Dataset used:

For this assignment's study, we will be dealing with the complete version of Schlumberger dataset. The dataset consists of time-series data from downhole equipment using intelligent completions which helps to evaluate and manage production in real-time with zonal downhole monitoring of pressures and temperatures, easing the maintenance of such costly equipment.

Cursory Analysis:

After downloading and opening the *data-large.csv* file from the GitHub repository as specified, we can immediately hint at a few probable errors we might face while importing the dataset into python with a script. Let's take a quick look. The immediate red flag I could notice right after opening the file is the issue with headers. This is a very common error; we can see that there are multiple header rows (3) in our file. The first row signifies if the equipment is an Electric Submersible Pump (ESP) or Intelligent Completion (IC). The second row indicates the type of sensor value recorded like motor temperature, choke pressure, VSDFreqOut, etc. The third row of the header shows the sensor's information with its unit of measure like psia, K or Hz etc.

These redundant headers can be removed and merged into a single header by eliminating the sensor name and the unit rows. The second-row names are renamed in an efficient manner to accurately depict the columns names by including the required information. Correspondingly, the first column contains information about the time in seconds. It can be pointed out that many columns are redundant, and our table should be free from these identical and indistinguishable column names. We can employ some modified arguments while importing and reading in the CSV file to resolve some issues. For example, since the dataset is very large, we can *set low_memory=False* minimalizing the storage consumption. Having *index_col=False* enforces the 'pandas' module not to use the first column as the index. In addition to these, explicitly stated the object type *dtype='unicode'* for it to return Unicode normal form.

We can observe an empty cell in the first row of the first column, which could result in an improper import as well. This cell is also removed as part of the data-cleaning process. The structure of the CSV file is absurd which results in an improper import if we read the file using a script in python. Thus, the data is organized and manipulated using Python's Panda module. To retain the values in the original data frame, the index is reset with the drop argument set to true. In addition, specifying dtype for columns can reduce storage requirements when storing the data frame in the database.

PHPMyAdmin & Possible Alternatives:

In many cases, MySQL is the most commonly used open-source RDBMS, due to its effectiveness, reliability, and simplicity of use - especially with the right tools. As a result, a variety of professional solutions are available for MySQL and MariaDB, a close relative of MySQL. Most of them are free or paid, more or less functional, and are created by various companies. The phpMyAdmin database management tool is well known to everyone who works with MySQL. As the name implies it's written in PHP, so it can be installed on your web server and accessed from a browser. The interface is multilingual and has 79 different language versions.

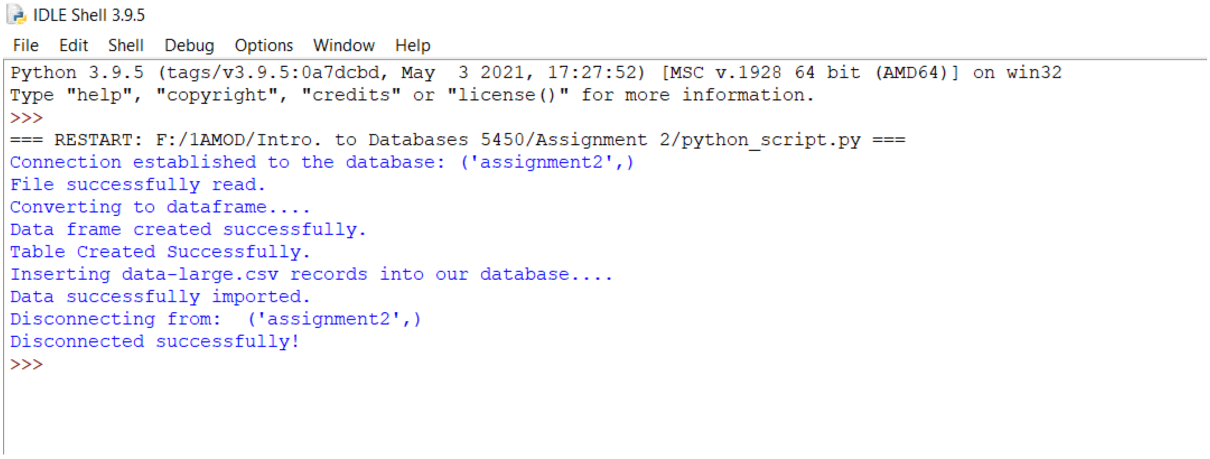
While it is a popular tool with a wide range of features, it cannot cover a few requirements. For example, phpMyAdmin does not provide scheme visualizations, full auto-completion capabilities, scheduled backup, and encryption. It is possible to export an unlimited number of records to SQL, CSV, or spreadsheets using PHPMyAdmin. The software is incapable of managing so much data when displaying a good interactive grid with visualizations as the contents of a table are injected into an object during rendering, and all of the contents are stored in memory. Trying to import a large database in MySQL (phpMyAdmin) with a file size of more than 20MB, the uploaded file will be rejected with an error. The problem occurs due to uploading a large file, hence “post max time” or “execution time” errors occur in PHP. These errors can be handled by doing minimal changes to XAMPP configuration. Without altering the XAMPP configuration, its possible to import large datasets by editing the php.ini file. For instance, PHP is unable to save information in a variable if 10,000 rows total more than 128M. That indicates that the script used more memory than the predetermined 128M limit. As a result, any significant growth in php.ini should be above 128M. Your next move should be to 256M. In order for the modifications to take effect, remember to restart Apache and XAMPP.

Noting this; as the current dataset deals with indistinguishable column names, varied data types and absurd structure, it would be vigilant to use python as an alternative to import our dataset and run queries due to the fact that it can handle a high volume of memory and data processing becomes a lot easier with the implementation of packages such as pandas and other great numbers of data-oriented feature packages that can speed up and simplify data processing, in turn saving a lot of time.

Python Script & Testing:

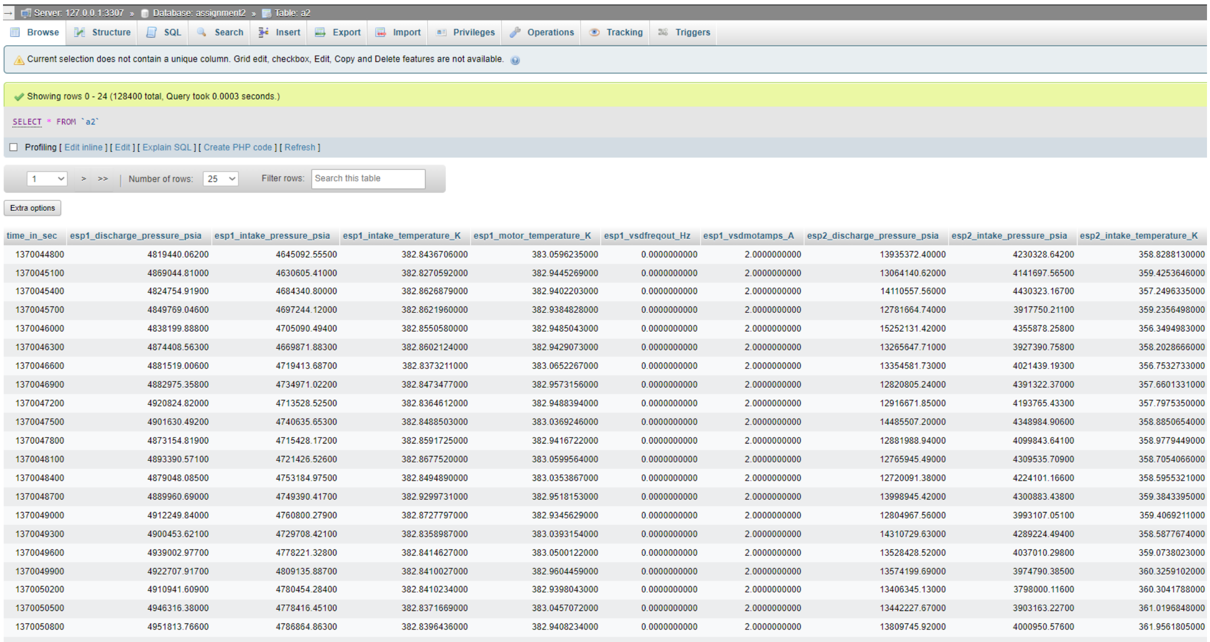
File attached.

Python O/P Screenshot:



```
Python 3.9.5 (tags/v3.9.5:0a7dcbd, May  3 2021, 17:27:52) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
=== RESTART: F:/1AMOD/Intro. to Databases 5450/Assignment 2/python_script.py ===
Connection established to the database: ('assignment2',)
File successfully read.
Converting to dataframe....
Data frame created successfully.
Table Created Successfully.
Inserting data-large.csv records into our database....
Data successfully imported.
Disconnecting from:  ('assignment2',)
Disconnected successfully!
>>> 
```

PHPMyAdmin Screenshot:


→ Server: 127.0.0.1:3307 » Database: assignment2 » Table: a2
Browse
Structure
SQL
Search
Insert
Export
Import
Privileges
Operations
Tracking
Triggers

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 24 (128400 total, Query took 0.0003 seconds.)

```
SELECT * FROM `a2`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
1 > >>
Extra options

Number of rows: 25 Filter rows: Search this table

time_in_sec	esp1_discharge_pressure_psia	esp1_intake_pressure_psia	esp1_intake_temperature_K	esp1_motor_temperature_K	esp1_vsdfreqout_Hz	esp1_vsdmotamps_A	esp2_discharge_pressure_psia	esp2_intake_pressure_psia	esp2_intake_temperature_K
1370044800	4819440.06200	4645092.55500	382.8436706000	383.0596235000	0.0000000000	2.0000000000	13935372.40000	4230328.64200	358.8288130000
1370045100	4869044.81000	4630605.41000	382.8270592000	382.9445269000	0.0000000000	2.0000000000	13064140.62000	4141697.56500	359.4253646000
1370045400	4824754.91900	4684340.80000	382.8626879000	382.9402203000	0.0000000000	2.0000000000	14110557.56000	4430323.16700	357.2496335000
1370045700	4849769.04600	4697244.12000	382.8621960000	382.9384828000	0.0000000000	2.0000000000	12781664.74000	3917750.21100	359.2356498000
1370046000	4838199.88800	4705090.49400	382.8550580000	382.9485043000	0.0000000000	2.0000000000	15252131.42000	4355878.25800	356.3494983000
1370046300	4874408.56300	4669871.88300	382.8602124000	382.9429073000	0.0000000000	2.0000000000	13265647.71000	3927390.75800	358.2028666000
1370046600	4881519.00600	4719413.68700	382.8373211000	383.0652267000	0.0000000000	2.0000000000	13354581.73000	4021439.19300	356.7532733000
1370046900	4882975.35800	4734971.02200	382.8473477000	382.9573156000	0.0000000000	2.0000000000	12820805.24000	4391322.37000	357.6601331000
1370047200	4920824.82000	4713528.52500	382.8364612000	382.9488394000	0.0000000000	2.0000000000	12916671.85000	4193765.43300	357.7975350000
1370047500	4901630.49200	4740635.65300	382.8488503000	383.0369246000	0.0000000000	2.0000000000	14485507.20000	4348984.90600	358.8850654000
1370047800	4873154.81900	4715428.17200	382.8591725000	382.9416722000	0.0000000000	2.0000000000	12881988.94000	4099843.64100	358.9779449000
1370048100	4893390.57100	4721426.52600	382.8677520000	383.0599564000	0.0000000000	2.0000000000	12765945.49000	4309535.70900	358.7054066000
1370048400	4879048.08500	4753184.97500	382.8494890000	383.0353867000	0.0000000000	2.0000000000	12720091.38000	4224101.16600	358.5955321000
1370048700	4889960.69000	4749390.41700	382.9299731000	382.9518153000	0.0000000000	2.0000000000	13998945.42000	4300883.43800	359.3843395000
1370049000	4912249.84000	4760800.27900	382.8727797000	382.9345629000	0.0000000000	2.0000000000	12804967.56000	3993107.05100	359.4069211000
1370049300	4900453.62100	4729708.42100	382.8358987000	383.0393154000	0.0000000000	2.0000000000	14310729.63000	4289224.49400	358.5877674000
1370049600	4939002.97700	4778221.32800	382.8414627000	383.0500122000	0.0000000000	2.0000000000	13528428.52000	4037010.29800	359.0738023000
1370049900	4922707.91700	4809135.88700	382.8410027000	382.9604459000	0.0000000000	2.0000000000	13574199.69000	3974790.38500	360.3259102000
1370050200	4910941.60900	4780454.28400	382.8410234000	382.9398043000	0.0000000000	2.0000000000	13406345.13000	3798000.11600	360.3041788000
1370050500	4946316.38000	4778416.45100	382.8371669000	383.0457072000	0.0000000000	2.0000000000	13442227.67000	3903163.22700	361.0196848000
1370050800	4951813.76600	4786864.86300	382.8396436000	382.9408234000	0.0000000000	2.0000000000	13809745.92000	4000950.57600	361.9561805000

Database Normal Form:

A functional dependency is a constraint between two attributes in a relation. Some of the different functional dependencies in Schlumberger dataset:

- The ESP motor speed and frequency depend on motor temperature.
- Additionally, ESP is also dependent on discharge and pump intake pressures.
- The temperature 2 characteristic specified in the dataset, which is measured in F and depends on the downhole temperature reported by the intelligent control system.
- As pressure 1 and pressure 2 (psia) parameters are subtracted to create delta pressure, the Intelligent Control System Liquid Rate (bbl/d) is dependent on these attributes.
- Also, the Choke position can be determined on pressure 1 and pressure 2.
- The Water rate (bbl/d) depends on Liquid rate and water cut and is defined as (Liquid Rate*Water Cut).
- The Oil rate depends on Liquid rate and water rate and is defined as (Liquid Rate - Water Rate).

Is the database table in 1NF, 2NF, or 3NF?

The database table is already in 1NF as there are no multi-valued characteristic records. With a unique column timestamp acting as the primary key in 1NF, there should be no partial dependencies for it to be in 2NF. So, the table is in 2NF by default. In order to convert it to 3NF, we can separate timestamp-IC, timestamp-EC values into separate tables along with other functional dependencies.

Useful Analytical Queries:

The explanation of the queries used here are self-explanatory. These analytical queries can be properly used to monitor and maintain the equipment as the consequences are significant.

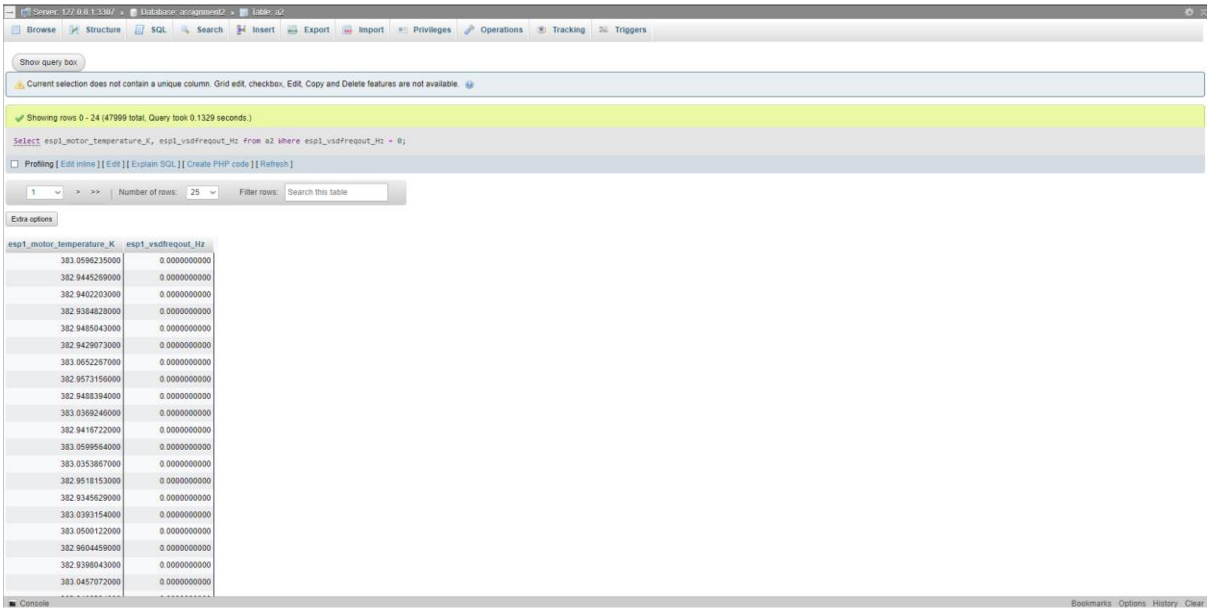
1. How is the motor temperature of ESP01 when the variable speed driver output frequency is 0 hz ?

Query:

SELECT esp1_motor_temperature_K, esp1_vsdfreqout_Hz from a2

WHERE esp1_vsdfreqout_Hz = 0;

Output:



Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 24 (47999 total, Query took 0.1329 seconds.)

```
Select esp1_motor_temperature_K, esp1_vsdfreqout_Hz from a2 Where esp1_vsdfreqout_Hz = 0;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> | Number of rows: 25 Filter rows: Search this table

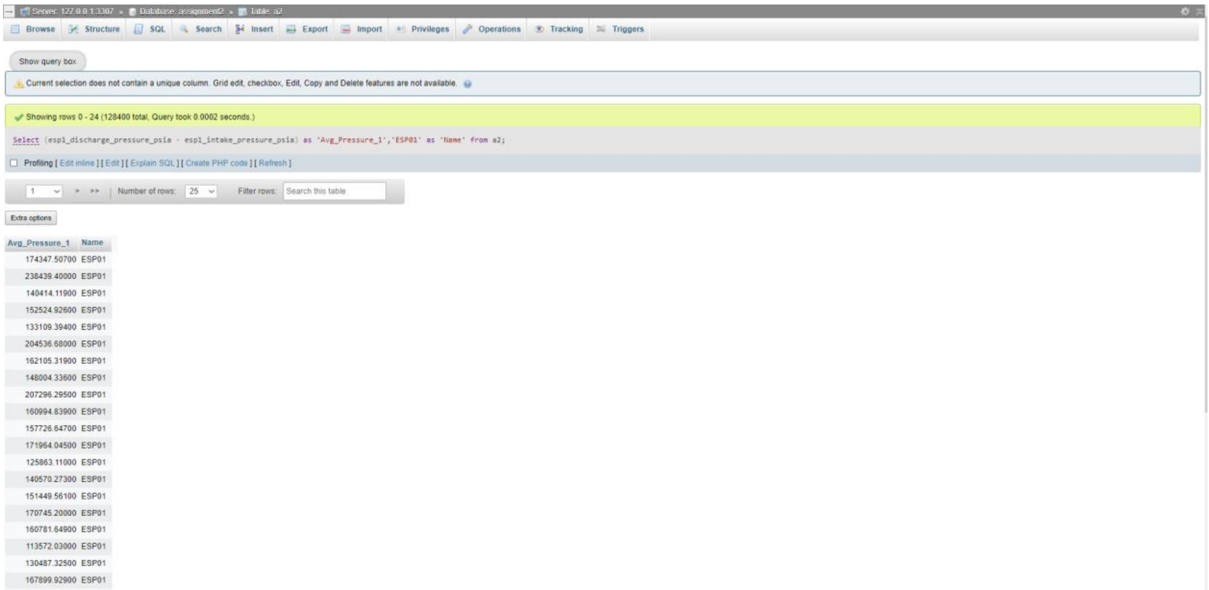
esp1_motor_temperature_K	esp1_vsdfreqout_Hz
383.0596235000	0.0000000000
382.9445269000	0.0000000000
382.9402203000	0.0000000000
382.9384828000	0.0000000000
382.9485043000	0.0000000000
382.9429073000	0.0000000000
383.0652267000	0.0000000000
382.9573156000	0.0000000000
382.9488394000	0.0000000000
383.0369246000	0.0000000000
382.9416722000	0.0000000000
383.0599564000	0.0000000000
383.0353867000	0.0000000000
382.9518153000	0.0000000000
382.9345629000	0.0000000000
383.0393154000	0.0000000000
383.0500122000	0.0000000000
382.9604459000	0.0000000000
382.9398043000	0.0000000000
383.0457072000	0.0000000000

2. Calculate the average change in pressure for pump 1.

Query:

SELECT (esp1_discharge_pressure_psia - esp1_intake_pressure_psia) as 'Avg_Pressure_1','ESP01' as 'Name' from a2;

Output:



Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 24 (128400 total, Query took 0.0002 seconds.)

```
Select (esp1_discharge_pressure_psia - esp1_intake_pressure_psia) as 'Avg_Pressure_1','ESP01' as 'Name' from a2;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> | Number of rows: 25 Filter rows: Search this table

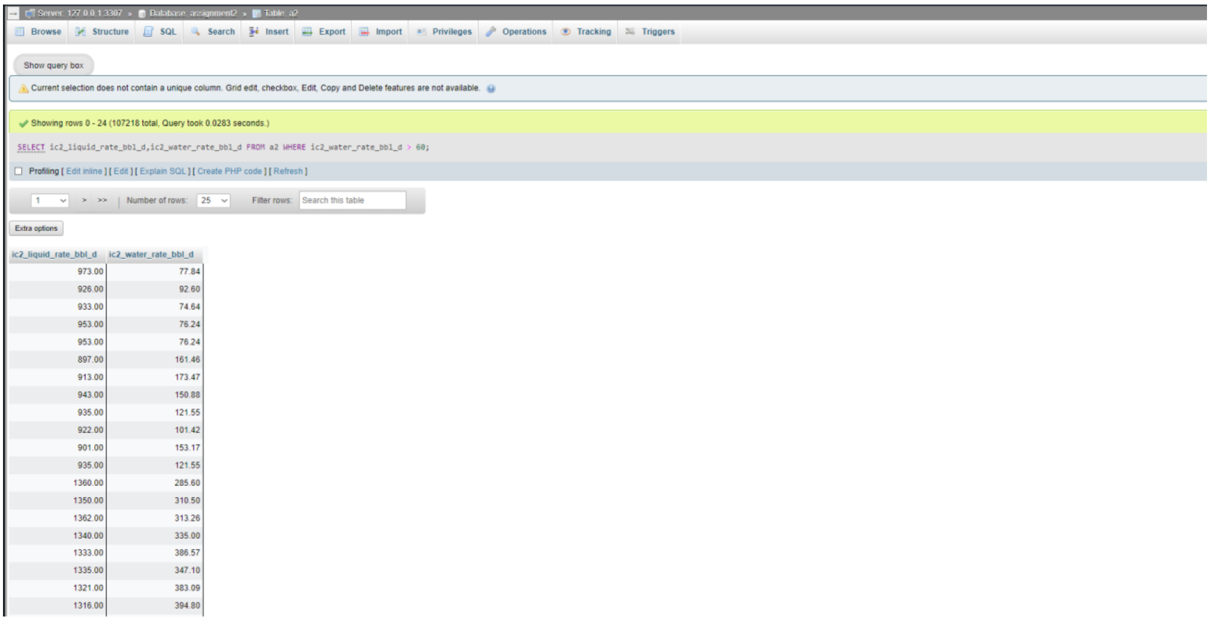
Avg_Pressure_1	Name
174347.50700	ESP01
238439.40000	ESP01
140414.11900	ESP01
152524.92600	ESP01
133109.39400	ESP01
204536.68000	ESP01
162105.31900	ESP01
148004.33600	ESP01
207296.29500	ESP01
160994.83900	ESP01
157726.64700	ESP01
171964.04500	ESP01
125863.11000	ESP01
140570.27300	ESP01
151449.56100	ESP01
170745.20000	ESP01
160781.64900	ESP01
113572.03000	ESP01
130487.32500	ESP01
167899.92900	ESP01

3. The volume flow of water generated is known as the water rate, while the volume flow of liquid produced is known as the liquid rate. Let's examine the liquid rate at a water rate of more than 60.

Query:

SELECT ic2_liquid_rate_bbl_d,ic2_water_rate_bbl_d FROM a2 WHERE ic2_water_rate_bbl_d > 60;

Output:



⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 24 (107218 total, Query took 0.0283 seconds.)

```
SELECT ic2_liquid_rate_bbl_d,ic2_water_rate_bbl_d FROM a2 WHERE ic2_water_rate_bbl_d > 60;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> | Number of rows: 25 Filter rows: Search this table

ic2_liquid_rate_bbl_d	ic2_water_rate_bbl_d
973.00	77.84
926.00	92.60
933.00	74.64
953.00	76.24
953.00	76.24
897.00	161.46
913.00	173.47
943.00	150.88
935.00	121.55
922.00	101.42
901.00	153.17
935.00	121.55
1360.00	285.60
1350.00	310.50
1362.00	313.26
1340.00	335.00
1333.00	386.57
1335.00	347.10
1321.00	383.09
1316.00	394.80

Attribution:

These assignments were completed by Ravi Chandan Pandi, and they represent his original work completed for academic purposes during his studies and self-learning purposes.

Please note that the documents shared here are intended for educational and informational purposes only. Any unauthorized use or reproduction is strictly prohibited. If you have any questions or would like to reach out to Ravi, you can contact him on LinkedIn. [https://www.linkedin.com/in/ravichandan/].