

Weather Data ETL Project

Diego Iván Perea Montealegre diego.perea@uao.edu.co
Facultad de Ingeniería, Universidad Autónoma de Occidente
Cali, Valle del Cauca

Introduction

The analysis of meteorological data is essential for understanding climate patterns, predicting weather conditions, and studying the impact of climate on various areas such as agriculture, transportation, and disaster management. This synthetic dataset provides a valuable opportunity to research and develop models without the restrictions of accessing real data, facilitating experimentation and the improvement of analytical techniques.

The dataset **Weather Data** of kaggle contains synthetic weather data generated for ten different locations, including New York, Los Angeles, Chicago, Houston, Phoenix, Philadelphia, San Antonio, San Diego, Dallas, and San Jose. The data includes information about temperature, humidity, precipitation, and wind speed, with 1 million data points generated for each parameter.

Features:

- **Location:** The city where the weather data was simulated.
- **Date_Time:** The date and time when the weather data was recorded.
- **Temperature_C:** The temperature in Celsius at the given location and time.
- **Humidity_pct:** The humidity in percentage at the given location and time.
- **Precipitation_mm:** The precipitation in millimeters at the given location and time.
- **Wind_Speed_kmh:** The wind speed in kilometers per hour at the given location and time.

Additional Information:

- **Variability and Complexity:** The dataset incorporates variability and complexity to simulate realistic weather patterns. For example, adjustments have been made to temperature and precipitation based on seasonal variations observed in certain locations. In New York, higher temperatures and precipitation are simulated during the summer months, while in Phoenix, lower temperatures and increased precipitation are simulated during the winter months.

Importance and Applications

1. **Development of Weather Prediction Models:** With parameter-based data, this dataset allows researchers and data scientists to train weather prediction models that can improve accuracy in estimating temperature, humidity, precipitation, and wind speed across different cities.
2. **Climate Studies and Trend Analysis:** By providing simulated weather data from various locations, this dataset enables the analysis of specific climate patterns in each region, the identification of long-term trends, and the evaluation of seasonal variations.
3. **Educational and Academic Use:** This dataset serves as an excellent tool for teaching data analysis, visualization, and modeling in meteorology. Students and educators can use it in courses related to data science, artificial intelligence, and applied statistics.
4. **Optimization in Climate-Dependent Sectors:** Industries such as agriculture, aviation, and logistics can utilize this data to simulate scenarios and develop adaptation strategies for

climate variations, contributing to better planning and decision-making.

5. **Evaluation of Machine Learning Algorithms:** Thanks to the dataset's quantity and diversity, different machine learning approaches can be tested and compared for weather prediction, allowing for model validation before applying them to real-world data.

This synthetic dataset represents a versatile and valuable tool for research, education, and the practical application of climate analysis and prediction techniques. Its availability facilitates the development of innovative solutions without relying on access restrictions to real meteorological data.

Objective

The ETL (Extract, Transform, Load) process for visualizing this synthetic weather dataset would primarily aim to efficiently prepare the data to generate charts and visual analyses that help identify climate patterns, trends, and correlations between variables.

Specific ETL Objectives:

Extraction (Extract):

- Retrieve data from the original Kaggle dataset, ensuring data integrity.
- Load the data into an analysis environment such as Python (using Pandas), SQL, or a visualization tool like Power BI or Tableau.

Transformation (Transform):

- Data Cleaning:
 - Remove null or outlier values that could affect visualization accuracy.
 - Verify the consistency of measurement units (temperature in °C, wind speed in km/h, etc.).
- Format Conversion:
 - Convert the Date_Time column into the appropriate date and time format for temporal analysis.
- Data Aggregation:
 - Group data by city to calculate daily, weekly, or monthly average values for temperature, humidity, precipitation, and wind speed.
- Generation of New Metrics:
 - Calculate moving averages or variation rates to observe climate trends.

Final Project Delivery Steps

1. Data Sources: Select one or more data sources (e.g., CSVs, APIs, databases).
2. Data Extraction: Use Python to extract data from the selected source and store it in a relational database.
3. Exploratory Data Analysis (EDA): Analyze the raw data to understand its structure and quality.
4. Read Raw Data: Retrieve the raw data from the staging area database using Python.
5. Data Transformation: Perform necessary transformations to create value and address the problem.
6. Merge Data (if needed): Combine different data sources through a merge task.
7. Load Processed Data: Store the transformed dataset back into the database.
8. Dashboard Creation: Retrieve data from the ETL pipeline database and create a

dashboard using your preferred tool (e.g., Power BI, Looker Studio)

Steps Project

- **Phase1** : Identification of the data problem or objective and dataset selection ,Data extraction or collection
- **Phase2** : Data transformation , Data pre-analysis and visualization (EDA)
- **Phase3**: Data load in a SQL database , Presentation and Data story telling

Phase 1

The first phase involves extracting data from a source, in this case a .csv file, using Python and uploading the information to a database, which in this case is Supabase.

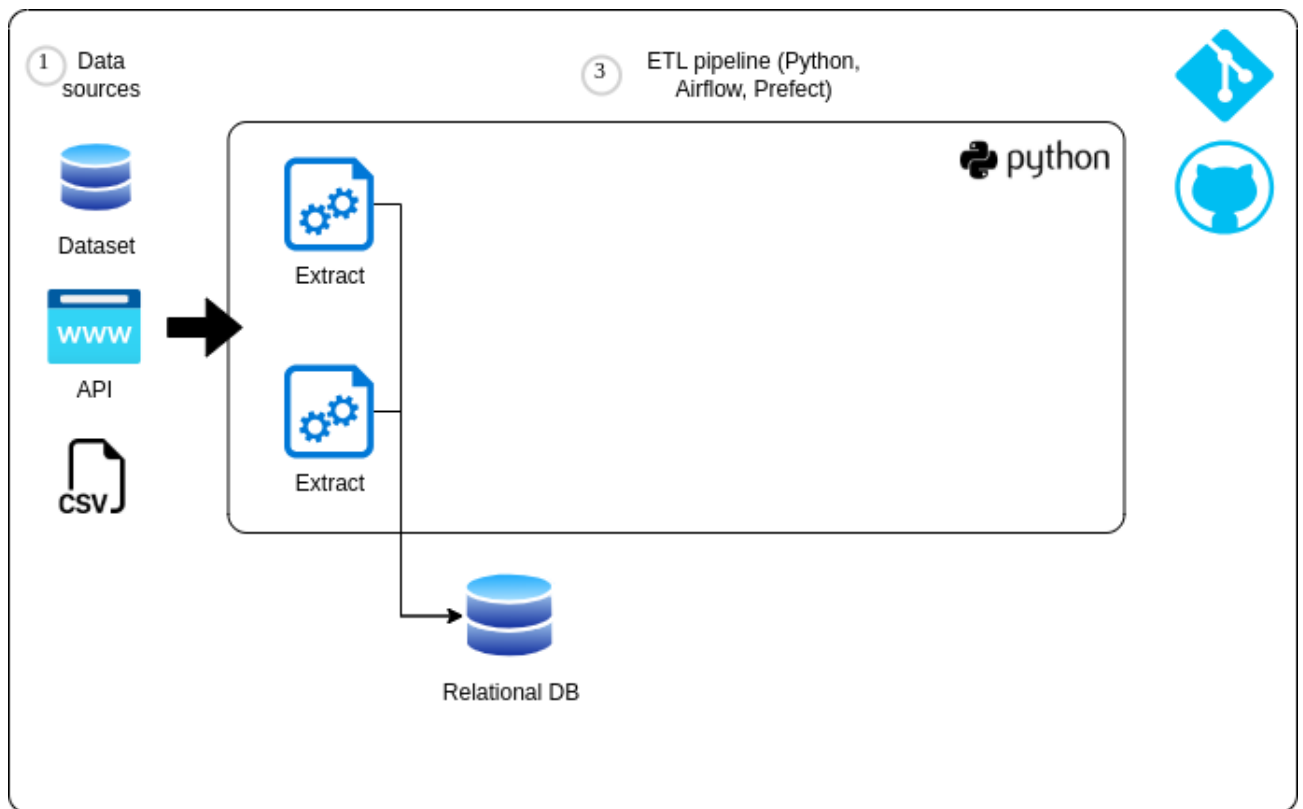
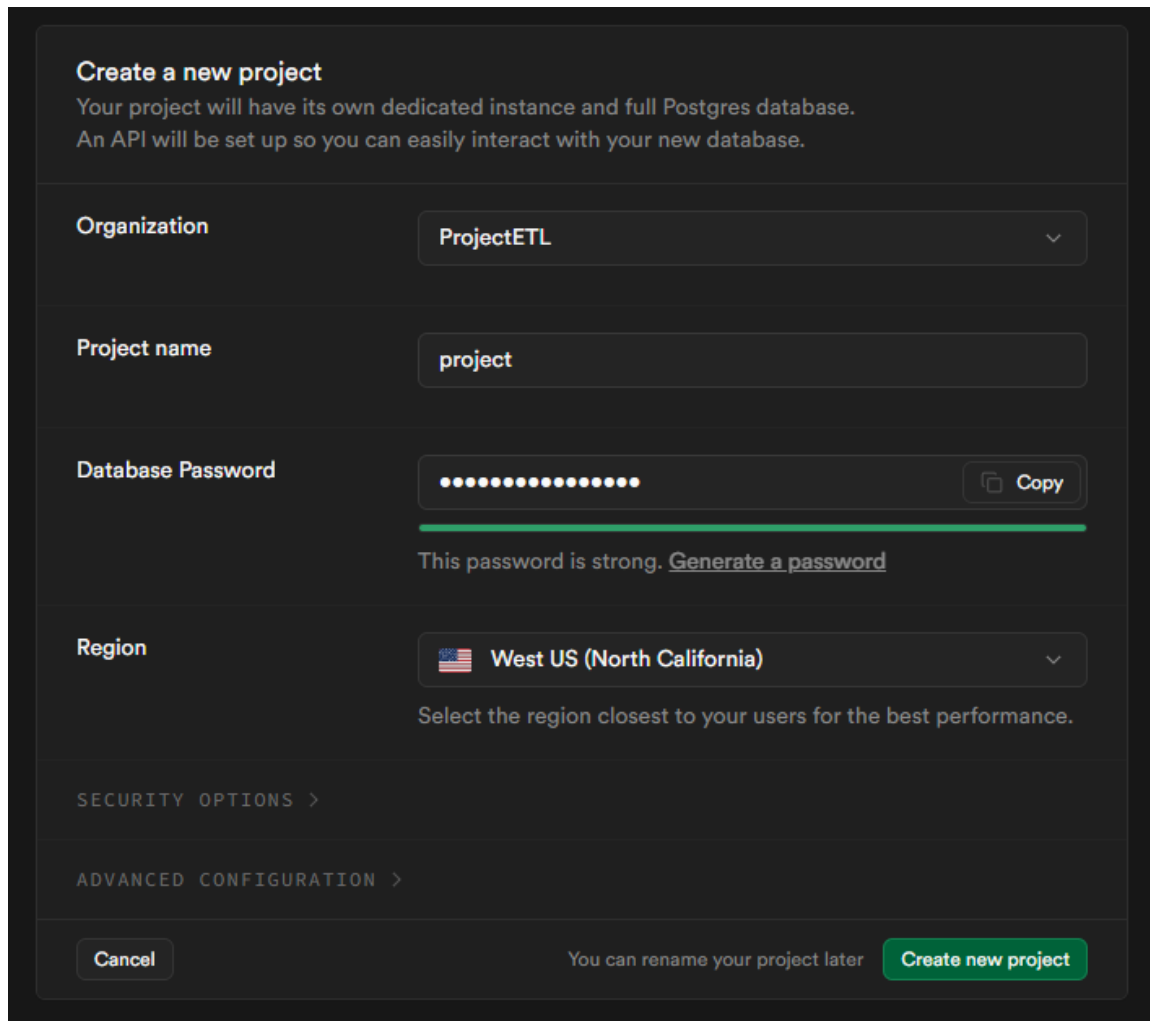


Figure 1. Diagram of Phase 1

Steps for develop Phase 1:

1. Create a database in Supabase. Supabase was selected because is an excellent choice for database management due to its ease of use, seamless setup, and cost-effectiveness. Unlike traditional databases that require complex installations and configurations, Supabase provides a fully managed, serverless PostgreSQL solution that can be accessed directly from the browser or through APIs. This eliminates the need for intricate setup processes, making it ideal for developers who want a hassle-free experience. Additionally, Supabase offers a generous free tier, allowing users to build and deploy applications without upfront costs. Its intuitive interface, real-time capabilities, and built-in authentication features further enhance its appeal, making it a powerful yet accessible choice for modern web and mobile applications.



Create a new project
Your project will have its own dedicated instance and full Postgres database.
An API will be set up so you can easily interact with your new database.

Organization ProjectETL

Project name project

Database Password Copy

This password is strong. [Generate a password](#)

Region West US (North California)

Select the region closest to your users for the best performance.

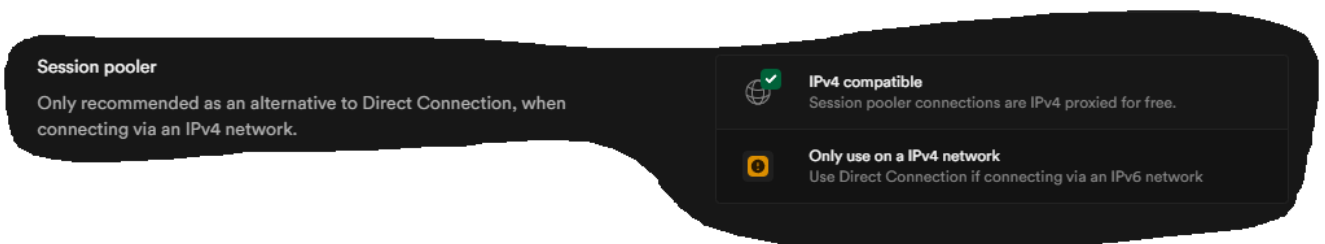
[SECURITY OPTIONS >](#)

[ADVANCED CONFIGURATION >](#)

Cancel You can rename your project later Create new project

Figure 2. SQL Database Creation in Supabase

2. Connect the Supabase database and choose the connection, in this case, "Session pooler"

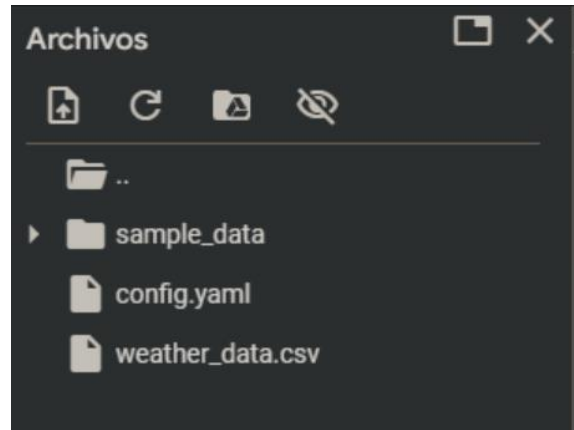


Session pooler
Only recommended as an alternative to Direct Connection, when connecting via an IPv4 network.

- ☒ **IPv4 compatible**
Session pooler connections are IPv4 proxied for free.
- ☐ **Only use on a IPv4 network**
Use Direct Connection if connecting via an IPv6 network

Figure 3. Connection with Session pooler

3. Upload data from the .csv to the database by following the file Phase1Load\001_loadPhase1.ipynb
Download the **weather_data.csv** file [1], create the **config.yaml** file following the steps, and upload it to Google Colab **.ipynb**.



4. View the uploaded data in the Supabase database. The next image shows evidence of the successful upload of the 11000 records handled in the previous step.

id	Location	Date_Time	Temperature_C	Humidity_pct	Precipitation_mm	Wind_Speed_kmh
1	San Diego	2024-01-14 21:12:46	10.6830010947154	41.1967535669445	4.0201871570867	8.23354024687302
2	San Diego	2024-05-17 15:22:10	8.7541397823536	58.3191073955202	9.1162344822938	27.7161612568925
3	San Diego	2024-05-11 09:30:59	11.6304363129309	38.820752691595	4.60751137714603	28.7329512882362
4	Philadelphia	2024-02-26 17:32:39	-6.62897589569391	54.0744739759457	3.18371974780765	26.3673026725366
5	San Antonio	2024-04-29 13:23:51	39.8082129746316	72.8999079529431	9.59828213674966	29.8986216692961
6	San Diego	2024-01-21 08:54:56	27.341054869124	49.0332360683476	9.16654330273274	27.4738960845188
7	San Jose	2024-01-13 02:10:54	1.88868336796818	65.7423245369102	0.22170904319721	1.073117812286
8	New York	2024-01-25 19:04:34	-6.89476554095456	30.8048940082328	6.02762351418163	16.8483368651636
9	New York	2024-03-29 05:20:30	0.963544894354438	38.8191676409565	3.64012916874015	7.98902397469938
10	San Jose	2024-05-18 09:14:02	-1.60708803842552	82.1987013132337	4.10149297151851	25.6472820704365
11	New York	2024-03-04 13:47:15	35.145559065071	64.7528656584597	8.34919496614864	25.4303100195307
12	Houston	2024-03-07 22:03:36	15.8167635681006	80.1199020345908	3.76000418615803	16.7821322062365
13	Dallas	2024-02-27 21:07:10	32.0168976079419	63.1943706422686	3.58267105442589	3.0501963856425
14	Houston	2024-05-09 00:53:10	38.6412692640686	85.957725559403	0.470781842603626	20.7792640109
15	Houston	2024-05-12 15:47:55	39.6667799833673	72.7470258115083	1.26372215889168	6.47949209737064
16	Philadelphia	2024-03-09 01:51:24	28.290116682928	35.2391696910979	9.34720473250742	14.0667646401653
17	San Antonio	2024-02-10 15:05:28	16.3497895003687	65.8126073101361	0.109090217726223	6.59703303956028
18	Chicago	2024-01-06 02:59:46	26.7368110913922	31.5136144229176	0.496024001203168	22.9800953046045
19	San Antonio	2024-05-08 16:20:53	35.0795476379057	35.0830709963318	9.59729410606991	4.50786270579149
20	San Diego	2024-01-31 06:38:46	14.6058192276471	66.6422351815403	1.516637132483	29.4318902008544
21	San Diego	2024-01-25 12:59:32	33.023390764717	62.6074849485407	0.212142687171479	16.7333348913416
22	New York	2024-02-19 12:16:07	-7.38381096899866	54.0899734429906	1.90573059680887	6.63706435992247

Figure 4. Uploaded Data in supabase database

Phase 2

The second phase performs the transformation and merging of the dataset uploaded to the DB.

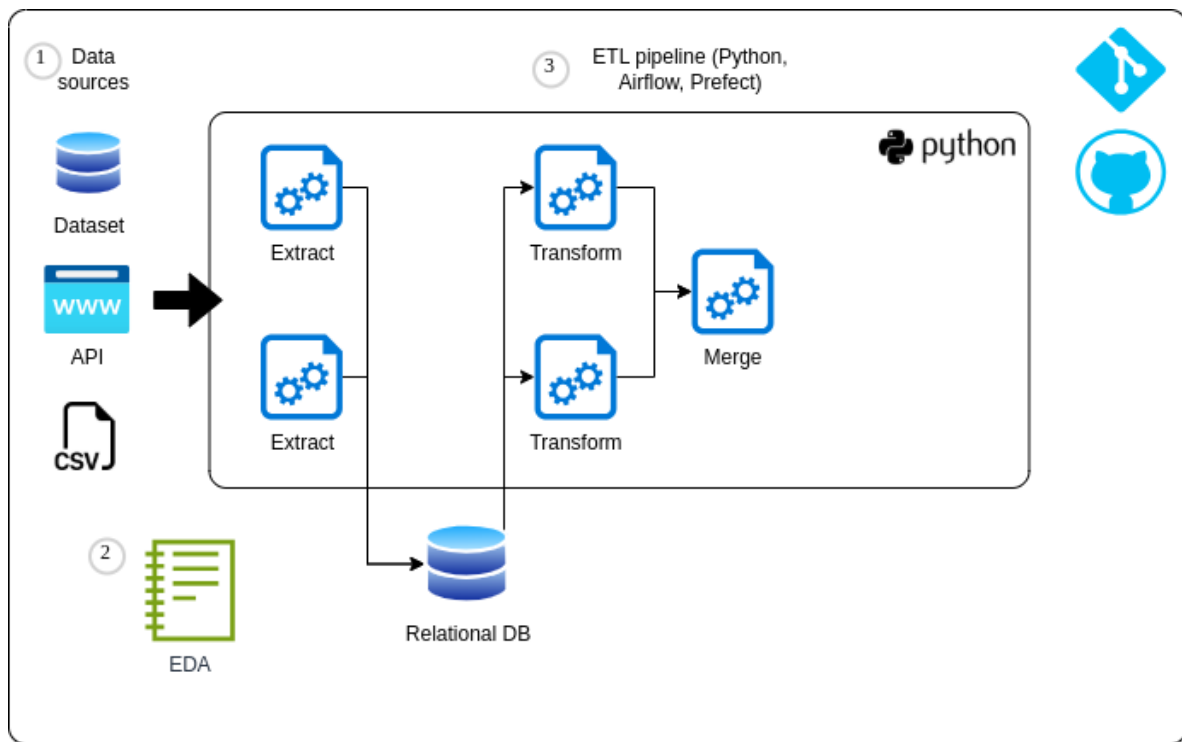


Figure 5. Diagram of Phase 2

The following transformations are performed:

- * Extract year, month, day, hour, and minute from Date_Time
- * Convert Location to uppercase
- * Normalize Temperature_C to Fahrenheit
- * Normalize Wind_Speed_kmh to mph
- * Average temperature per location

Each of these transformations enhances the dataset's usability and improves its relevance for:

Extracting Year, Month, Day, Hour, and Minute from Date_Time

- Reason: Breaking down timestamps into separate components allows for more granular time-based analysis. For example, it enables trend analysis by day, month, or year, which is useful for identifying seasonal patterns or hourly variations in weather conditions.
- Value: This facilitates time-series analysis, forecasting, and comparisons over different time scales.

Converting Location to Uppercase

- Reason: Standardizing location names eliminates inconsistencies caused by different letter cases (e.g., "San Diego" vs. "san diego"), ensuring that the same location is not treated as different entries.
- Value: Improves data integrity, prevents duplicate entries, and ensures consistency in filtering, grouping, and merging data.

Normalizing Temperature_C to Fahrenheit

- Reason: Temperature is commonly reported in different units depending on the region. Converting Celsius to Fahrenheit ensures compatibility with audiences or models that require temperature in Fahrenheit.
- Value: Enhances accessibility for users in regions where Fahrenheit is the standard unit and

allows for better cross-regional comparisons.

Normalizing Wind_Speed_kmh to mph

- Reason: Similar to temperature conversion, wind speed is often recorded in different units (km/h vs. mph). Converting it ensures consistency and usability across different datasets.
- Value: Enables seamless integration with other weather data sources and enhances the interpretability of wind speed information for users accustomed to mph.

Computing the Average Temperature per Location

- Reason: Aggregating temperature values provides insights into long-term trends and climate patterns for each location. This transformation helps smooth out short-term fluctuations and highlights the overall climate conditions in different regions.
- Value: Useful for climate monitoring, regional weather comparisons, and decision-making in sectors such as agriculture, energy, and urban planning.

Importance of These Transformations

These transformations enhance the dataset's accuracy, consistency, and analytical potential. They enable:

- Better trend analysis (seasonal/weather patterns over time).
- Improved data consistency (avoiding duplicate or inconsistent values).
- Enhanced usability for different audiences (unit conversions).
- More effective aggregation and comparisons across locations.

id	Location	Date_Time	Temperature_C	Humidity_pct	Precipitation_mm	Wind_Speed_kmh	Year	Month	Day	Hour	Minute	Temperature_F	Wind_Speed_mph	Avg_Temperature_C
10996	SAN JOSE	2024-01-21 01:22:18	26.610890	87.424733	7.197925	5.749313	2024	1	21	1	22	79.899602	3.572457	15.470219
10997	DALLAS	2024-03-07 11:37:08	28.178715	86.330654	2.093757	10.659999	2024	3	7	11	37	82.721687	6.623814	14.884943
10998	SAN JOSE	2024-01-19 01:03:35	38.268248	66.925891	3.060138	17.813180	2024	1	19	1	3	100.882847	11.068594	15.470219
10999	SAN JOSE	2024-04-23 18:56:54	19.845844	36.820104	3.341647	14.061083	2024	4	23	18	56	67.722519	8.737149	15.470219
11000	SAN DIEGO	2024-02-05 11:08:00	23.833778	81.874137	8.119941	13.543413	2024	2	5	11	8	74.900800	8.415484	14.858508

Figure 6. View of transformation

Upload to supabase

id	Location	Date_Time	Temperature_C	Humidity_pct	Precipitation_mm	Wind_Speed_kmh	Year	Month	Day	Hour	Minute	Temperature_F	Wind_Speed_mph	Avg_Temperature_C
1	SAN DIEGO	2024-01-14 21:12:46	10.683001094754	41.1987336609448	4.0201867070867	8.23364034467202	2024	01	14	21	12	50.8274	5.1151	5.0942
2	SAN DIEGO	2024-05-17 16:30:40	8.734397763304	68.3101072968203	9.1862344832998	27.716105689328	2024	05	17	16	30	47.5275	17.2536	5.0942
3	SAN DIEGO	2024-06-19 09:30:59	16.4334383303039	58.8201076918956	4.8078103774803	26.7829610843362	2024	06	19	09	30	61.7827	16.7536	5.0942
4	PHILADELPHIA	2024-02-06 17:32:39	-6.0289758959331	54.074473970617	3.58379374780765	26.3673067036366	2024	02	06	17	32	30.3395	16.3536	5.0942
5	SAN ANTONIO	2024-04-29 13:23:01	39.808219746236	72.899979103431	9.08620213671966	29.898621693961	2024	04	29	13	23	103.4466	18.7536	5.0942
6	SAN DIEGO	2024-01-21 08:54:56	27.341054869204	49.0232360683476	9.5654330273274	27.4738960845188	2024	01	21	08	54	81.2285	17.2536	5.0942
7	SAN JOSE	2024-01-12 02:10:54	1.8818336759618	65.7423432569102	0.3270590429721	1.0731178192265	2024	01	12	02	10	35.3275	0.8151	5.0942
8	NEW YORK	2024-01-25 09:24:34	-4.85470554026456	30.8049840062328	8.0276230143893	16.543336616936	2024	01	25	09	24	24.3395	10.3536	5.0942
9	NEW YORK	2024-03-09 08:50:30	0.90334805354438	38.8191676405865	3.6401021674018	7.98930297169938	2024	03	09	08	50	33.4275	4.7536	5.0942
10	SAN JOSE	2024-05-18 09:14:02	-1.607038813842823	82.9487031003337	4.10460297768881	26.6477807014366	2024	05	18	09	14	29.7025	16.5536	5.0942
11	NEW YORK	2024-03-04 13:47:16	35.1483360600071	54.7128658168497	8.34994946614864	26.4303100198307	2024	03	04	13	47	95.4575	16.7536	5.0942
12	HOUSTON	2024-03-07 22:03:36	16.8167635681006	60.1990303450008	3.76000418818803	16.78133201043365	2024	03	07	22	03	62.2825	9.3536	5.0942
13	DALLAS	2024-03-27 21:07:10	32.096889079419	53.9437064102686	3.58867605440188	3.05019633656485	2024	03	27	21	07	90.5275	7.2536	5.0942
14	HOUSTON	2024-05-09 00:53:10	38.4415689540168	85.9527253589403	0.47078143003506	30.7726540309	2024	05	09	00	53	100.6075	16.7536	5.0942
15	HOUSTON	2024-05-16 16:57:55	39.666776936373	72.7470268180683	1.2637215889168	6.47946020787064	2024	05	16	16	57	103.4075	8.1536	5.0942
16	PHILADELPHIA	2024-02-09 01:01:04	28.2501160382208	38.2391696910979	9.54705472330762	14.0667646404653	2024	02	09	01	01	82.6525	9.3536	5.0942
17	SAN ANTONIO	2024-01-10 16:06:58	16.3697950033687	65.8106077013181	0.100900207726223	6.89703901964058	2024	01	10	16	06	58.2825	0.8151	5.0942
18	CHICAGO	2024-01-06 02:59:46	26.7861910939322	21.6136144259176	0.0494024001012918	22.38000930164045	2024	01	06	02	59	80.1125	5.5536	5.0942
19	SAN ANTONIO	2024-05-08 16:30:53	35.795476379057	35.0830709963318	9.5875940606898	4.50786070079489	2024	05	08	16	30	95.2325	8.1536	5.0942
20	SAN DIEGO	2024-01-31 05:38:46	14.6058989278471	66.6423258195403	1.818637133483	28.4389020005844	2024	01	31	05	38	58.2825	16.7536	5.0942
21	SAN DIEGO	2024-01-23 02:59:32	33.023302764717	62.607648468407	0.21014268771479	16.7333348913416	2024	01	23	02	59	91.3325	5.5536	5.0942

Figure 7. Transformation merged 1

The screenshot shows the ProjectETL Table Editor interface. On the left, there's a sidebar with 'schema public' and a 'New table' button. Below that is a search bar and a list of tables, with 'weather' selected. The main area displays a table with the following columns: `ts_time` (timestamp), `Year` (int4), `Month` (int4), `Day` (int4), `Hour` (int4), `Minute` (int4), `Temperature_F` (float8), `Wind_Speed_mph` (float8), and `Avg_Temperature_C` (float8). The table contains 100 rows of data, with a total of 11,000 records. The bottom status bar shows 'Page 1 of 110', '100 rows', and '11,000 records'. There are also buttons for 'Refresh', 'Data', and 'Definition'.

<code>ts_time</code>	<code>Year</code>	<code>Month</code>	<code>Day</code>	<code>Hour</code>	<code>Minute</code>	<code>Temperature_F</code>	<code>Wind_Speed_mph</code>	<code>Avg_Temperature_C</code>
24-01-14 21:12:46	2024	1	14	21	12	51.2294019704877	5.11608313673973	14.8585080184286
24-05-17 15:22:10	2024	5	17	15	22	47.7214516082365	17.2213974653565	14.8585080184286
24-05-11 09:30:59	2024	5	11	9	30	52.9383863632756	17.8538226749226	14.8585080184286
24-02-26 17:32:39	2024	2	26	17	32	16.467843387751	16.3838772289367	14.9804392867645
24-04-29 13:23:51	2024	4	29	13	23	103.654783354337	18.5781364462722	14.3025535075805
24-01-21 08:54:56	2024	1	21	8	54	81.21389876544232	17.0714822839317	14.8585080184286
24-01-13 02:10:54	2024	1	13	2	10	35.3873900623427	0.666800540548093	15.4702188292907
24-01-25 19:04:34	2024	1	25	19	4	19.5894220262818	10.4690679262436	14.670617925177
24-03-29 05:20:30	2024	3	29	5	20	33.734380809638	4.96414781431882	14.670617925177
24-05-18 09:14:02	2024	5	18	9	14	29.1072415308341	15.9364773073892	15.4702188292907
24-03-04 13:47:15	2024	3	4	13	47	95.2620063170308	15.8016571671458	14.670617925177
24-03-07 22:03:36	2024	3	7	22	3	60.4701764225811	10.4092891411214	14.7136202619159
24-02-27 21:07:10	2024	2	27	21	7	89.6304156942954	1.89530357829444	14.8849426799027
24-05-09 00:53:10	2024	5	9	0	53	101.554284676323	12.916320577169	14.7136202619159
24-05-12 15:57:55	2024	5	12	15	57	103.400189453061	4.02616848403529	14.7136202619159
24-03-09 01:51:24	2024	3	9	1	51	82.922206438927	8.74067961122415	14.9804392867645
24-02-10 15:05:28	2024	2	10	15	5	61.4296211006637	4.09920874505061	14.3025535075805
24-01-06 02:59:46	2024	1	6	2	59	80.216259964505	14.27916479995174	15.5166113027131
24-05-08 16:20:53	2024	5	8	16	20	95.3231857482303	2.80105515736036	14.3025535075805
24-01-31 05:38:46	2024	1	31	5	38	58.2904746097648	18.2881230459961	14.8585080184286
24-01-25 12:59:32	2024	1	25	12	59	91.4420312896491	10.3976028210578	14.8585080184286

Figure 8. Transformation merged 2

References

- [1] Prasad22. (n.d.). *Weather data* [Dataset]. Kaggle. Retrieved from <https://www.kaggle.com/datasets/prasad22/weather-data>