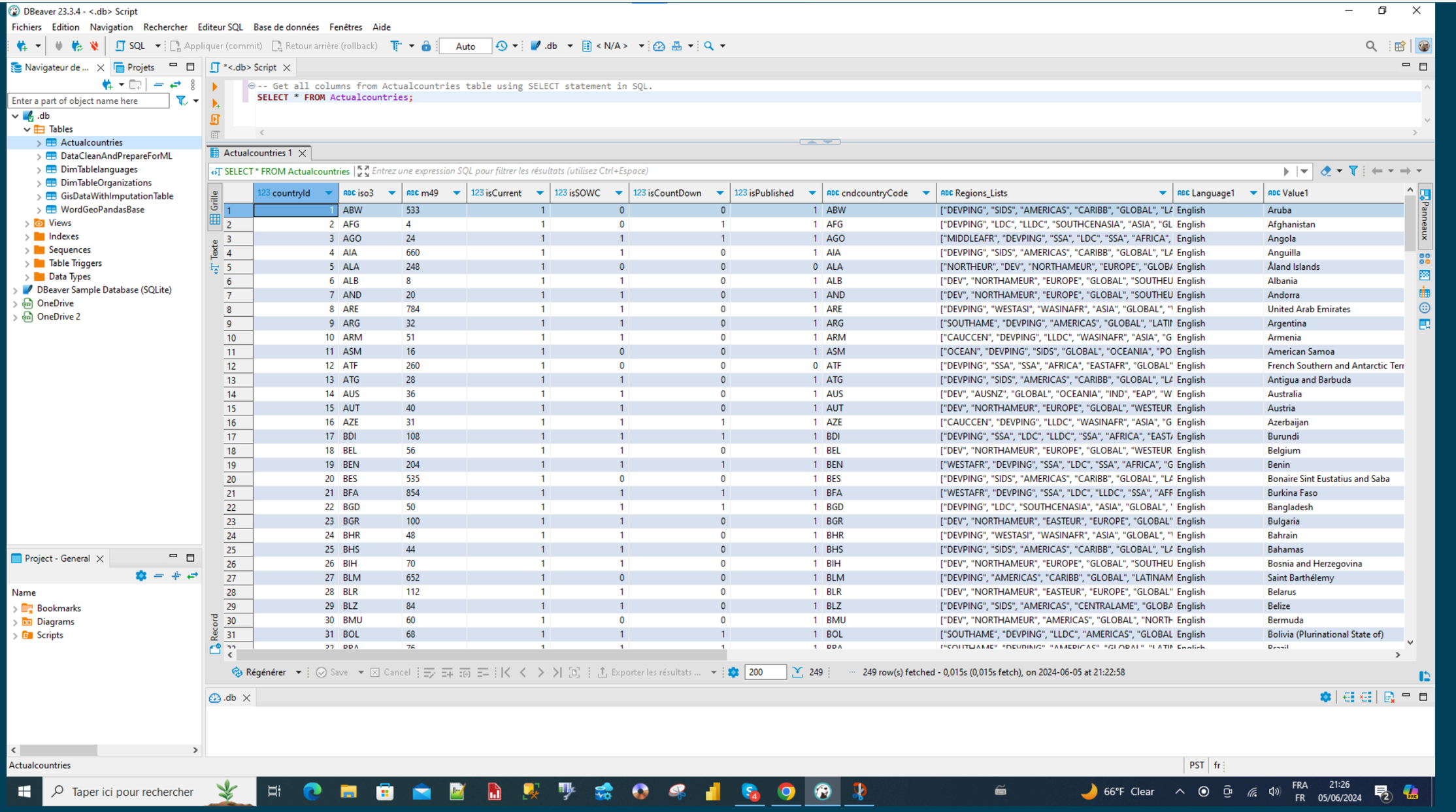
Collecting this information enables the organization to create a contextually appropriate, efficient, and effective humanitarian response that aligns with the needs and conditions of the affected country or continent.
To achieve this, we gather data from the UNICEF API at the following base URL: https://rdmapi.unicef.org/api/doc/index.html. Additionally, we utilize resources from websites such as https://www.learndatasci.com/tutorials/geospatial-data-python-geopandas-shapely/ and https://datascience.quantecon.org/tools/maps.html.

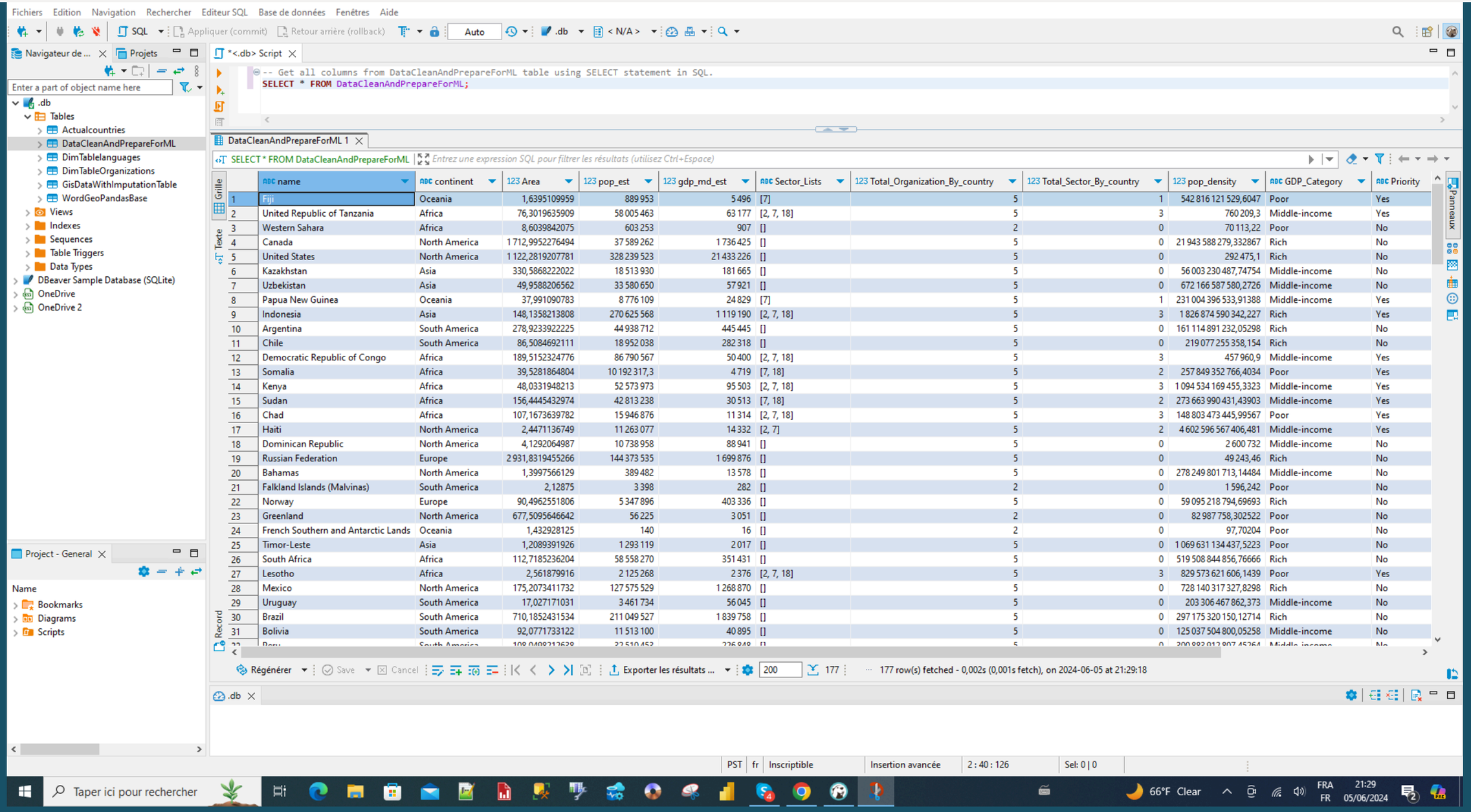
```
import ast
def get_data(endpoint):
    """
    Function to make a GET request to the specified endpoint and return the response as pandas DataFrame.
    """
    try:
        response = r.get(base_url + endpoint)
        response.raise_for_status()  # Raise an exception for HTTP errors
        return pd.DataFrame(response.json())
    except r.exceptions.RequestException as e:
        print("Error:", e)
        return None
# Base URL for the UNICEF API
# https://rdmapi.unicef.org/api/doc/index.html
base_url = "https://rdmapi.unicef.org"
# Endpoint: /api/countries
df_countries = get_data("/api/countries")
if df_countries is not None:
    #print("\nDataFrame from /api/countries:")
    df_countries
df_languages = get_data("/api/languages")
if df_languages is not None:
    #print("\nDataFrame from /api/countries:")
    df_languages
# Endpoint: /api/countries/current
df_current = get_data("/api/countries/current")
if df_current is not None:
    #print("\nDataFrame from /api/countries/current:")
    df_current
# Endpoint: /api/countries/organizations
df_organizations = get_data("/api/countries/organizations")
if df_organizations is not None:
    #print("\nDataFrame from /api/countries/organizations:")
    df_organizations
```

```
import sqlite3
try:
    conn = sqlite3.connect('unicef.db')
    print("Opened database successfully")
except sqlite3.Error as e:
    print("Error connecting to SQLite database:", e)

DB ="/content/unicef.db"
def create_and_load_table(df, table_name, db_name):
    try:
        # Connect to SQLite database
        conn = sqlite3.connect(db_name)
        print("Opened database successfully")

        # Convert lists to strings
        df_copy = df.copy()  # Create a copy to avoid modifying the original DataFrame
        df_copy['Regions_Lists'] = df_copy['Regions_Lists'].apply(json.dumps)
        df_copy['Sector_Lists'] = df_copy['Sector_Lists'].apply(json.dumps)

        # Write DataFrame to SQLite database
        df_copy.to_sql(table_name, conn, if_exists='replace', index=False)
        print("Table created and loaded successfully")

    except sqlite3.Error as e:
        print("Error connecting to SQLite database:", e)
    finally:
        # Close connection
        conn.close()

create_and_load_table(CurrentcountriesTable, "Actualcountries", DB)
```

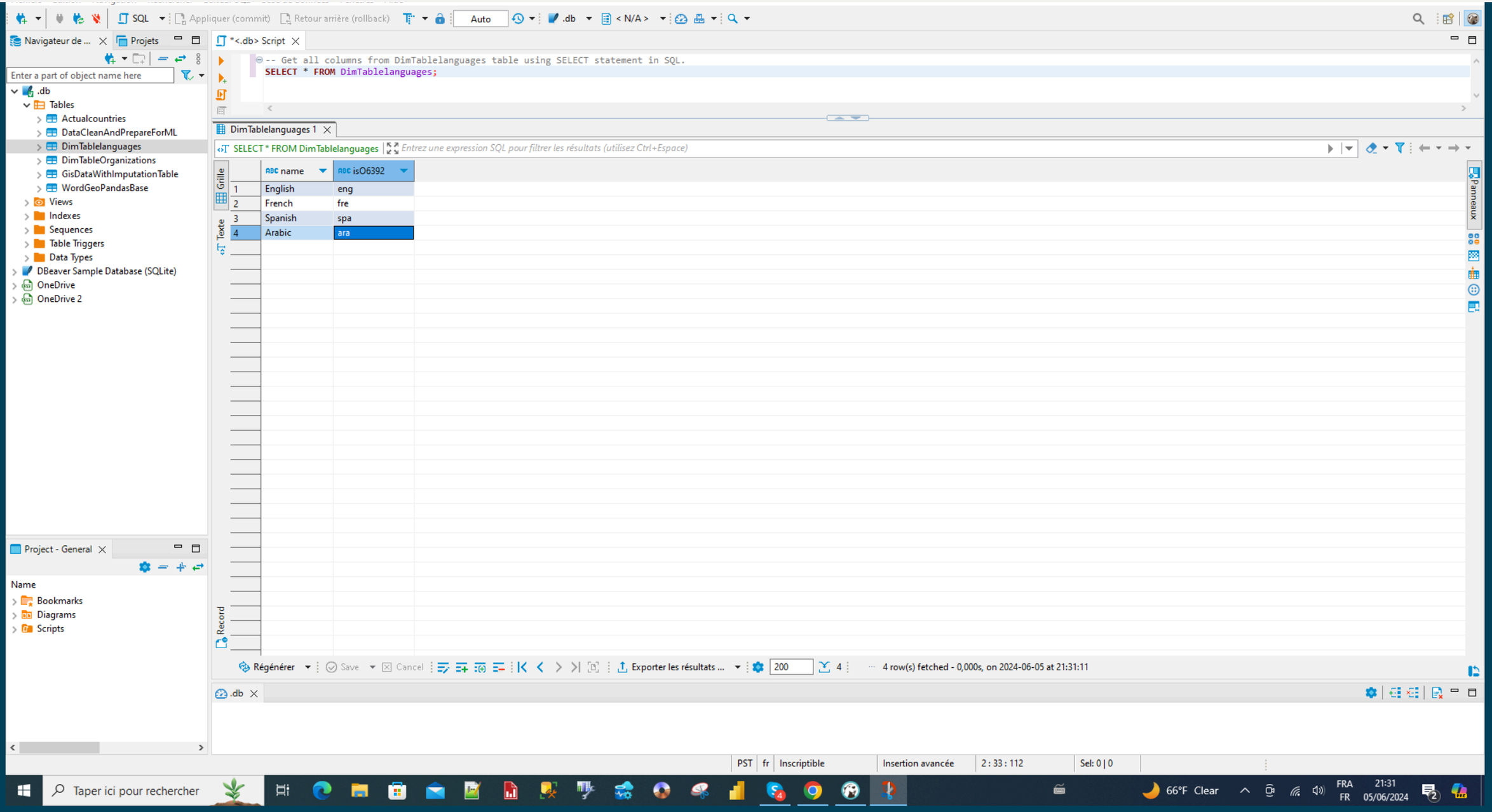
```
    try:
        # Connect to SQLite database
        conn = sqlite3.connect(db_name)
        print("Opened database successfully")
        # Convert lists to strings
        df_copy = df.copy()  # Create a copy to avoid modifying the original DataFrame
        df_copy['Regions_Lists'] = df_copy['Regions_Lists'].apply(json.dumps)
        df_copy['Sector_Lists'] = df_copy['Sector_Lists'].apply(json.dumps)

        # Write DataFrame to SQLite database
        df_copy.to_sql(table_name, conn, if_exists='replace', index=False)
        print("Table created and loaded successfully")
    except sqlite3.Error as e:
        print("Error connecting to SQLite database:", e)
    finally:
        # Close connection
        conn.close()
create_and_load_table(CurrentcountriesTable, "Actualcountries", DB)
def create_and_load_table(df, table_name, db_name):
    try:
        # Connect to SQLite database
        conn = sqlite3.connect(db_name)
        print("Opened database successfully")

        # Write DataFrame to SQLite database
        df.to_sql(table_name, conn, if_exists='replace', index=False)
        print("Table created and loaded successfully")

    except sqlite3.Error as e:
        print("Error connecting to SQLite database:", e)
    finally:
        # Close connection
        conn.close()
# Example usage to load DimTablelanguages and  DimTableOrganizations to database DB
create_and_load_table(DimTablelanguages, "DimTablelanguages", DB)
create_and_load_table(DimTableOrganizations, "DimTableOrganizations", DB)
```

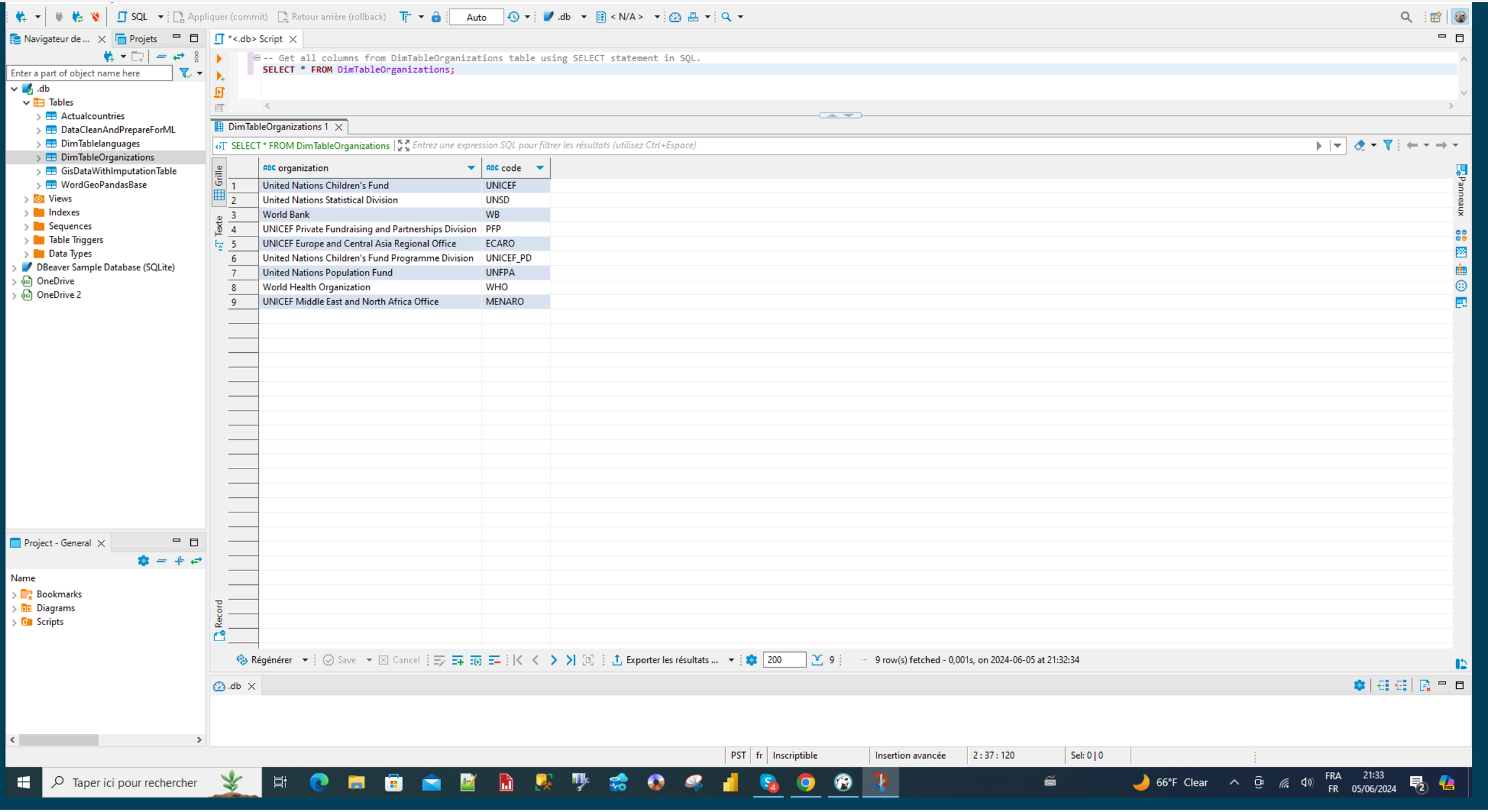
```
    try:
        # Connect to SQLite database
        conn = sqlite3.connect(db_name)
        print("Opened database successfully")
        # Convert unsupported types to strings
        df_copy = df.copy()  # Create a copy to avoid modifying the original DataFrame
        for column in df_copy.columns:
            if df_copy[column].dtype == 'object':
                # Convert lists, dicts, or other unsupported types to JSON strings
                df_copy[column] = df_copy[column].apply(lambda x: json.dumps(x) if isinstance(x, (list, dict)) else x)
        # Write DataFrame to SQLite database
        df_copy.to_sql(table_name, conn, if_exists='replace', index=False)
        print(f"Table '{table_name}' created and loaded successfully")
    except sqlite3.Error as e:
        print(f"Error connecting to SQLite database: {e}")
    finally:
        # Close connection
        conn.close()
# Create and load tables
create_and_load_table(DataCleanAndPrepareForML, "DataCleanAndPrepareForML", DB)

def create_and_load_table(df, table_name, db_name):
    try:
        # Connect to SQLite database
        conn = sqlite3.connect(db_name)
        print("Opened database successfully")
        # Convert GeoPandas geometries to WKT (Well-Known Text)
        df_copy = df.copy()  # Create a copy to avoid modifying the original DataFrame
        if isinstance(df_copy, gpd.GeoDataFrame):
            df_copy['geometry'] = df_copy['geometry'].apply(lambda geom: geom.wkt)
        # Write DataFrame to SQLite database
        df_copy.to_sql(table_name, conn, if_exists='replace', index=False)
        print(f"Table '{table_name}' created and loaded successfully")
    except sqlite3.Error as e:
        print(f"Error connecting to SQLite database: {e}")
    finally:
        # Close connection
        conn.close()
create_and_load_table(WordGeoPandasBase, "WordGeoPandasBase", DB)
create_and_load_table(GisDataWithImputationTable, "GisDataWithImputationTable", DB)
```

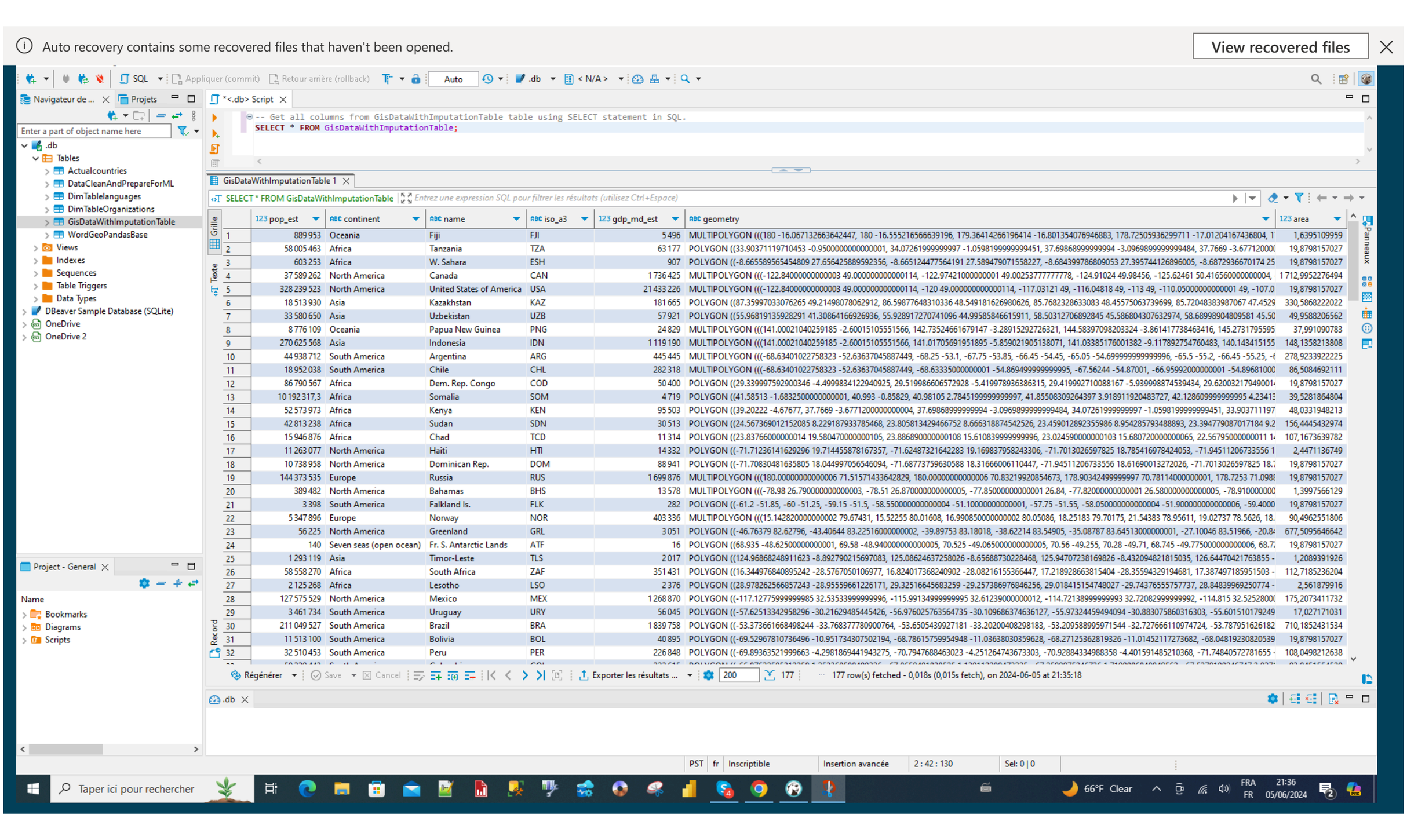




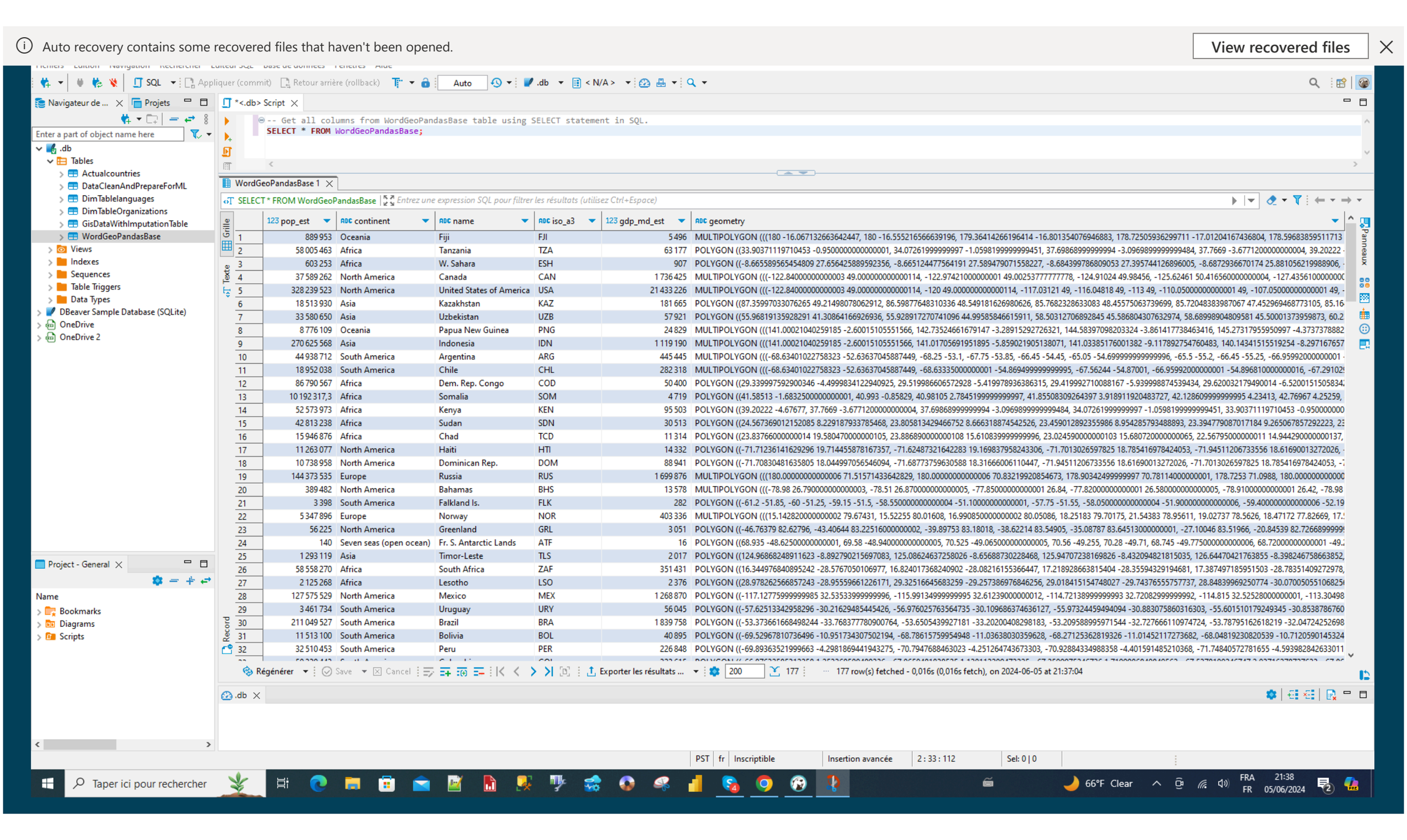
	name	continent	Area	pop_est	gdp_md_est	Sector_Lists	Total_Organization_By_country	Total_Sector_By_country	pop_density	GDP_Category	Priority
1	Fiji	Oceania	1,6395109959	889 953	5 496	[7]	5	1	542 816 121 529,6047	Poor	Yes
2	United Republic of Tanzania	Africa	76,3019635909	58 005 463	63 177	[2, 7, 18]	5	3	760 209,3	Middle-income	Yes
3	Western Sahara	Africa	8,6039842075	603 253	907	[]	2	0	70 113,22	Poor	No
4	Canada	North America	1 712,9952276494	37 589 262	1 736 425	[]	5	0	21 943 588 279,332867	Rich	No
5	United States	North America	1 122,2819207781	328 239 523	21 433 226	[]	5	0	292 475,1	Rich	No
6	Kazakhstan	Asia	330,5868222022	18 513 930	181 665	[]	5	0	56 003 230 487,74754	Middle-income	No
7	Uzbekistan	Asia	49,9588206562	33 580 650	57 921	[]	5	0	672 166 587 580,2726	Middle-income	No
8	Papua New Guinea	Oceania	37,991090783	8 776 109	24 829	[7]	5	1	231 004 396 533,91388	Middle-income	Yes
9	Indonesia	Asia	148,1358213808	270 625 568	1 119 190	[2, 7, 18]	5	3	1 826 874 590 342,227	Rich	Yes
10	Argentina	South America	278,9233922225	44 938 712	445 445	[]	5	0	161 114 891 232,05298	Rich	No
11	Chile	South America	86,5084692111	18 952 038	282 318	[]	5	0	219 077 255 358,154	Rich	No
12	Democratic Republic of Congo	Africa	189,5152324776	86 790 567	50 400	[2, 7, 18]	5	3	457 960,9	Middle-income	Yes
13	Somalia	Africa	39,5281864804	10 192 317,3	4 719	[7, 18]	5	2	257 849 352 766,4034	Poor	Yes
14	Kenya	Africa	48,0331948213	52 573 973	95 503	[2, 7, 18]	5	3	1 094 534 169 455,3323	Middle-income	Yes
15	Sudan	Africa	156,4445432974	42 813 238	30 513	[7, 18]	5	2	273 663 990 431,43903	Middle-income	Yes
16	Chad	Africa	107,1673639782	15 946 876	11 314	[2, 7, 18]	5	3	148 803 473 445,99567	Poor	Yes
17	Haiti	North America	2,4471136749	11 263 077	14 332	[2, 7]	5	2	4 602 596 567 406,481	Middle-income	Yes
18	Dominican Republic	North America	4,1292064987	10 738 958	88 941	[]	5	0	2 600 732	Middle-income	No
19	Russian Federation	Europe	2 931,8319455266	144 373 535	1 699 876	[]	5	0	49 243,46	Rich	No
20	Bahamas	North America	1,3997566129	389 482	13 578	[]	5	0	278 249 801 713,14484	Middle-income	No
21	Falkland Islands (Malvinas)	South America	2,12875	3 398	282	[]	2	0	1 596,242	Poor	No
22	Norway	Europe	90,4962551806	5 347 896	403 336	[]	5	0	59 095 218 794,69693	Rich	No
23	Greenland	North America	677,5095646642	56 225	3 051	[]	2	0	82 987 758,302522	Poor	No
24	French Southern and Antarctic Lands	Oceania	1,432928125	140	16	[]	2	0	97,70204	Poor	No
25	Timor-Leste	Asia	1,2089391926	1 293 119	2 017	[]	5	0	1 069 631 134 437,5223	Poor	No
26	South Africa	Africa	112,7185236204	58 558 270	351 431	[]	5	0	519 508 844 856,76666	Rich	No
27	Lesotho	Africa	2,561879916	2 125 268	2 376	[2, 7, 18]	5	3	829 573 621 606,1439	Poor	Yes
28	Mexico	North America	175,2073411732	127 575 529	1 268 870	[]	5	0	728 140 317 327,8298	Rich	No
29	Uruguay	South America	17,027171031	3 461 734	56 045	[]	5	0	203 306 467 862,373	Middle-income	No
30	Brazil	South America	710,1852431534	211 049 527	1 839 758	[]	5	0	297 175 320 150,12714	Rich	No
31	Bolivia	South America	92,0771733122	11 513 100	40 895	[]	5	0	125 037 504 800,05258	Middle-income	No

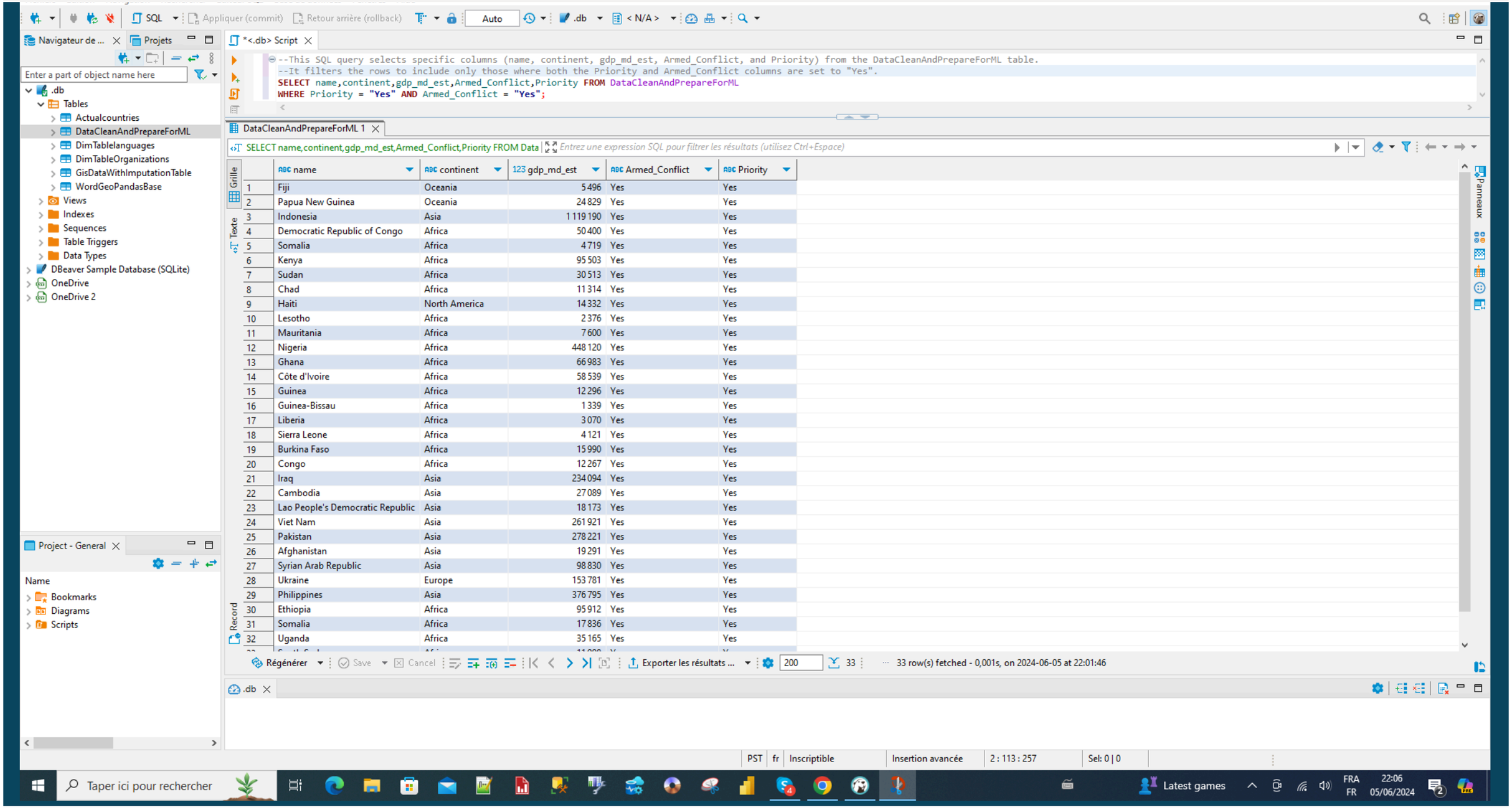




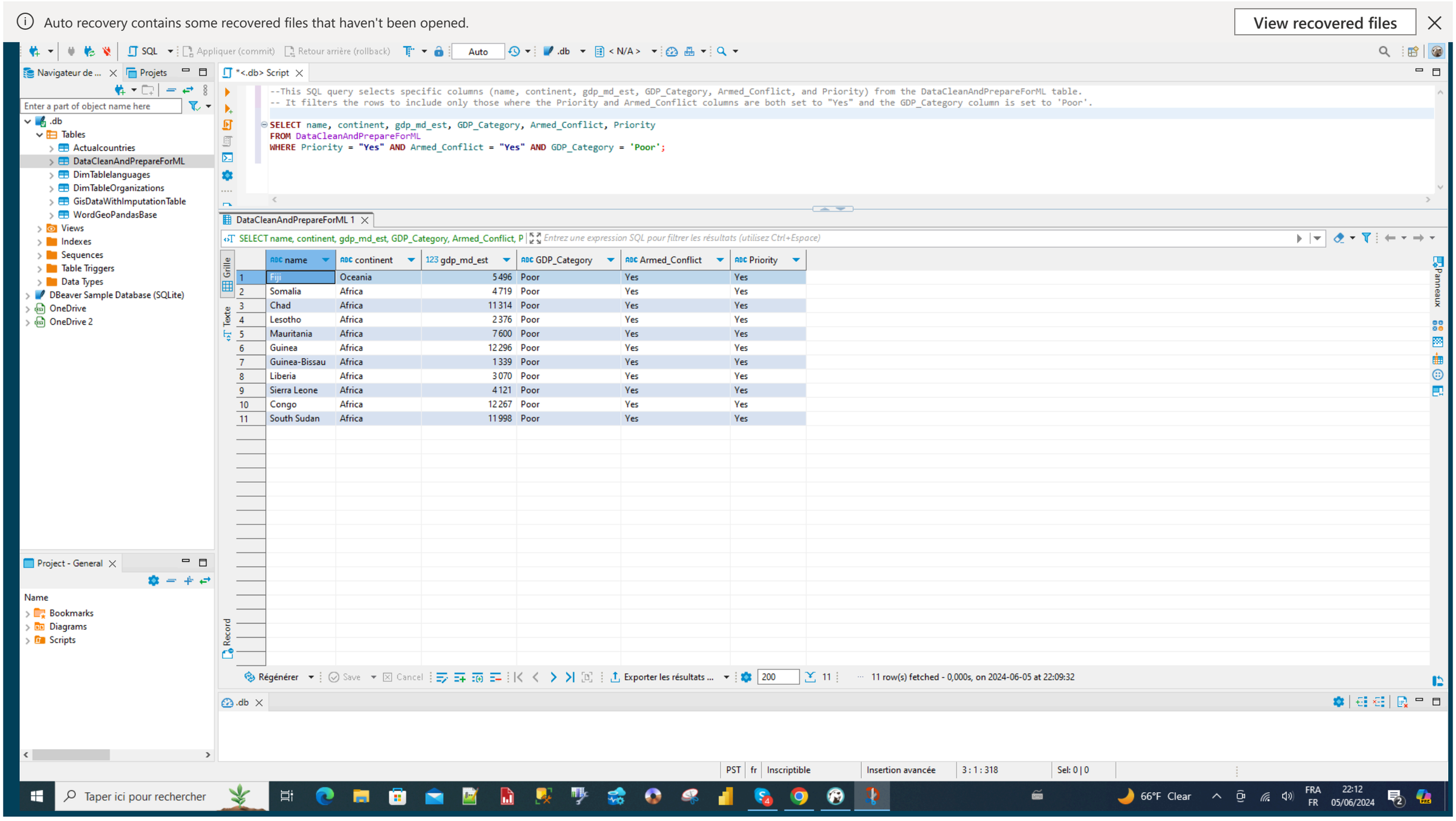


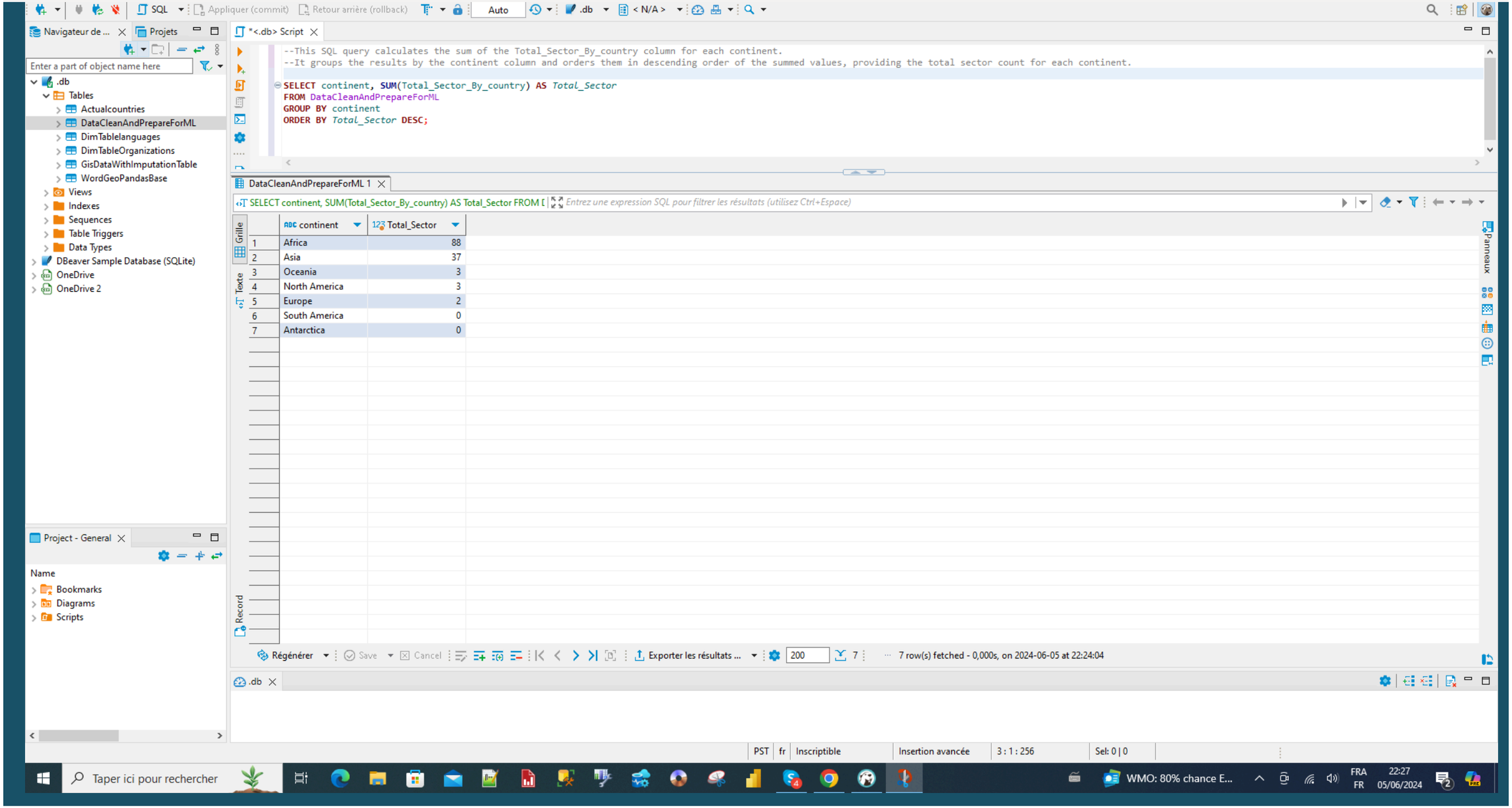
	123 pop_est	ABC continent	ABC name	ABC iso_a3	123 gdp_md_est	ABC geometry	123 area
1	889 953	Oceania	Fiji	FJI	5 496	MULTIPOLYGON (((180 -16.067132663642447, 180 -16.555216566639196, 179.36414266196414 -16.801354076946883, 178.72505936299711 -17.01204167436804, 1	1,6395109959
2	58 005 463	Africa	Tanzania	TZA	63 177	POLYGON ((33.90371119710453 -0.9500000000000001, 34.07261999999997 -1.0598199999999451, 37.69868999999994 -3.0969899999999484, 37.7669 -3.677120000	19,8798157027
3	603 253	Africa	W. Sahara	ESH	907	POLYGON ((-8.665589565454809 27.656425889592356, -8.665124477564191 27.589479071558227, -8.684399786809053 27.395744126896005, -8.6872936670174 25	19,8798157027
4	37 589 262	North America	Canada	CAN	1 736 425	MULTIPOLYGON (((-122.84000000000003 49.000000000000114, -122.97421000000001 49.00253777777778, -124.91024 49.98456, -125.62461 50.416560000000004,	1 712,9952276494
5	328 239 523	North America	United States of America	USA	21 433 226	MULTIPOLYGON (((-122.84000000000003 49.000000000000114, -120 49.000000000000114, -117.03121 49, -116.04818 49, -113 49, -110.05000000000001 49, -107.0	19,8798157027
6	18 513 930	Asia	Kazakhstan	KAZ	181 665	POLYGON ((87.35997033076265 49.21498078062912, 86.59877648310336 48.549181626980626, 85.7682328633083 48.45575063739699, 85.72048383987067 47.4529	330,5868222022
7	33 580 650	Asia	Uzbekistan	UZB	57 921	POLYGON ((55.96819135928291 41.30864166926936, 55.928917270741096 44.99585846615911, 58.50312706892845 45.58680430763297 4, 58.68998904809581 45.50	49,9588206562
8	8 776 109	Oceania	Papua New Guinea	PNG	24 829	MULTIPOLYGON (((141.00021040259185 -2.60015105551566, 142.73524661679147 -3.28915292726321, 144.58397098203324 -3.861417738463416, 145.2731795595	37,991090783
9	270 625 568	Asia	Indonesia	IDN	1 119 190	MULTIPOLYGON (((141.00021040259185 -2.60015105551566, 141.01705691951895 -5.859021905138071, 141.03385176001382 -9.117892754760483, 140.143415155	148,1358213808
10	44 938 712	South America	Argentina	ARG	445 445	MULTIPOLYGON (((-68.63401022758323 -52.63637045887449, -68.25 -53.1, -67.75 -53.85, -66.45 -54.45, -65.05 -54.699999999999996, -65.5 -55.2, -66.45 -55.25, -(	278,9233922225
11	18 952 038	South America	Chile	CHL	282 318	MULTIPOLYGON (((-68.63401022758323 -52.63637045887449, -68.63335000000001 -54.869499999999995, -67.56244 -54.87001, -66.95992000000001 -54.89681000	86,5084692111
12	86 790 567	Africa	Dem. Rep. Congo	COD	50 400	POLYGON ((29.339997592900346 -4.499983412294092 5, 29.519986606572928 -5.419978936386315, 29.419992710088167 -5.939998874539434, 29.62003217949001	19,8798157027
13	10 192 317,3	Africa	Somalia	SOM	4 719	POLYGON ((41.58513 -1.683250000000001, 40.993 -0.85829, 40.98105 2.7845199999999997, 41.85508309264397 3.918911920483727, 42.128609999999995 4.2313	39,5281864804
14	52 573 973	Africa	Kenya	KEN	95 503	POLYGON ((39.20222 -4.67677, 37.7669 -3.6771200000000004, 37.69868999999994 -3.0969899999999484, 34.07261999999997 -1.0598199999999451, 33.903711197	48,0331948213
15	42 813 238	Africa	Sudan	SDN	30 513	POLYGON ((24.567369012152085 8.229187933785468, 23.805813429466752 8.666318874542526, 23.459012892355986 8.954285793488893, 23.394779087017184 9.2	156,4445432974
16	15 946 876	Africa	Chad	TCD	11 314	POLYGON ((23.83766000000014 19.580470000000105, 23.886890000000108 15.610839999999996, 23.024590000000103 15.680720000000065, 22.567950000000011 1	107,1673639782
17	11 263 077	North America	Haiti	HTI	14 332	POLYGON ((-71.71236141629296 19.714455878167357, -71.62487321642283 19.169837958243306, -71.7013026597825 18.785416978424053, -71.945112067335561	2,4471136749
18	10 738 958	North America	Dominican Rep.	DOM	88 941	POLYGON ((-71.70830481635805 18.044997056546094, -71.68773759630588 18.31666006110447, -71.94511206733556 18.61690013272026, -71.7013026597825 18.7	19,8798157027
19	144 373 535	Europe	Russia	RUS	1 699 876	MULTIPOLYGON (((180.00000000000006 71.51571433642829, 180.00000000000006 70.83219920854673, 178.90342499999997 70.78114000000001, 178.7253 71.0988	19,8798157027
20	389 482	North America	Bahamas	BHS	13 578	MULTIPOLYGON (((-78.98 26.790000000000003, -78.51 26.870000000000005, -77.85000000000001 26.84, -77.82000000000001 26.580000000000005, -78.910000000	1,3997566129
21	3 398	South America	Falkland Is.	FLK	282	POLYGON ((-61.2 -51.85, -60 -51.25, -59.15 -51.5, -58.550000000000004 -51.10000000000001, -57.75 -51.55, -58.050000000000004 -51.900000000000006, -59.4000	19,8798157027
22	5 347 896	Europe	Norway	NOR	403 336	MULTIPOLYGON ((15.142820000000002 79.67431, 15.52255 80.01608, 16.990850000000002 80.05086, 18.25183 79.70175, 21.54383 78.95611, 19.02737 78.5626, 18.	90,4962551806
23	56 225	North America	Greenland	GRL	3 051	POLYGON ((-46.76379 82.62796, -43.40644 83.22516000000002, -39.89753 83.18018, -38.62214 83.54905, -35.08787 83.64513000000001, -27.10046 83.51966, -20.8	677,5095646642
24	140	Seven seas (open ocean)	Fr. S. Antarctic Lands	ATF	16	POLYGON ((68.935 -48.62500000000001, 69.58 -48.940000000000005, 70.525 -49.065000000000005, 70.56 -49.255, 70.28 -49.71, 68.745 -49.775000000000006, 68.7	19,8798157027
25	1 293 119	Asia	Timor-Leste	TLS	2 017	POLYGON ((124.96868248911623 -8.892790215697083, 125.08624637258026 -8.65688730228468, 125.94707238169826 -8.432094821815035, 126.64470421763855 -	1,2089391926
26	58 558 270	Africa	South Africa	ZAF	351 431	POLYGON ((16.344976840895242 -28.576705010697 7, 16.824017368240902 -28.08216155366447, 17.218928663815404 -28.355943291946 81, 17.387497185951503 -	112,7185236204
27	2 125 268	Africa	Lesotho	LSO	2 376	POLYGON ((28.978262566857243 -28.955596612261 71, 29.325166456832 59 -29.257386976846256, 29.018415154748027 -29.743765557737 37, 28.84839969250774 -	2,561879916
28	127 575 529	North America	Mexico	MEX	1 268 870	POLYGON ((-117.12775999999985 32.53533999999996, -115.99134999999995 32.61239000000012, -114.72138999999993 32.72082999999992, -114.815 32.5252800(	175,2073411732
29	3 461 734	South America	Uruguay	URY	56 045	POLYGON ((-57.62513342958296 -30.21629485445426, -56.976025763564735 -30.109686374636127, -55.97324459494094 -30.883075860316303, -55.601510179249	17,027171031
30	211 049 527	South America	Brazil	BRA	1 839 758	POLYGON ((-53.373661668498244 -33.768377780900764, -53.6505439927181 -33.20200408298183, -53.209588995971544 -32.727666110974724, -53.787951626182	710,1852431534
31	11 513 100	South America	Bolivia	BOL	40 895	POLYGON ((-69.52967810736496 -10.951734307502194, -68.78615759954948 -11.036380303596 28, -68.27125362819326 -11.014521172736 82, -68.04819230820539	19,8798157027
32	32 510 453	South America	Peru	PER	226 848	POLYGON ((-69.89363521999663 -4.2981869441943275, -70.7947688463023 -4.251264743673303, -70.92884334988358 -4.401591485210368, -71.74840572781655 -	108,0498212638

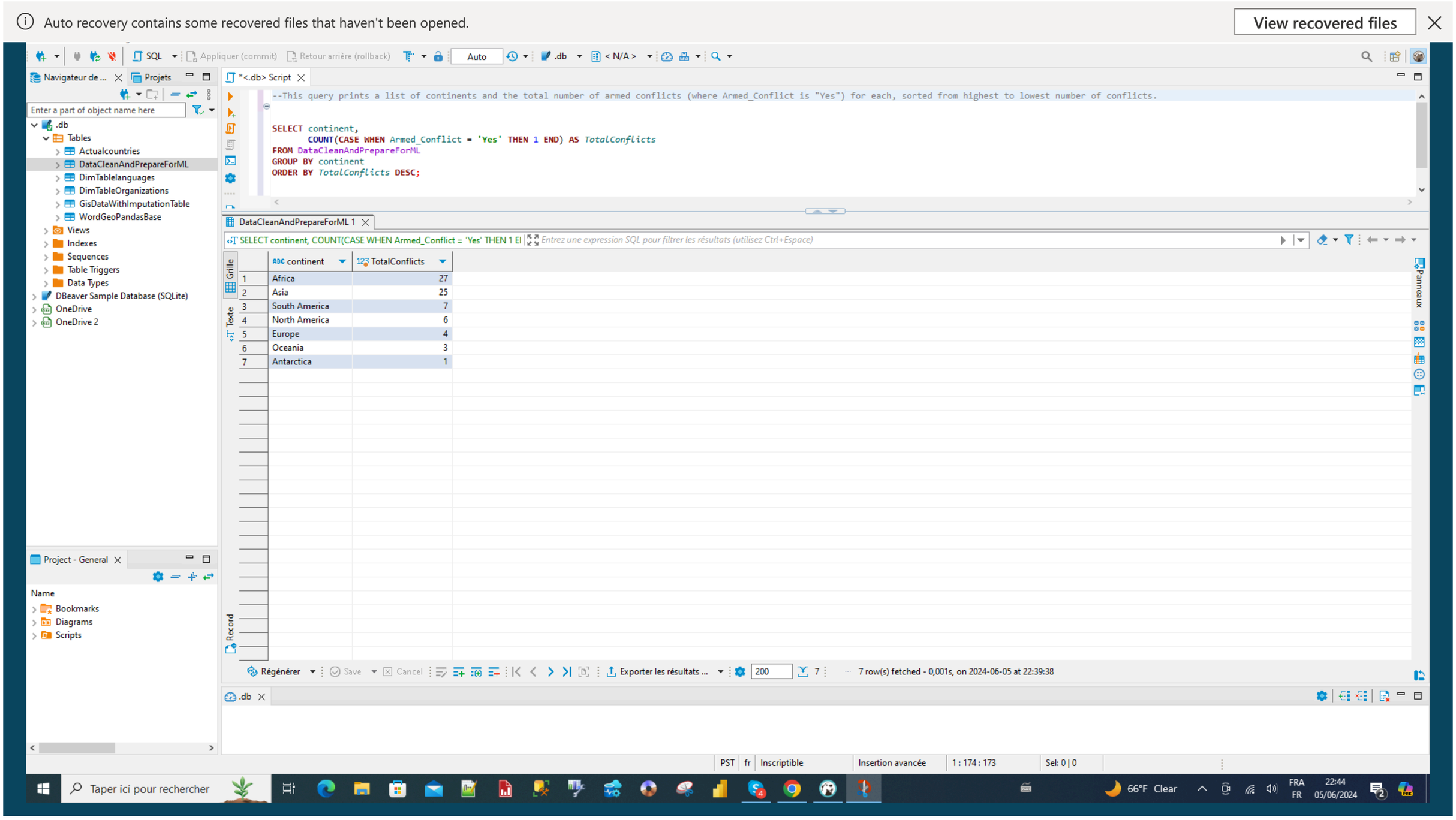

Auto recovery contains some recovered files that haven't been opened.
View recovered files
-- Get all columns from WordGeoPandasBase table using SELECT statement in SQL.
SELECT * FROM WordGeoPandasBase;
WordGeoPandasBase 1
SELECT * FROM WordGeoPandasBase
Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)
pop_est
continent
name
iso_a3
gdp_md_est
geometry
1 889 953 Oceania Fiji FJI 5 496 MULTIPOLYGON (((180 -16.067132663642447, 180 -16.555216566639196, 179.36414266196414 -16.801354076946883, 178.72505936299711 -17.01204167436804, 178.59683859511713
2 58 005 463 Africa Tanzania TZA 63 177 POLYGON ((33.90371119710453 -0.9500000000000001, 34.07261999999997 -1.0598199999999451, 37.69868999999994 -3.0969899999999484, 37.7669 -3.6771200000000004, 39.20222
3 603 253 Africa W. Sahara ESH 907 POLYGON ((-8.665589565454809 27.656425889592356, -8.665124477564191 27.589479071558227, -8.684399786809053 27.395744126896005, -8.6872936670174 25.881056219988906,
4 37 589 262 North America Canada CAN 1 736 425 MULTIPOLYGON (((-122.84000000000003 49.00000000000114, -122.97421000000001 49.00253777777778, -124.91024 49.98456, -125.62461 50.416560000000004, -127.43561000000
5 328 239 523 North America United States of America USA 21 433 226 MULTIPOLYGON (((-122.84000000000003 49.000000000000114, -120 49.000000000000114, -117.03121 49, -116.04818 49, -113 49, -110.05000000000001 49, -107.05000000000001 49,
6 18 513 930 Asia Kazakhstan KAZ 181 665 POLYGON ((87.35997033076265 49.21498078062912, 86.59877648310336 48.549181626980626, 85.76823286330 83 48.45575063739699, 85.72048383987067 47.452969468773105, 85.16
7 33 580 650 Asia Uzbekistan UZB 57 921 POLYGON ((55.96819135928291 41.30864166926936, 55.928917270741096 44.99585846615911, 58.50312706892845 45.586804307632974, 58.68998904809581 45.50001373959873, 60.2
8 8 776 109 Oceania Papua New Guinea PNG 24 829 MULTIPOLYGON (((141.00021040259185 -2.60015105551566, 142.73524661679147 -3.28915292726321, 144.58397098203324 -3.861417738463416, 145.27317955950997 -4.3737378882
9 270 625 568 Asia Indonesia IDN 1 119 190 MULTIPOLYGON (((141.00021040259185 -2.60015105551566, 141.01705691951895 -5.859021905138071, 141.03385176001382 -9.117892754760483, 140.14341515519254 -8.297167657
10 44 938 712 South America Argentina ARG 445 445 MULTIPOLYGON (((-68.63401022758323 -52.63637045887449, -68.25 -53.1, -67.75 -53.85, -66.45 -54.45, -65.05 -54.699999999999996, -65.5 -55.2, -66.45 -55.25, -66.95992000000001
11 18 952 038 South America Chile CHL 282 318 MULTIPOLYGON (((-68.63401022758323 -52.63637045887449, -68.63335000000001 -54.869499999999995, -67.56244 -54.87001, -66.95992000000001 -54.896810000000016, -67.291025
12 86 790 567 Africa Dem. Rep. Congo COD 50 400 POLYGON ((29.339997592900346 -4.4999834122940925, 29.519986606572928 -5.419978936386315, 29.419992710088167 -5.939998874539434, 29.620032179490014 -6.52001515058342
13 10 192 317,3 Africa Somalia SOM 4 719 POLYGON ((41.58513 -1.6832500000000001, 40.993 -0.85829, 40.98105 2.7845199999999997, 41.85508309264397 3.918911920483727, 42.128609999999995 4.23413, 42.76967 4.25259,
14 52 573 973 Africa Kenya KEN 95 503 POLYGON ((39.20222 -4.67677, 37.7669 -3.677120000000004, 37.69868999999994 -3.0969899999999484, 34.07261999999997 -1.0598199999999451, 33.90371119710453 -0.9500000000
15 42 813 238 Africa Sudan SDN 30 513 POLYGON ((24.567369012152085 8.229187933785468, 23.805813429466752 8.666318874542526, 23.459012892355986 8.954285793488893, 23.394779087017184 9.265067857292223, 23
16 15 946 876 Africa Chad TCD 11 314 POLYGON ((23.837660000000014 19.580470000000105, 23.886890000000108 15.610839999999996, 23.024590000000103 15.680720000000065, 22.567950000000011 14.944290000000137,
17 11 263 077 North America Haiti HTI 14 332 POLYGON ((-71.71236141629296 19.714455878167357, -71.62487321642283 19.169837958243306, -71.7013026597825 18.785416978424053, -71.94511206733556 18.616900132720 26,
18 10 738 958 North America Dominican Rep. DOM 88 941 POLYGON ((-71.70830481635805 18.044997056546094, -71.68773759630588 18.31666006110447, -71.94511206733556 18.616900132720 26, -71.7013026597825 18.785416978424053, -7
19 144 373 535 Europe Russia RUS 1 699 876 MULTIPOLYGON (((180.00000000000006 71.51571433642829, 180.00000000000006 70.83219920854673, 178.90342499999997 70.78114000000001, 178.7253 71.0988, 180.000000000000
20 389 482 North America Bahamas BHS 13 578 MULTIPOLYGON (((-78.98 26.790000000000003, -78.51 26.870000000000005, -77.85000000000001 26.84, -77.82000000000001 26.580000000000005, -78.91000000000001 26.42, -78.98
21 3 398 South America Falkland Is. FLK 282 POLYGON ((-61.2 -51.85, -60 -51.25, -59.15 -51.5, -58.550000000000004 -51.10000000000001, -57.75 -51.55, -58.050000000000004 -51.900000000000006, -59.400000000000006 -52.19
22 5 347 896 Europe Norway NOR 403 336 MULTIPOLYGON (((15.142820000000002 79.67431, 15.52255 80.01608, 16.990850000000002 80.05086, 18.25183 79.70175, 21.54383 78.95611, 19.02737 78.5626, 18.47172 77.82669, 17.
23 56 225 North America Greenland GRL 3 051 POLYGON ((-46.76379 82.62796, -43.40644 83.22516000000002, -39.89753 83.18018, -38.62214 83.54905, -35.08787 83.64513000000001, -27.10046 83.51966, -20.84539 82.72668999995
24 140 Seven seas (open ocean) Fr. S. Antarctic Lands ATF 16 POLYGON ((68.935 -48.62500000000001, 69.58 -48.940000000000005, 70.525 -49.065000000000005, 70.56 -49.255, 70.28 -49.71, 68.745 -49.775000000000006, 68.72000000000001 -49.
25 1 293 119 Asia Timor-Leste TLS 2 017 POLYGON ((124.96868248911623 -8.892790215697083, 125.08624637258026 -8.65688730228468, 125.94707238169826 -8.432094821815035, 126.64470421763855 -8.398246758663852,
26 58 558 270 Africa South Africa ZAF 351 431 POLYGON ((16.344976840895242 -28.576705010697 7, 16.824017368240902 -28.08216155366447, 17.218928663815404 -28.35594329194681, 17.387497185951503 -28.783514092727 8,
27 2 125 268 Africa Lesotho LSO 2 376 POLYGON ((28.978262566857243 -28.95559661226171, 29.32516645683259 -29.257386976846256, 29.018415154748027 -29.74376555757737, 28.84839969250774 -30.070050551068256
28 127 575 529 North America Mexico MEX 1 268 870 POLYGON ((-117.12775999999985 32.53533999999996, -115.99134999999995 32.61239000000012, -114.72138999999993 32.72082999999992, -114.815 32.52528000000001, -113.30498
29 3 461 734 South America Uruguay URY 56 045 POLYGON ((-57.62513342958296 -30.21629485445426, -56.976025763564735 -30.109686374636127, -55.97324459494094 -30.883075860316303, -55.601510179249345 -30.8538786760
30 211 049 527 South America Brazil BRA 1 839 758 POLYGON ((-53.373661668498244 -33.768377780900764, -53.6505439927181 -33.20200408298183, -53.209588995971544 -32.727666110974724, -53.78795162618219 -32.04724252698
31 11 513 100 South America Bolivia BOL 40 895 POLYGON ((-69.52967810736496 -10.951734307502194, -68.78615759954948 -11.03638030359628, -68.27125362819326 -11.014521172736 2, -68.04819230820539 -10.712059014532 4
32 32 510 453 South America Peru PER 226 848 POLYGON ((-69.89363521999663 -4.2981869441943275, -70.7947688463023 -4.251264743673303, -70.92884334988358 -4.401591485210368, -71.74840572781655 -4.593982842633011
Régénérer
Save
Cancel
Exporter les résultats ...
200
177
177 row(s) fetched - 0,016s (0,016s fetch), on 2024-06-05 at 21:37:04
Navigateur de ...
Projets
Enter a part of object name here
.db
Tables
Actualcountries
DataCleanAndPrepareForML
DimTablelanguages
DimTableOrganizations
GisDataWithImputationTable
WordGeoPandasBase
Views
Indexes
Sequences
Table Triggers
Data Types
DBeaver Sample Database (SQLite)
OneDrive
OneDrive 2
Project - General
Name
Bookmarks
Diagrams
Scripts
PST
fr
Insertion avancée
2 : 33 : 112
Sel: 0 | 0
Taper ici pour rechercher
66°F Clear
FRA
FR
21:38
05/06/2024

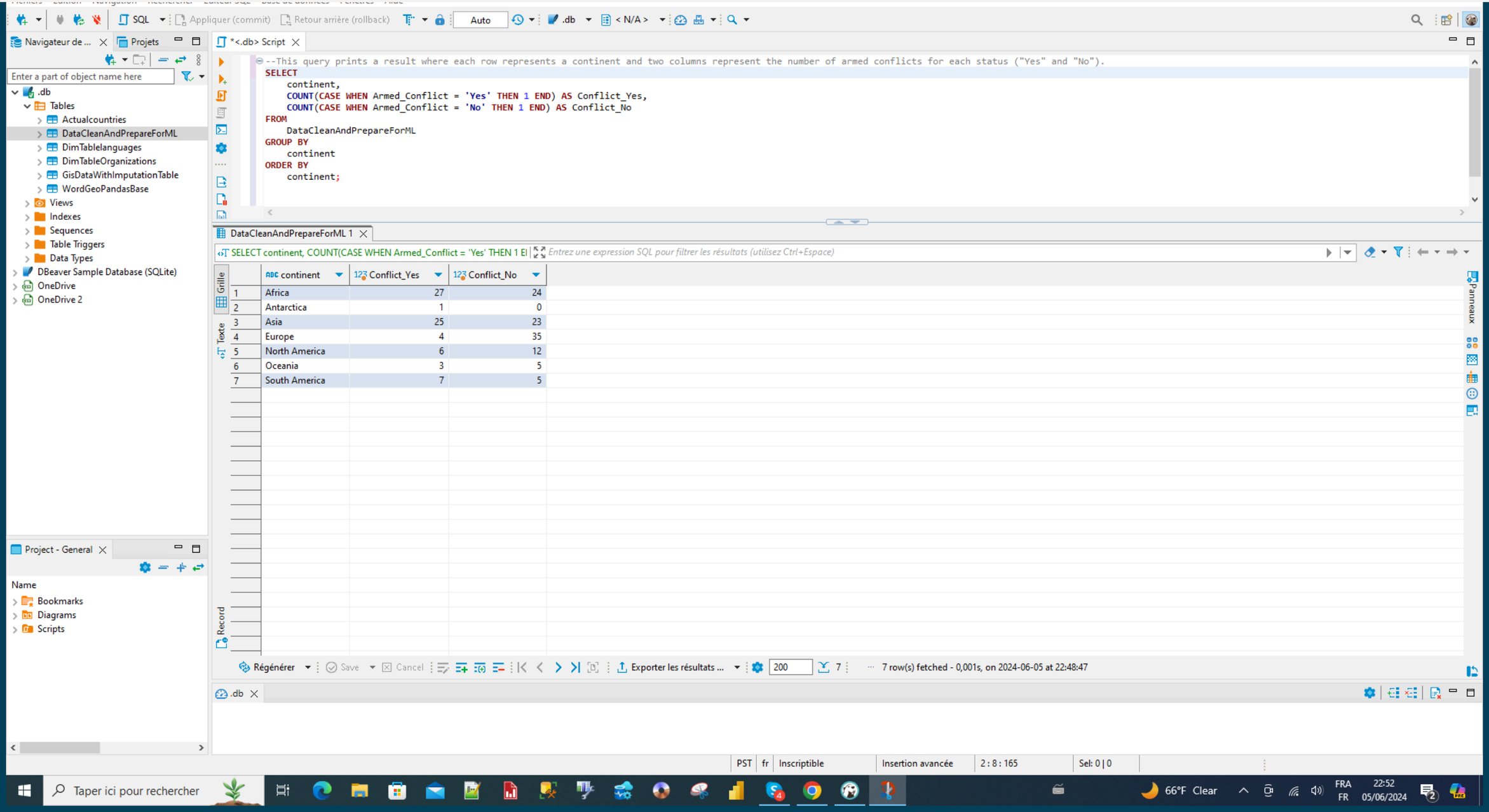


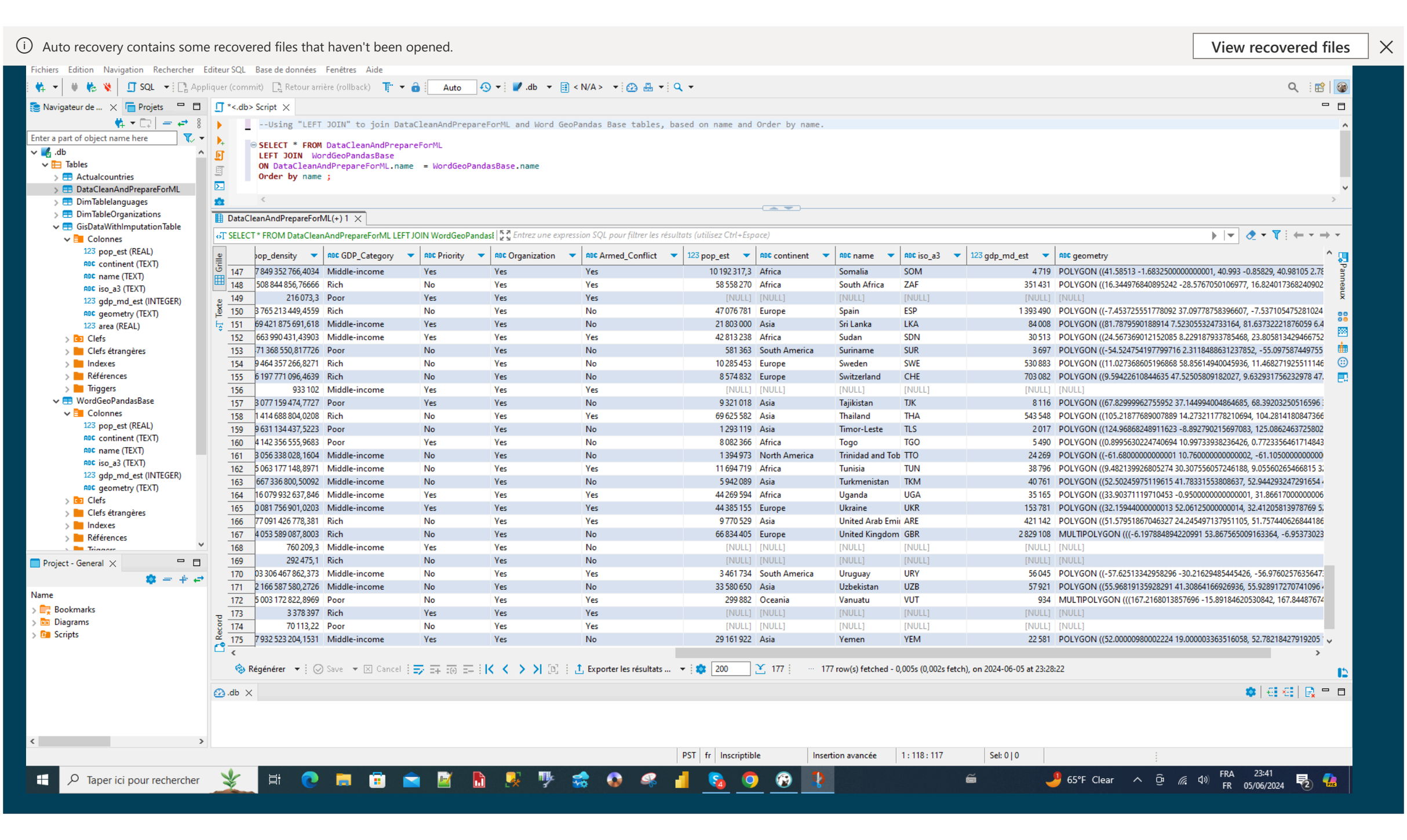
	name	continent	gdp_md_est	Armed_Conflict	Priority
1	Fiji	Oceania	5496	Yes	Yes
2	Papua New Guinea	Oceania	24829	Yes	Yes
3	Indonesia	Asia	1119190	Yes	Yes
4	Democratic Republic of Congo	Africa	50400	Yes	Yes
5	Somalia	Africa	4719	Yes	Yes
6	Kenya	Africa	95503	Yes	Yes
7	Sudan	Africa	30513	Yes	Yes
8	Chad	Africa	11314	Yes	Yes
9	Haiti	North America	14332	Yes	Yes
10	Lesotho	Africa	2376	Yes	Yes
11	Mauritania	Africa	7600	Yes	Yes
12	Nigeria	Africa	448120	Yes	Yes
13	Ghana	Africa	66983	Yes	Yes
14	Côte d'Ivoire	Africa	58539	Yes	Yes
15	Guinea	Africa	12296	Yes	Yes
16	Guinea-Bissau	Africa	1339	Yes	Yes
17	Liberia	Africa	3070	Yes	Yes
18	Sierra Leone	Africa	4121	Yes	Yes
19	Burkina Faso	Africa	15990	Yes	Yes
20	Congo	Africa	12267	Yes	Yes
21	Iraq	Asia	234094	Yes	Yes
22	Cambodia	Asia	27089	Yes	Yes
23	Lao People's Democratic Republic	Asia	18173	Yes	Yes
24	Viet Nam	Asia	261921	Yes	Yes
25	Pakistan	Asia	278221	Yes	Yes
26	Afghanistan	Asia	19291	Yes	Yes
27	Syrian Arab Republic	Asia	98830	Yes	Yes
28	Ukraine	Europe	153781	Yes	Yes
29	Philippines	Asia	376795	Yes	Yes
30	Ethiopia	Africa	95912	Yes	Yes
31	Somalia	Africa	17836	Yes	Yes
32	Uganda	Africa	35165	Yes	Yes

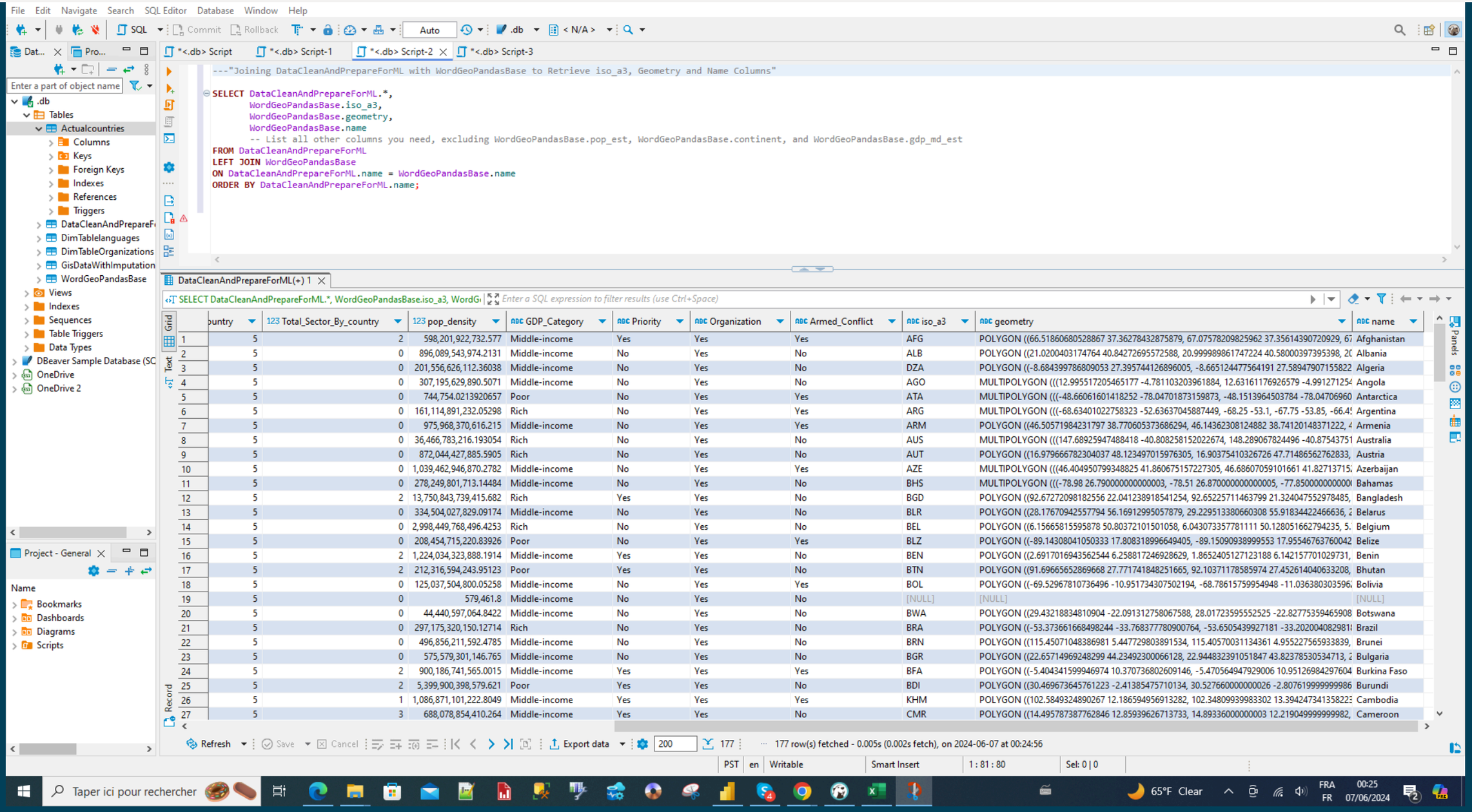

Auto recovery contains some recovered files that haven't been opened.
View recovered files
Navigateur de ...
Projets
Enter a part of object name here
.db
Tables
Actualcountries
DataCleanAndPrepareForML
DimTablelanguages
DimTableOrganizations
GisDataWithImputationTable
WordGeoPandasBase
Views
Indexes
Sequences
Table Triggers
Data Types
DBeaver Sample Database (SQLite)
OneDrive
OneDrive 2
*<.db> Script
--This SQL query selects specific columns (name, continent, gdp_md_est, GDP_Category, Armed_Conflict, and Priority) from the DataCleanAndPrepareForML table.
-- It filters the rows to include only those where the Priority and Armed_Conflict columns are both set to "Yes" and the GDP_Category column is set to 'Poor'.
SELECT name, continent, gdp_md_est, GDP_Category, Armed_Conflict, Priority
FROM DataCleanAndPrepareForML
WHERE Priority = "Yes" AND Armed_Conflict = "Yes" AND GDP_Category = 'Poor';
DataCleanAndPrepareForML 1
SELECT name, continent, gdp_md_est, GDP_Category, Armed_Conflict, P
Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)
name continent gdp_md_est GDP_Category Armed_Conflict Priority
1 Fiji Oceania 5496 Poor Yes Yes
2 Somalia Africa 4719 Poor Yes Yes
3 Chad Africa 11314 Poor Yes Yes
4 Lesotho Africa 2376 Poor Yes Yes
5 Mauritania Africa 7600 Poor Yes Yes
6 Guinea Africa 12296 Poor Yes Yes
7 Guinea-Bissau Africa 1339 Poor Yes Yes
8 Liberia Africa 3070 Poor Yes Yes
9 Sierra Leone Africa 4121 Poor Yes Yes
10 Congo Africa 12267 Poor Yes Yes
11 South Sudan Africa 11998 Poor Yes Yes
Grille
Texte
Record
Panneaux
Régénérer
Save
Cancel
Exporter les résultats ...
200
11
11 row(s) fetched - 0,000s, on 2024-06-05 at 22:09:32
Project - General
Name
Bookmarks
Diagrams
Scripts
.db
PST
fr
Inscriptible
Insertion avancée
3 : 1 : 318
Sel: 0 | 0
Taper ici pour rechercher
66°F Clear
FRA
FR
22:12
05/06/2024

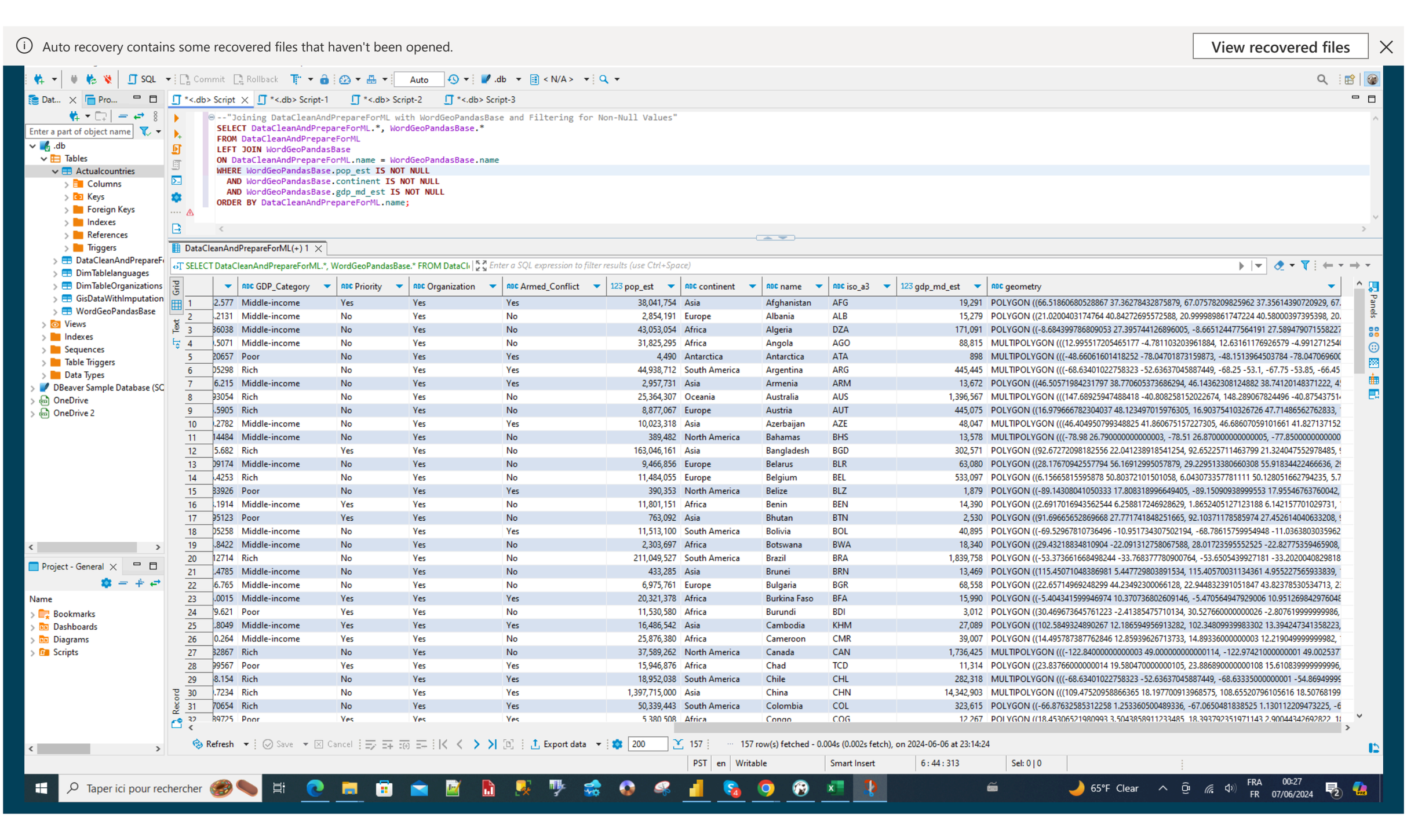



Auto recovery contains some recovered files that haven't been opened.
View recovered files
SQL
Appliquer (commit)
Retour arrière (rollback)
Auto
.db
< N/A >
Navigateur de ...
Projets
Enter a part of object name here
.db
Tables
Actualcountries
DataCleanAndPrepareForML
DimTablelanguages
DimTableOrganizations
GisDataWithImputationTable
WordGeoPandasBase
Views
Indexes
Sequences
Table Triggers
Data Types
DBeaver Sample Database (SQLite)
OneDrive
OneDrive 2
*<.db> Script
--This query prints a list of continents and the total number of armed conflicts (where Armed_Conflict is "Yes") for each, sorted from highest to lowest number of conflicts.
SELECT continent,
COUNT(CASE WHEN Armed_Conflict = 'Yes' THEN 1 END) AS TotalConflicts
FROM DataCleanAndPrepareForML
GROUP BY continent
ORDER BY TotalConflicts DESC;
DataCleanAndPrepareForML 1
SELECT continent, COUNT(CASE WHEN Armed_Conflict = 'Yes' THEN 1 EI
Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)
Grille
Texte
Record
Panneaux
continent
TotalConflicts
1 Africa 27
2 Asia 25
3 South America 7
4 North America 6
5 Europe 4
6 Oceania 3
7 Antarctica 1
Régénérer
Save
Cancel
Exporter les résultats ...
200
7 row(s) fetched - 0,001s, on 2024-06-05 at 22:39:38
Project - General
Name
Bookmarks
Diagrams
Scripts
PST
fr
Inscriptible
Insertion avancée
1 : 174 : 173
Sel: 0 | 0
Taper ici pour rechercher
66°F Clear
FRA
FR
22:44
05/06/2024




Auto recovery contains some recovered files that haven't been opened.
View recovered files
Fichiers Edition Navigation Rechercher Editeur SQL Base de données Fenêtres Aide
Navigateur de ... Projets
Enter a part of object name here
.db
Tables
Actualcountries
DataCleanAndPrepareForML
DimTablelanguages
DimTableOrganizations
GisDataWithImputationTable
Colonnes
pop_est (REAL)
continent (TEXT)
name (TEXT)
iso_a3 (TEXT)
gdp_md_est (INTEGER)
geometry (TEXT)
area (REAL)
Clefs
Clefs étrangères
Indexes
Références
Triggers
WordGeoPandasBase
Colonnes
pop_est (REAL)
continent (TEXT)
name (TEXT)
iso_a3 (TEXT)
gdp_md_est (INTEGER)
geometry (TEXT)
Clefs
Clefs étrangères
Indexes
Références
Project - General
Name
Bookmarks
Diagrams
Scripts
*<.db> Script
--Using "LEFT JOIN" to join DataCleanAndPrepareForML and Word GeoPandas Base tables, based on name and Order by name.
SELECT * FROM DataCleanAndPrepareForML
LEFT JOIN WordGeoPandasBase
ON DataCleanAndPrepareForML.name = WordGeoPandasBase.name
Order by name ;
DataCleanAndPrepareForML(+) 1
SELECT * FROM DataCleanAndPrepareForML LEFT JOIN WordGeoPandasE
Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)
Grille
Texte
Record
Panneaux
pop_density | GDP_Category | Priority | Organization | Armed_Conflict | pop_est | continent | name | iso_a3 | gdp_md_est | geometry
147 | 7 849 352 766,4034 | Middle-income | Yes | Yes | Yes | 10 192 317,3 | Africa | Somalia | SOM | 4 719 | POLYGON ((41.58513 -1.6832500000000001, 40.993 -0.85829, 40.98105 2.78
148 | 508 844 856,76666 | Rich | No | Yes | Yes | 58 558 270 | Africa | South Africa | ZAF | 351 431 | POLYGON ((16.344976840895242 -28.5767050106977, 16.824017368240902
149 | 216 073,3 | Poor | Yes | Yes | Yes | [NULL] | [NULL] | [NULL] | [NULL] | [NULL] | [NULL]
150 | 3 765 213 449,4559 | Rich | No | Yes | No | 47 076 781 | Europe | Spain | ESP | 1 393 490 | POLYGON ((-7.453725551778092 37.09778758396607, -7.537105475281024
151 | 69 421 875 691,618 | Middle-income | Yes | Yes | No | 21 803 000 | Asia | Sri Lanka | LKA | 84 008 | POLYGON ((81.78795901889 14 7.523055324733164, 81.63732221876059 6.4
152 | 663 990 431,43903 | Middle-income | Yes | Yes | Yes | 42 813 238 | Africa | Sudan | SDN | 30 513 | POLYGON ((24.567369012152085 8.229187933785468, 23.805813429466752
153 | 71 368 550,817726 | Poor | No | Yes | No | 581 363 | South America | Suriname | SUR | 3 697 | POLYGON ((-54.524754197799716 2.3118488631237852, -55.097587449755
154 | 9 464 357 266,8271 | Rich | No | Yes | No | 10 285 453 | Europe | Sweden | SWE | 530 883 | POLYGON ((11.027368605196868 58.85614940045936, 11.468271925511146
155 | 6 197 771 096,4639 | Rich | No | Yes | No | 8 574 832 | Europe | Switzerland | CHE | 703 082 | POLYGON ((9.59422610844635 47.52505809182027, 9.632931756232978 47.
156 | 933 102 | Middle-income | Yes | Yes | Yes | [NULL] | [NULL] | [NULL] | [NULL] | [NULL] | [NULL]
157 | 3 077 159 474,7727 | Poor | Yes | Yes | No | 9 321 018 | Asia | Tajikistan | TJK | 8 116 | POLYGON ((67.82999962755952 37.144994004864685, 68.39203250516596
158 | 1 414 688 804,0208 | Rich | No | Yes | Yes | 69 625 582 | Asia | Thailand | THA | 543 548 | POLYGON ((105.21877689007889 14.273211778210694, 104.2814180847366
159 | 9 631 134 437,5223 | Poor | No | Yes | No | 1 293 119 | Asia | Timor-Leste | TLS | 2 017 | POLYGON ((124.96868248911623 -8.892790215697083, 125.0862463725802
160 | 4 142 356 555,9683 | Poor | Yes | Yes | No | 8 082 366 | Africa | Togo | TGO | 5 490 | POLYGON ((0.8995630224740694 10.99733938236426, 0.7723356461714843
161 | 3 056 338 028,1604 | Middle-income | No | Yes | No | 1 394 973 | North America | Trinidad and Tob | TTO | 24 269 | POLYGON ((-61.68000000000001 10.760000000000002, -61.1050000000000
162 | 5 063 177 148,8971 | Middle-income | No | Yes | Yes | 11 694 719 | Africa | Tunisia | TUN | 38 796 | POLYGON ((9.482139926805274 30.307556057246188, 9.05560265466815 3
163 | 667 336 800,50092 | Middle-income | No | Yes | No | 5 942 089 | Asia | Turkmenistan | TKM | 40 761 | POLYGON ((52.50245975119615 41.78331553808637, 52.944293247291654
164 | 16 079 932 637,846 | Middle-income | Yes | Yes | Yes | 44 269 594 | Africa | Uganda | UGA | 35 165 | POLYGON ((33.90371119710453 -0.9500000000000001, 31.866170000000006
165 | 0 081 756 901,0203 | Middle-income | Yes | Yes | Yes | 44 385 155 | Europe | Ukraine | UKR | 153 781 | POLYGON ((32.15944000000013 52.06125000000014, 32.412058139787695
166 | 77 091 426 778,381 | Rich | No | Yes | Yes | 9 770 529 | Asia | United Arab Emir | ARE | 421 142 | POLYGON ((51.57951867046327 24.245497137951105, 51.757440626844186
167 | 4 053 589 087,8003 | Rich | No | Yes | No | 66 834 405 | Europe | United Kingdom | GBR | 2 829 108 | MULTIPOLYGON (((-6.197884894220991 53.867565009163364, -6.95373023
168 | 760 209,3 | Middle-income | Yes | Yes | No | [NULL] | [NULL] | [NULL] | [NULL] | [NULL] | [NULL]
169 | 292 475,1 | Rich | No | Yes | No | [NULL] | [NULL] | [NULL] | [NULL] | [NULL] | [NULL]
170 | 03 306 467 862,373 | Middle-income | No | Yes | Yes | 3 461 734 | South America | Uruguay | URY | 56 045 | POLYGON ((-57.62513342958296 -30.21629485445426, -56.9760257635647
171 | 2 166 587 580,2726 | Middle-income | No | Yes | No | 33 580 650 | Asia | Uzbekistan | UZB | 57 921 | POLYGON ((55.96819135928291 41.30864166926936, 55.928917270741096
172 | 5 003 172 822,8969 | Poor | No | Yes | Yes | 299 882 | Oceania | Vanuatu | VUT | 934 | MULTIPOLYGON (((167.2168013857696 -15.89184620530842, 167.84487674
173 | 3 378 397 | Rich | Yes | Yes | Yes | [NULL] | [NULL] | [NULL] | [NULL] | [NULL] | [NULL]
174 | 70 113,22 | Poor | No | Yes | Yes | [NULL] | [NULL] | [NULL] | [NULL] | [NULL] | [NULL]
175 | 7 932 523 204,1531 | Middle-income | Yes | Yes | No | 29 161 922 | Asia | Yemen | YEM | 22 581 | POLYGON ((52.00000980002224 19.000003363516058, 52.78218427919205
Régénérer Save Cancel Exporter les résultats ... 200 177 177 row(s) fetched - 0,005s (0,002s fetch), on 2024-06-05 at 23:28:22
.db
PST fr Inscriptible Insertion avancée 1 : 118 : 117 Sel: 0 | 0
Taper ici pour rechercher
65°F Clear
FRA FR 23:41 05/06/2024



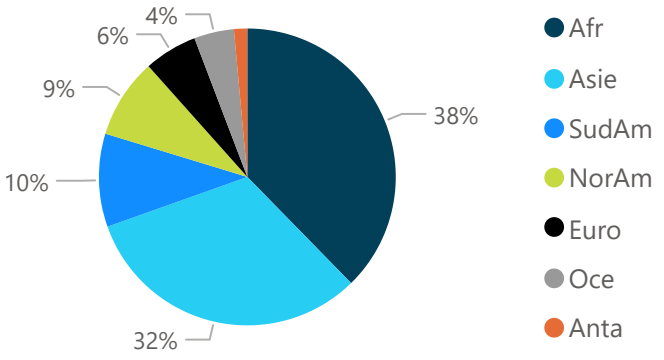

Auto recovery contains some recovered files that haven't been opened.
View recovered files
*<.db> Script
*<.db> Script-1
*<.db> Script-2
*<.db> Script-3
--"Joining DataCleanAndPrepareForML with WordGeoPandasBase and Filtering for Non-Null Values"
SELECT DataCleanAndPrepareForML.*, WordGeoPandasBase.*
FROM DataCleanAndPrepareForML
LEFT JOIN WordGeoPandasBase
ON DataCleanAndPrepareForML.name = WordGeoPandasBase.name
WHERE WordGeoPandasBase.pop_est IS NOT NULL
AND WordGeoPandasBase.continent IS NOT NULL
AND WordGeoPandasBase.gdp_md_est IS NOT NULL
ORDER BY DataCleanAndPrepareForML.name;
DataCleanAndPrepareForML(+) 1
Refresh
Save
Cancel
Export data
157 row(s) fetched - 0.004s (0.002s fetch), on 2024-06-06 at 23:14:24
Taper ici pour rechercher
65°F Clear
FRA
FR
00:27
07/06/2024

Humanitarian Response Dashboard

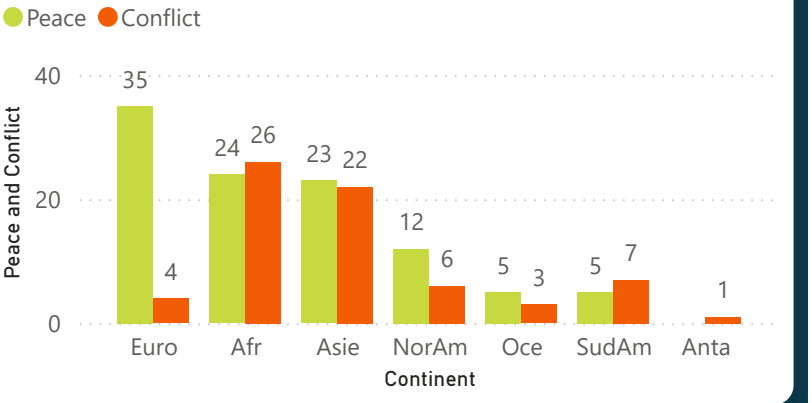
Countries	Gdp	Pop	Continent	Area	Organ	Sector
173	86M	7.60bn	7	214.05M	841	131

Conflicts: All
GDP: All
Countries: All
Continent: All
Sectors: All
Priorities: All

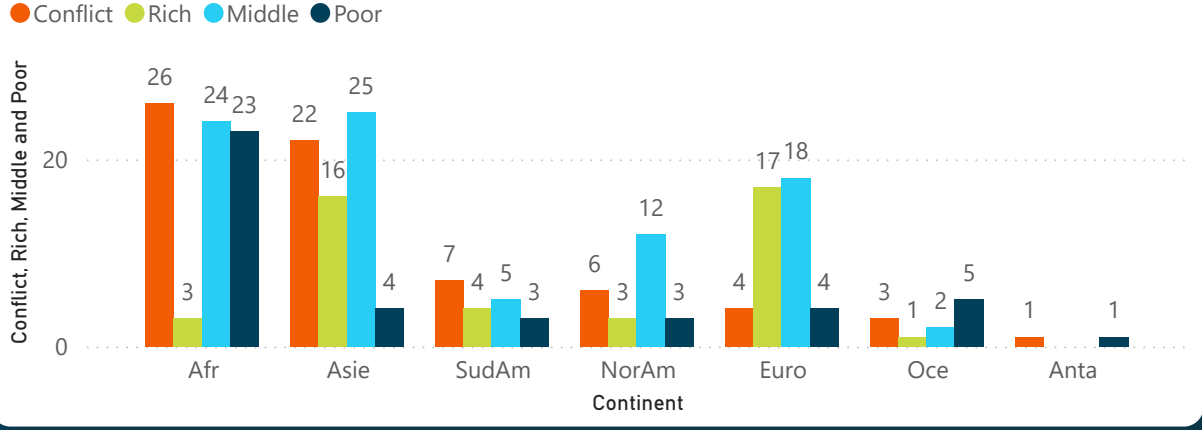
Ongoing Armed Conflicts by Continent



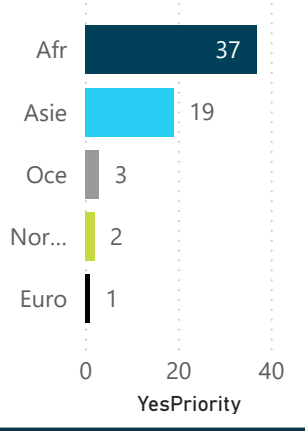
Conflict and Peace by Continent



Conflicts and Development by Continent



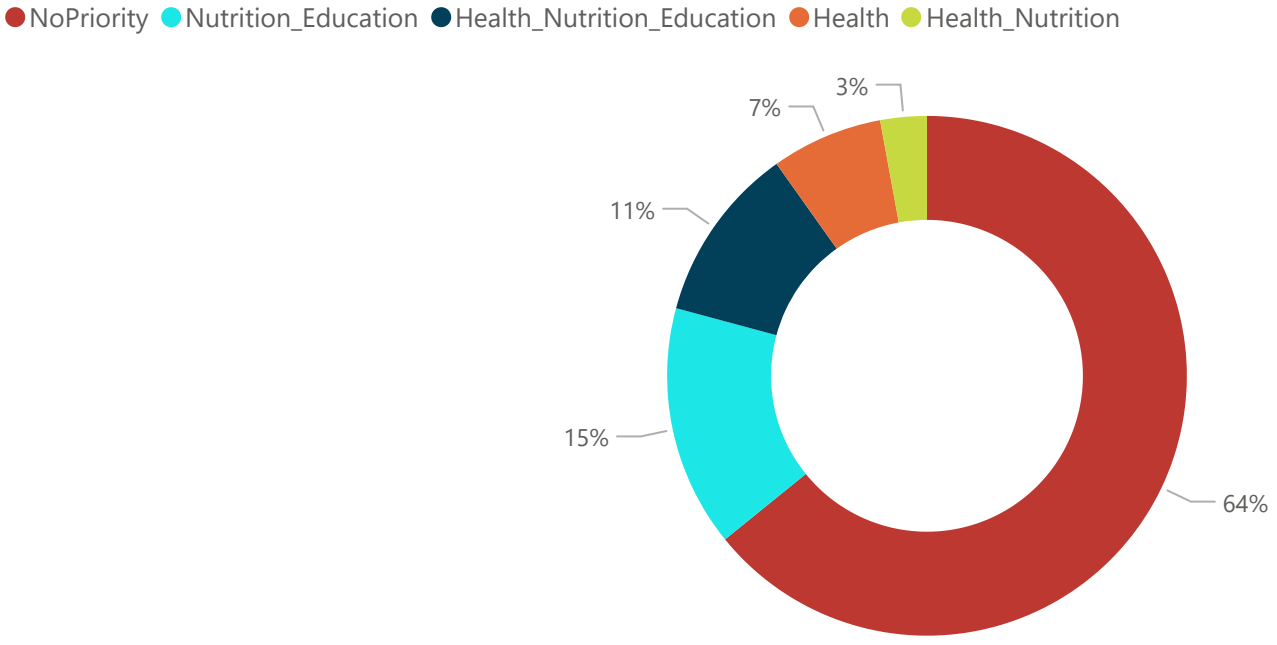
Total Of Priorities



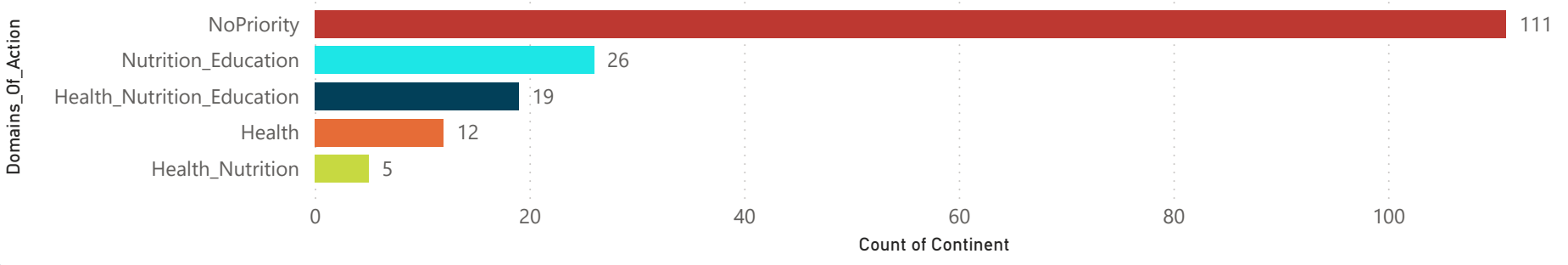
All

All

Total areas of intervention in %

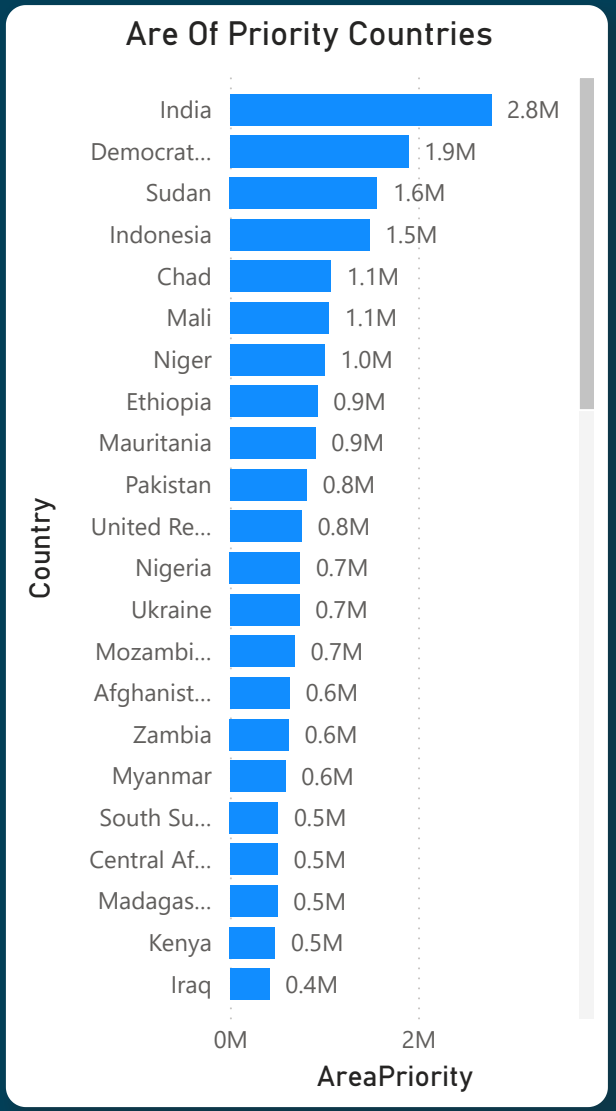


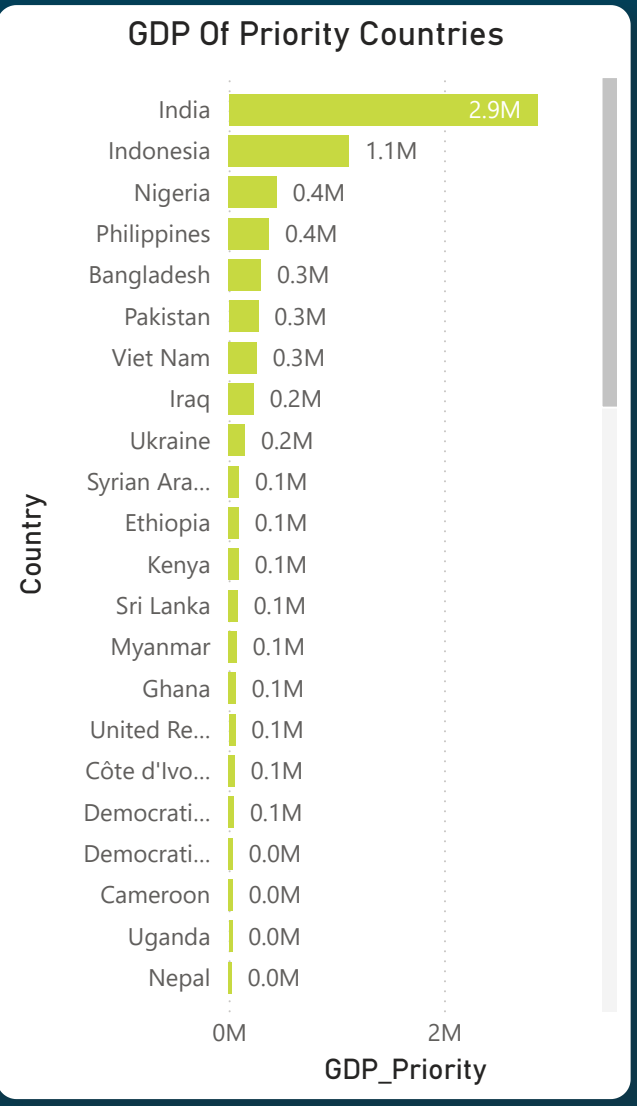
Frequency Of Action Areas



Continent

All





All

All

Conflict Zone