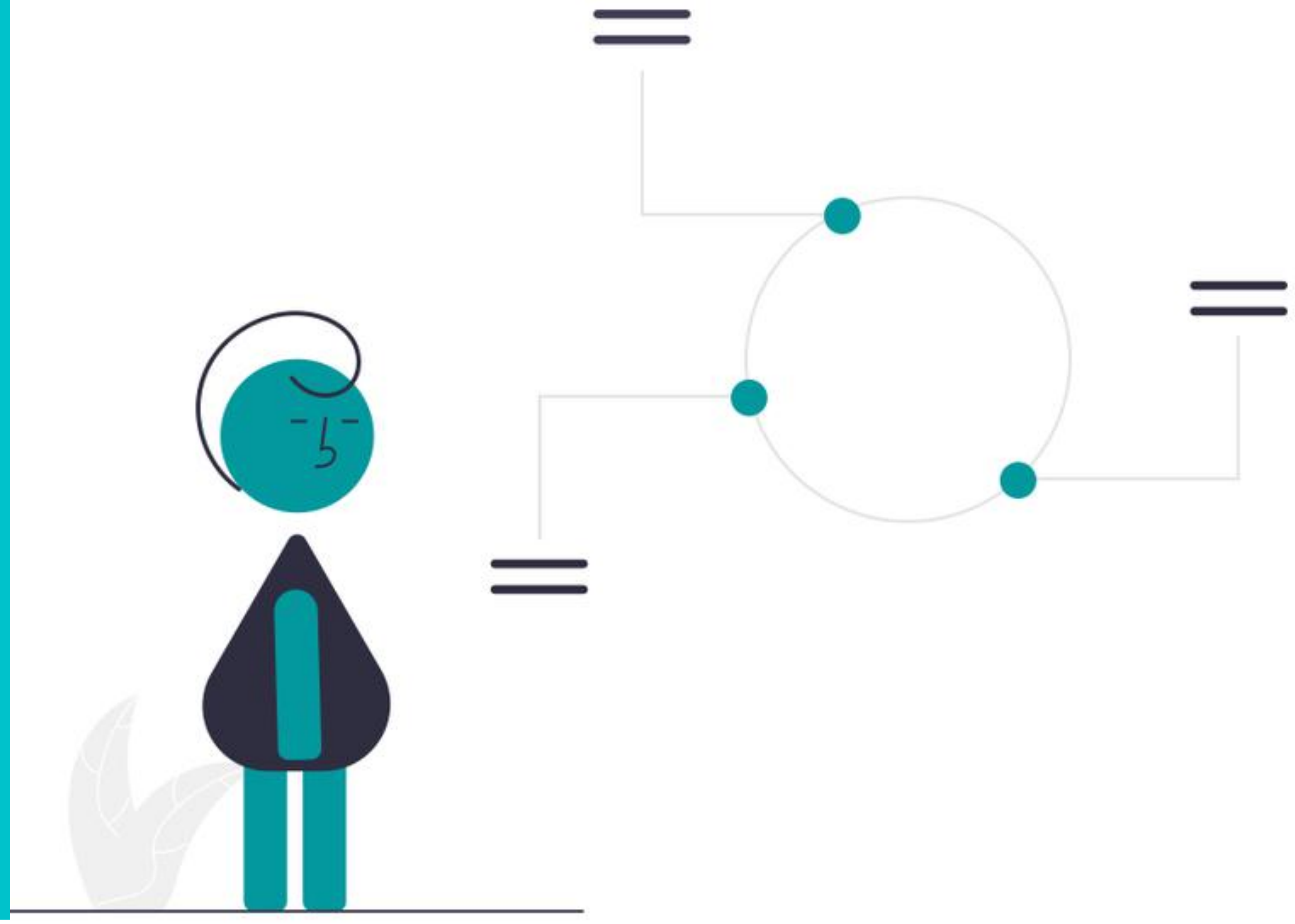
BIG DATA & ANALYTICS

TORTURE THE DATA
AND IT WILL CONFESS TO ANYTHING

From data to analytics



?

What do we define as data?

What do we define as data?

Information that is collected and translated for a certain cause.

What is the difference between **qualitative** and **quantitative** data?

Mention a **qualitative** and a **quantitative** datum for a dog.

!

What is the difference between **qualitative** and **quantitative** data?

Some data are **qualitative** (they describe something) while others are **quantitative** (information is numerical.

e.g."I had a nice time" vs "I have 5 apples"

What is the difference between **qualitative** and **quantitative** data?

Mention a **qualitative** and a **quantitative** datum for a dog.

!

Mention a **qualitative** and a **quantitative** datum for a dog.

It is black
It has long hair
It is very energetic

Vs

It has 4 legs
It has 2 brothers
It weighs 20 kg

Types of data

1

2

3

Structured

- **Easy** to store, process and analyze
- **~5–10%** of total data

Types of data

1

2

3

Structured

- **Easy** to store, process and analyze
- **~5–10%** of total data

Unstructured

- **Difficult** to categorize
- **~80%** of total data

Types of data

1

Structured

- **Easy** to store, process and analyze
- **~5-10%** of total data

2

Semi-Structured

- Mixture of other 2
- Can be **categorized** and are **easier** to analyze

3

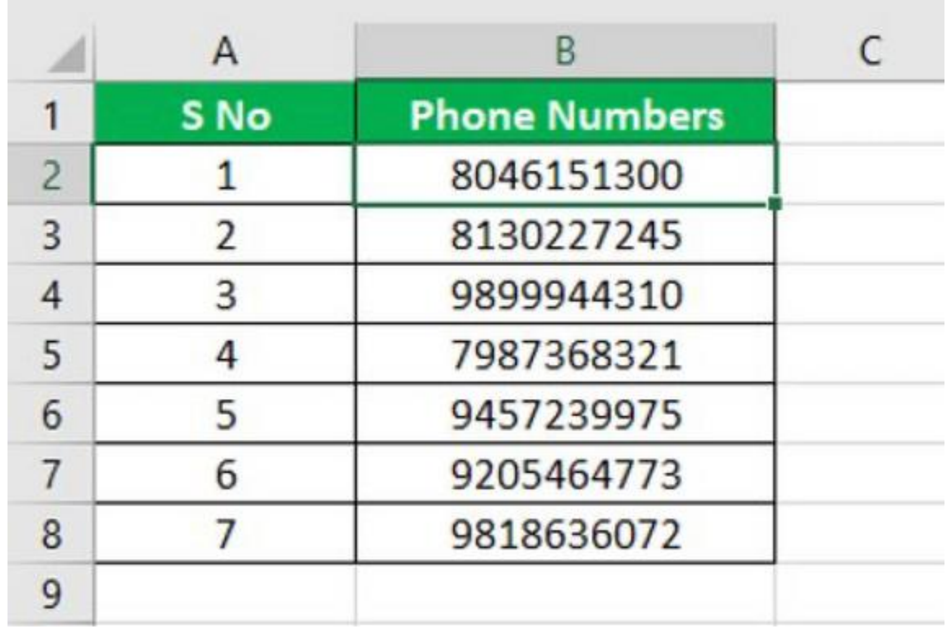
Unstructured

- **Difficult** to categorize
- **~80%** of total data

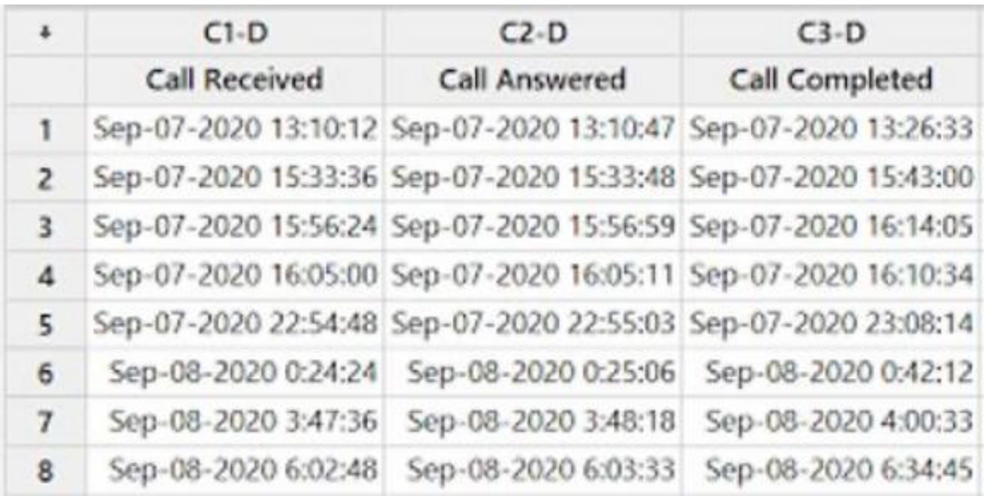
Examples of Data Types

Structured

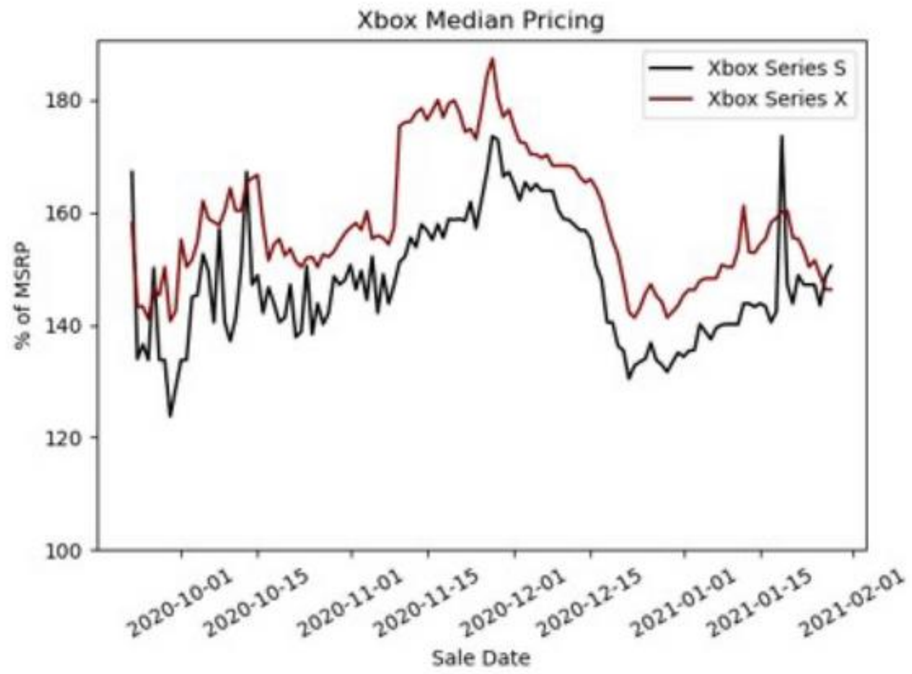
- Time and Date
- Phone numbers
- Bank transactions information
- Names, addresses and e-mail
- Product prices



	A	B	C
1	S No	Phone Numbers	
2	1	8046151300	
3	2	8130227245	
4	3	9899944310	
5	4	7987368321	
6	5	9457239975	
7	6	9205464773	
8	7	9818636072	
9			



↓	C1-D	C2-D	C3-D
	Call Received	Call Answered	Call Completed
1	Sep-07-2020 13:10:12	Sep-07-2020 13:10:47	Sep-07-2020 13:26:33
2	Sep-07-2020 15:33:36	Sep-07-2020 15:33:48	Sep-07-2020 15:43:00
3	Sep-07-2020 15:56:24	Sep-07-2020 15:56:59	Sep-07-2020 16:14:05
4	Sep-07-2020 16:05:00	Sep-07-2020 16:05:11	Sep-07-2020 16:10:34
5	Sep-07-2020 22:54:48	Sep-07-2020 22:55:03	Sep-07-2020 23:08:14
6	Sep-08-2020 0:24:24	Sep-08-2020 0:25:06	Sep-08-2020 0:42:12
7	Sep-08-2020 3:47:36	Sep-08-2020 3:48:18	Sep-08-2020 4:00:33
8	Sep-08-2020 6:02:48	Sep-08-2020 6:03:33	Sep-08-2020 6:34:45

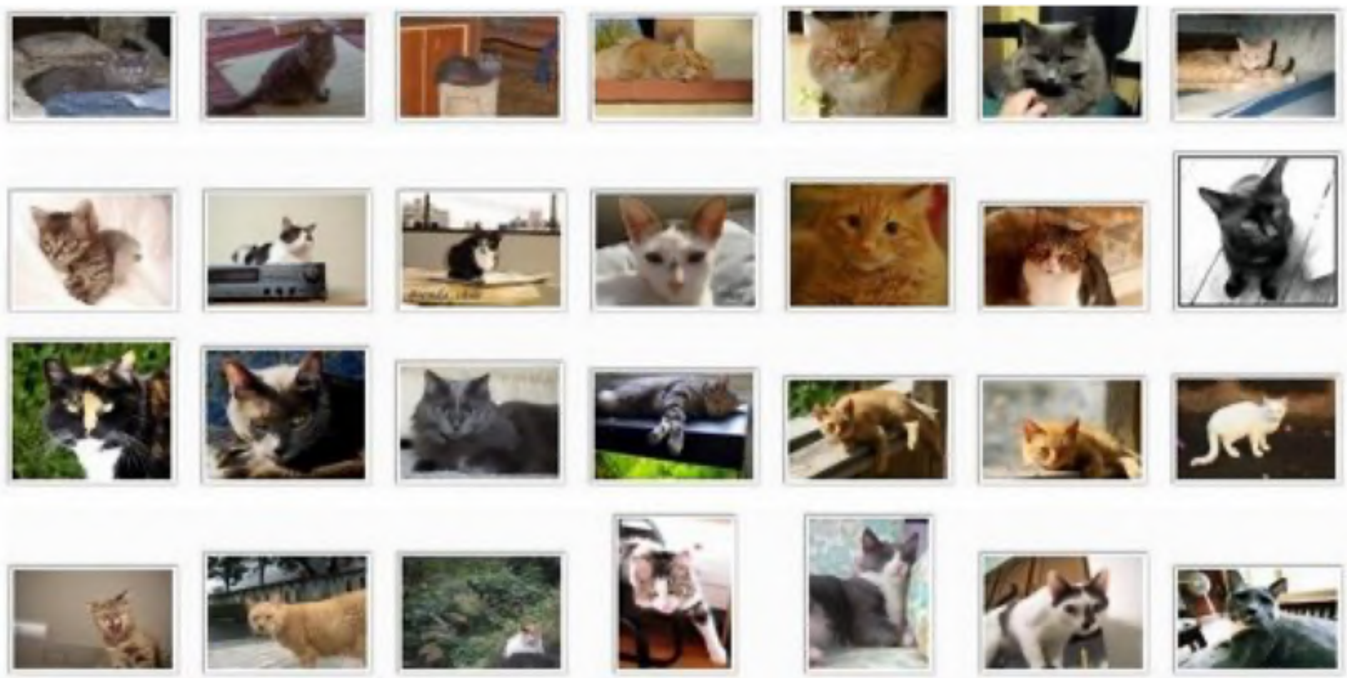


- Hotel reservation systems
- Purchase log software
- Medical devices

Examples of Data Types

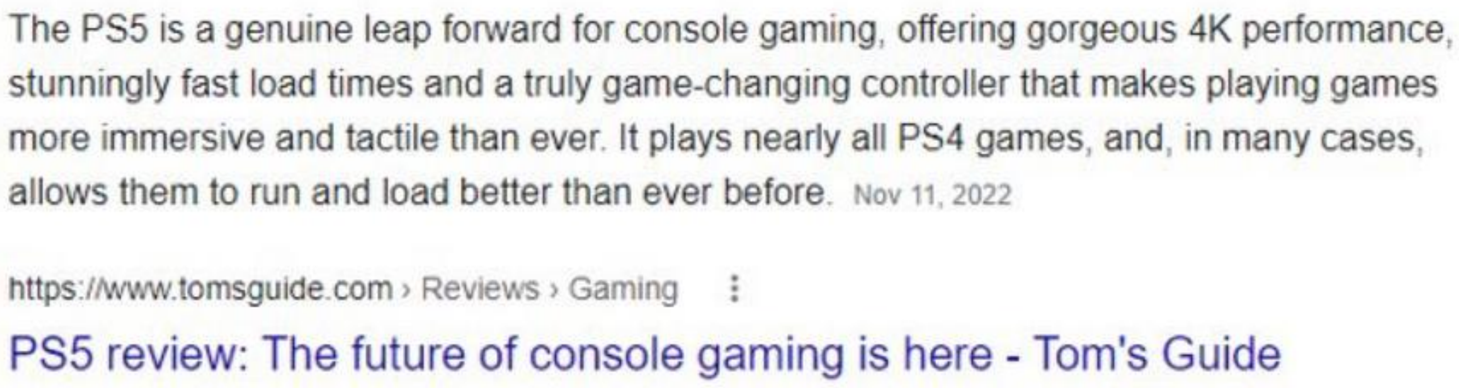
- txt files
- e-mail content
- Sound and image files
- Photos
- Camera recordings
- Books
- Product ratings

Unstructured



TXT

- Social media
- Websites
- Text processing software
- Presentations
- GPS, satellites
- Messaging apps



The PS5 is a genuine leap forward for console gaming, offering gorgeous 4K performance, stunningly fast load times and a truly game-changing controller that makes playing games more immersive and tactile than ever. It plays nearly all PS4 games, and, in many cases, allows them to run and load better than ever before. Nov 11, 2022

https://www.tomsguide.com › Reviews › Gaming ⋮

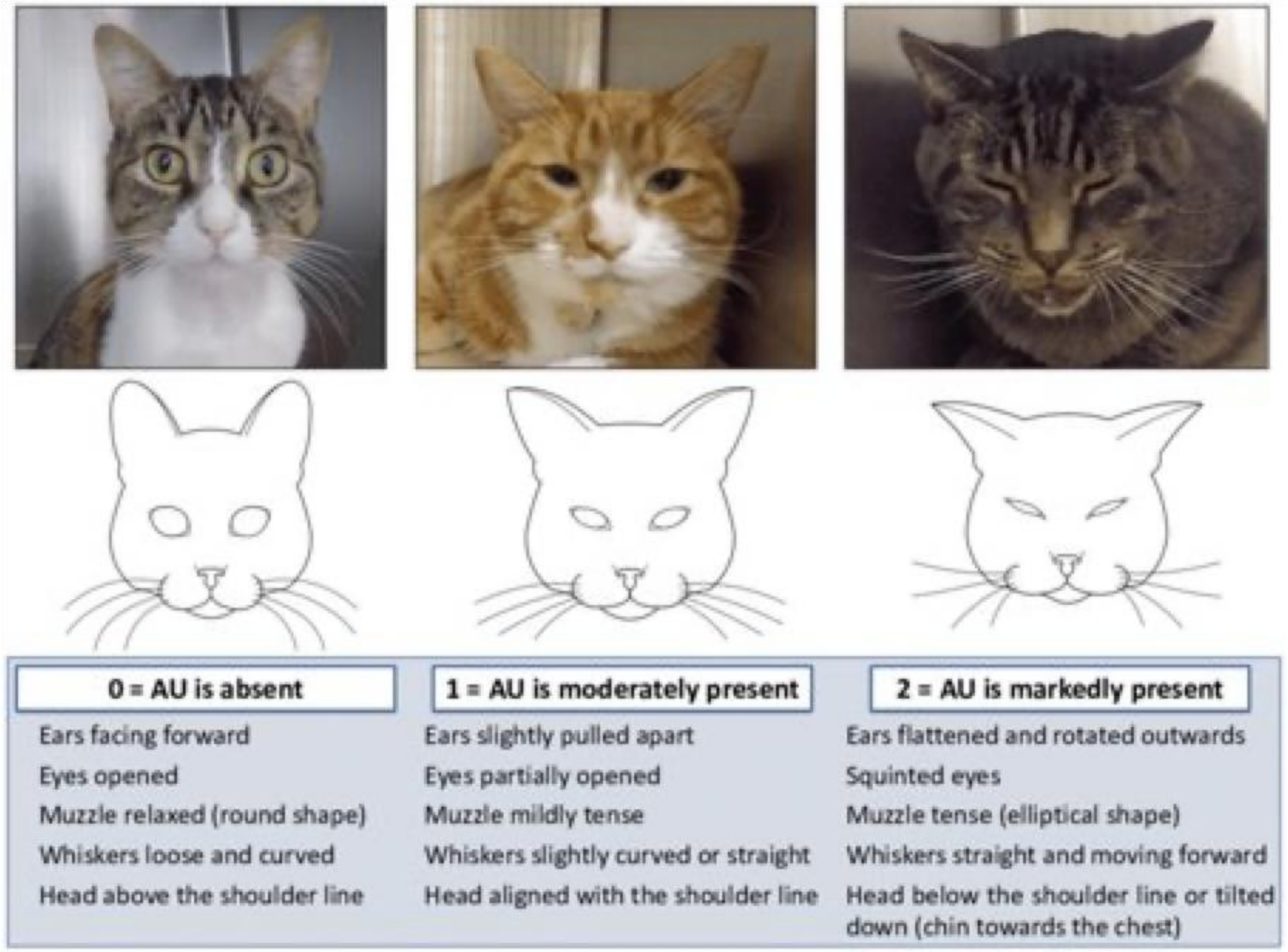
PS5 review: The future of console gaming is here - Tom's Guide

Examples of Data Types

Semi-structured

- Website that contains title, small description and helps differentiate its contents

- Images online that are accompanied by a brief description



0 = AU is absent	1 = AU is moderately present	2 = AU is markedly present
Ears facing forward	Ears slightly pulled apart	Ears flattened and rotated outwards
Eyes opened	Eyes partially opened	Squinted eyes
Muzzle relaxed (round shape)	Muzzle mildly tense	Muzzle tense (elliptical shape)
Whiskers loose and curved	Whiskers slightly curved or straight	Whiskers straight and moving forward
Head above the shoulder line	Head aligned with the shoulder line	Head below the shoulder line or tilted down (chin towards the chest)

Structured

Pros

- Easy to be used by **Machine Learning algorithms**
- Easy to be used by **user with no prior extensive knowledge**
- Can be managed by a great amount of **analytics tools**

- Limited **use**
- Limited **storing options**

Cons

Unstructured

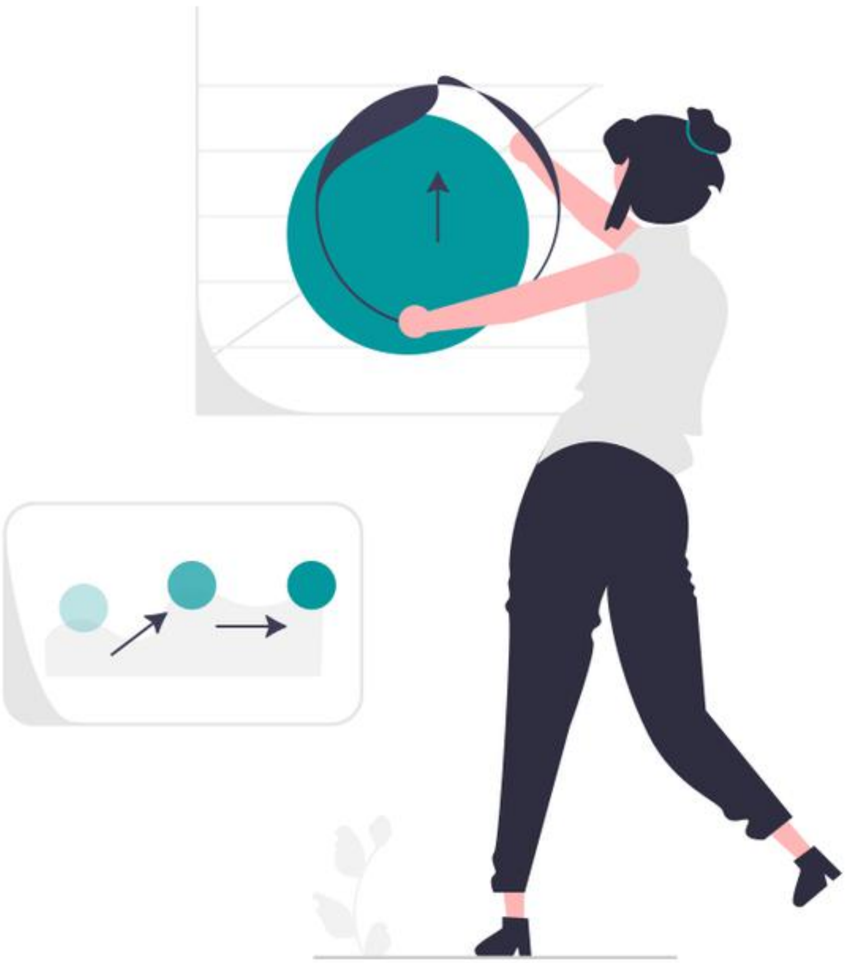
Pros

- In their **initial, raw form** and can be modified by the data engineer accordingly
- **Faster generation rates**
- **Easier and cheaper storage** only of the necessary data

Cons

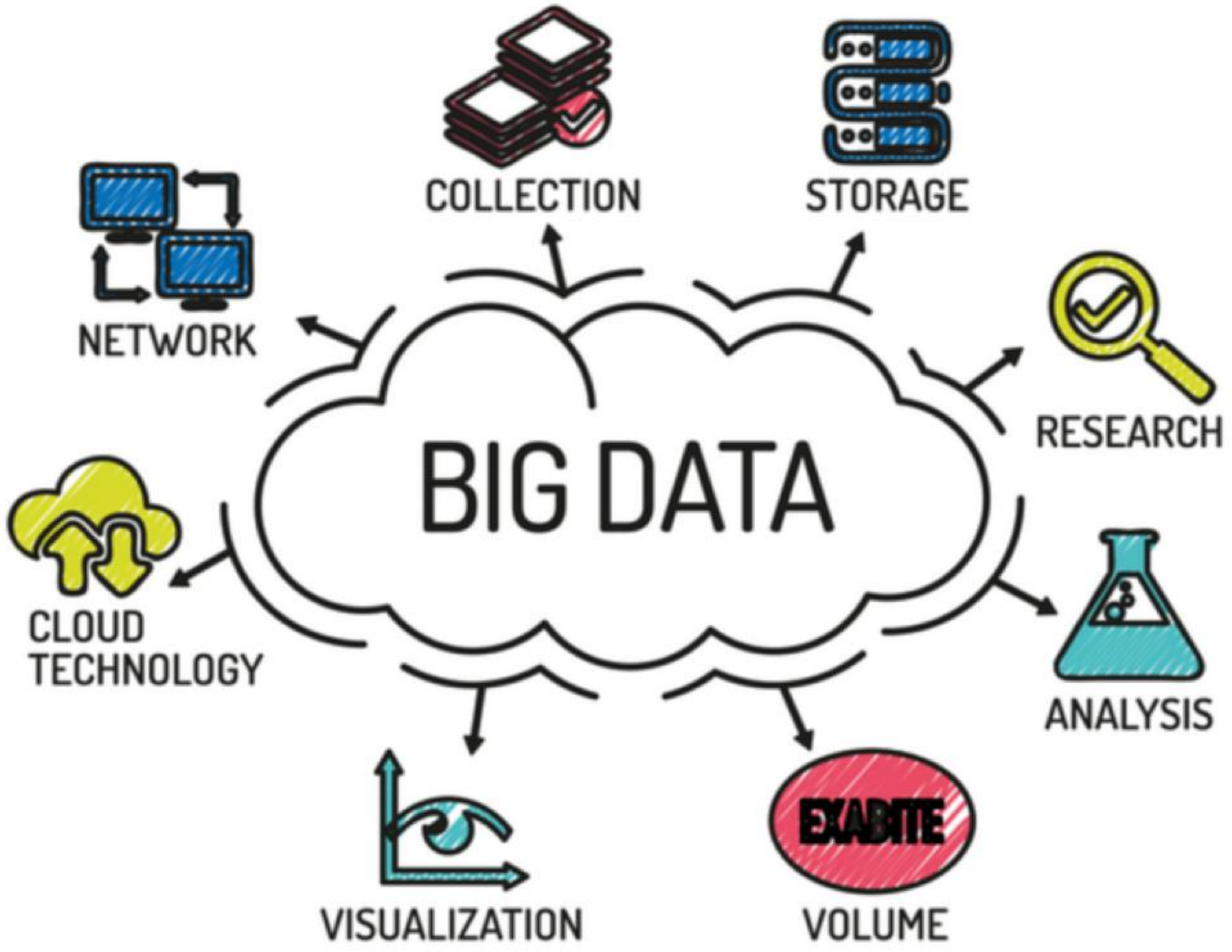
- Require **specialized knowledge**
- Require **special analytics and processing tools**

Big Data



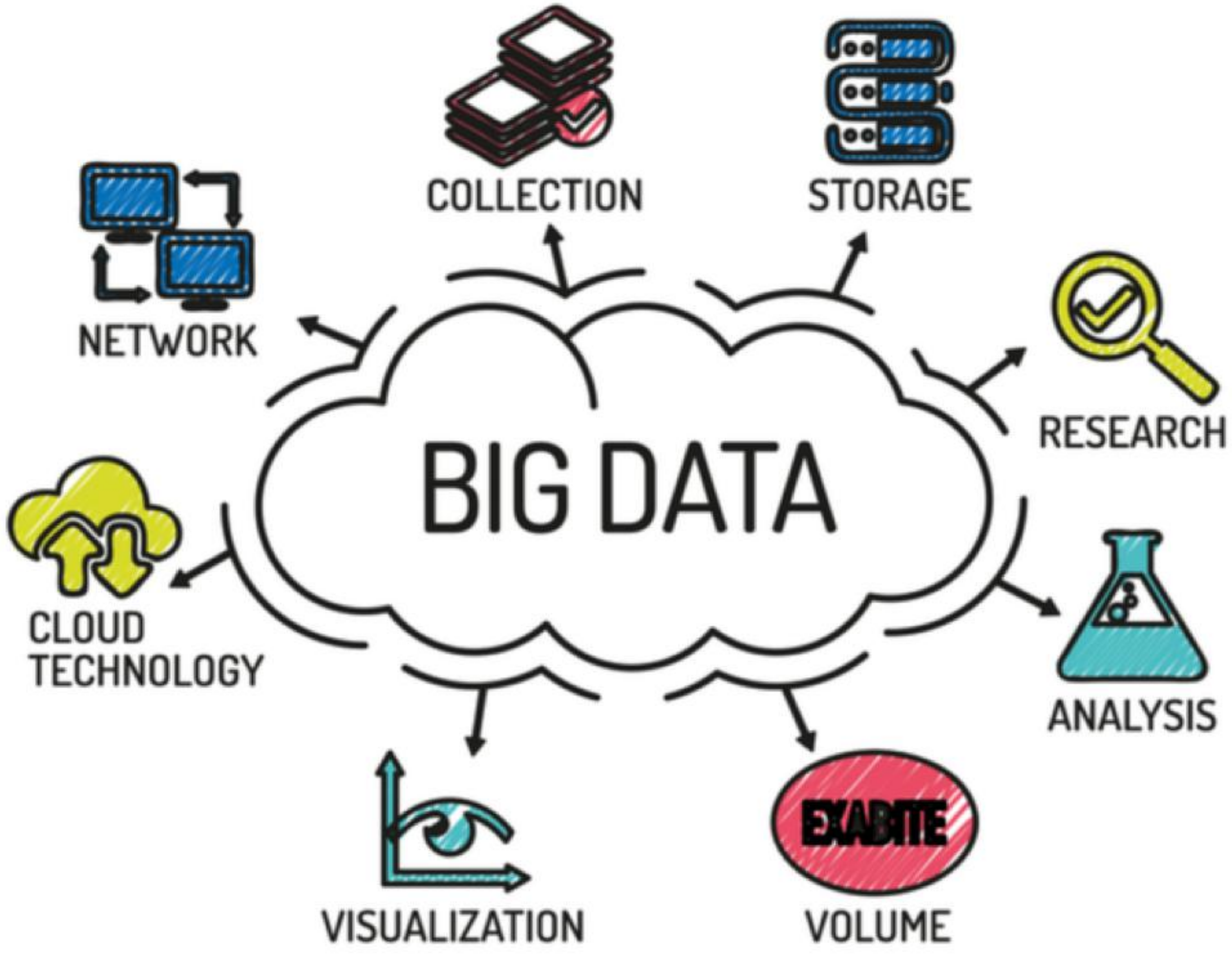
?

What is the definition of Big Data?



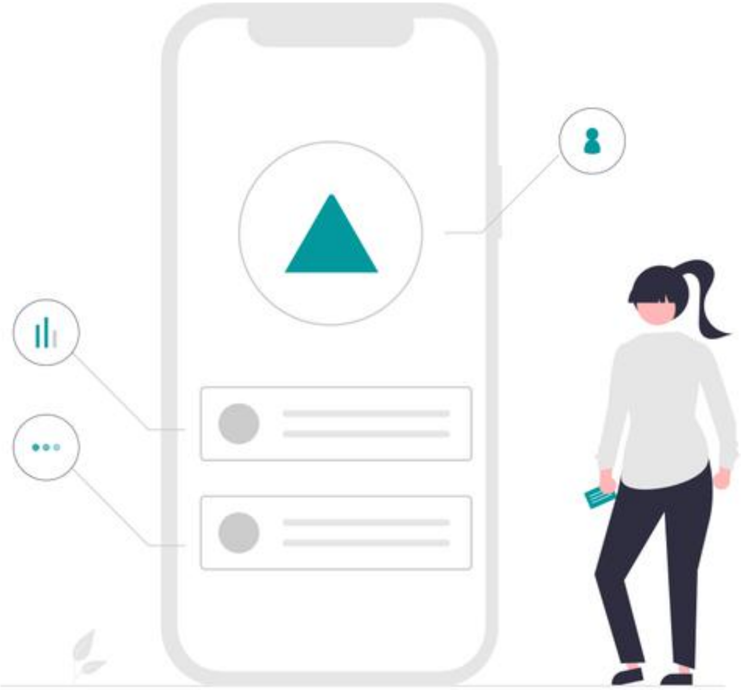
What is the definition of Big Data?

Information whose amount, complexity, and fast acquisition times are difficult to be processed and analyzed using traditional techniques and tools.



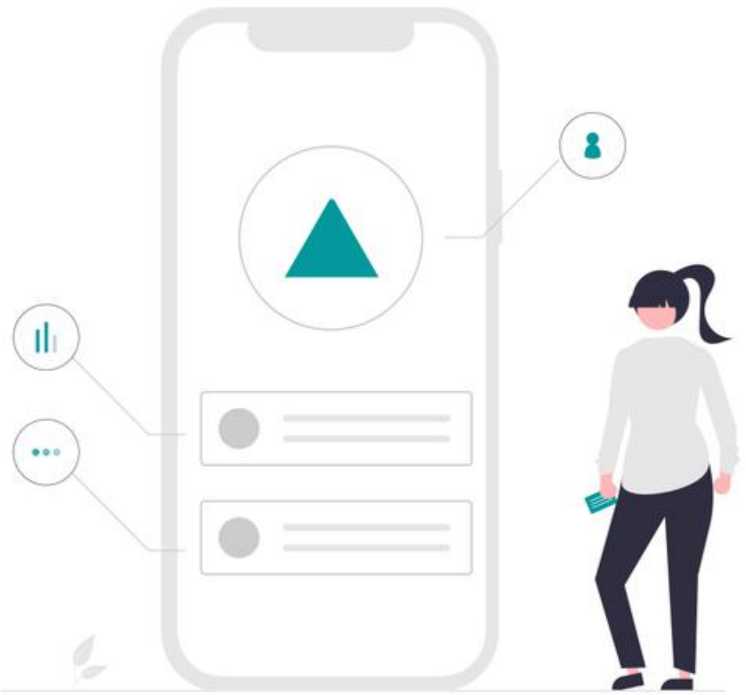
?

But why Big Data?



?

But why Big Data?



!

They help us find:

- Hidden patterns
- Hidden correlations
- Purchasing trends
- Consumer preferances

Leads to better **informed decision making** and **strategic moves**

What led us to Big Data?

Historical Evolution

2005

social media start to become popular

2010

5 billion cell phones => Realization of the huge amount of data that are generated

2015

Google and Microsoft construct big data centers

2021

Data centers speeds surpass 1000G

2022

Google serves 40.000 searches per second (3.5 billion per day)

What led us to Big Data?

Fun Fact !

Facebook, Twitter, LinkedIn, Instagram etc generated around **2.5 million Terrabytes of data daily**

Historical Evolution

2005

social media start to become popular

2010

5 billion cell phones => Realization of the huge amount of data that are generated

2015

Google and Microsoft construct big data centers

2021

Data centers speeds surpass 1000G

2022

Google serves 40.000 searches per second (3.5 billion per day)

What are Data Centers?

What are Data Centers?

A company's infrastructure is used for all IT needs of an organization, meaning **storing**, **processing** and data **analysis**.

Their security and reliability are the organizations' primary target

Statistics

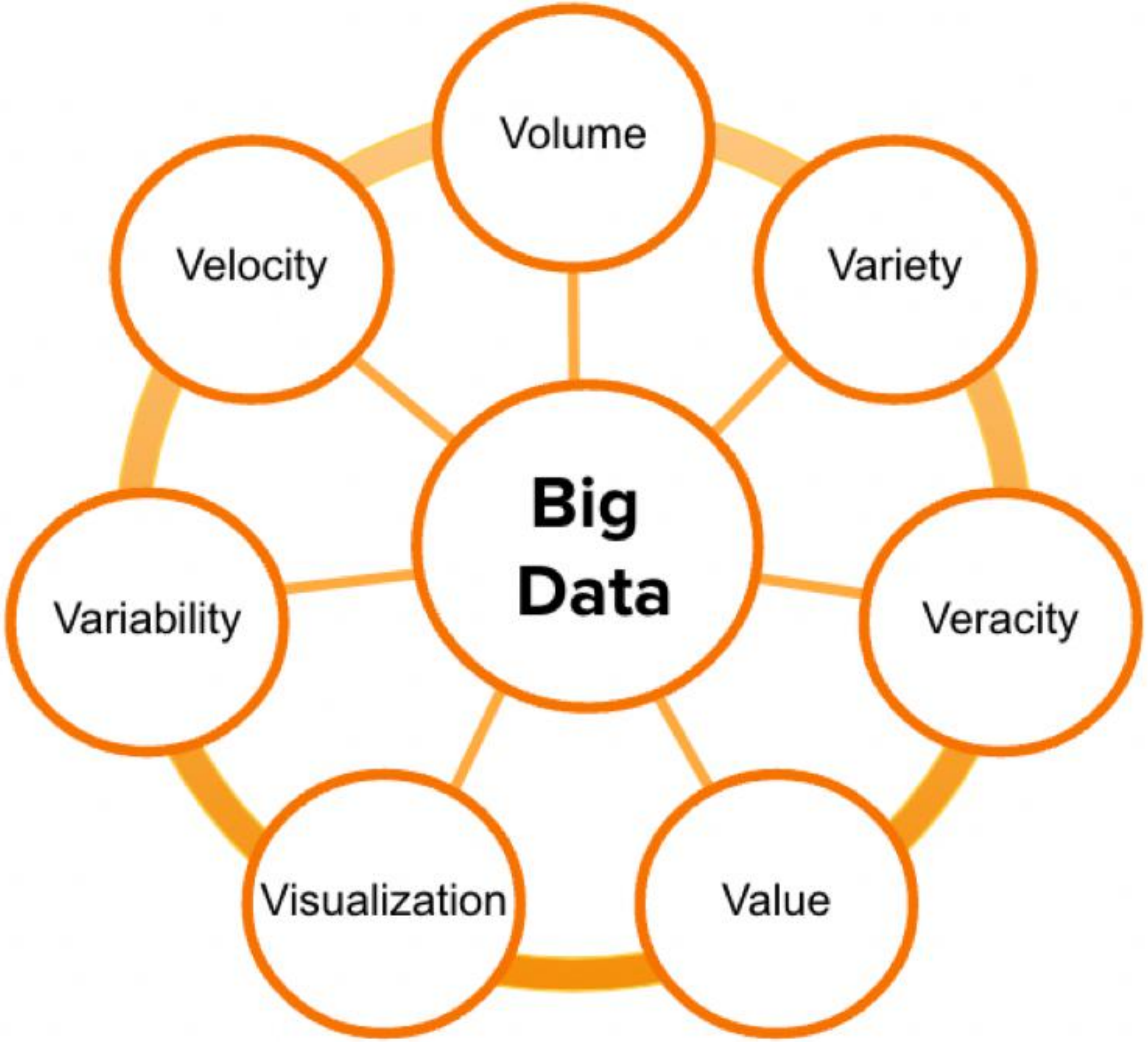
Every second:

3 million emails, of which 67% are spam
Each person produces 1.7MB
300 hours of video are uploaded on YouTube

Every minute:

1.4 million calls are made
350.000 stories are uploaded on Instagram
We watch 400.000 hours on Netflix
Amazon sends 6.500 packets

7 V'S OF BIG DATA



7 V's of Big Data

1

VOLUME

- Amount of data
- Used to be in Gigabytes (GB), now in Yottabytes (YB) or even Zettabytes (ZB)
- A huge increase in the amount of generated data is expected

2

3

4

7 V's of Big Data

1

VOLUME

- Amount of data
- Used to be in Gigabytes (GB), now in Yottabytes (YB) or even Zettabytes (ZB)
- A huge increase in the amount of generated data is expected

2

VELOCITY

- How fast data are processed and become available
- Today, if they are not real-time, it is considered slow

3

4

7 V's of Big Data

1
VOLUME

- Amount of data
- Used to be in Gigabytes (GB), now in Yottabytes (YB) or even Zettabytes (ZB)
- A huge increase in the amount of generated data is expected

2
VELOCITY

- How fast data are processed and become available
- Today, if they are not real-time, it is considered slow

3
VARIETY

- One of the greatest challenges
- Various data types and structures
- Difficult and important to organize them

4

7 V's of Big Data

1

VOLUME

- Amount of data
- Used to be in Gigabytes (GB), now in Yottabytes (YB) or even Zettabytes (ZB)
- A huge increase in the amount of generated data is expected

2

VELOCITY

- How fast data are processed and become available
- Today, if they are not real-time, it is considered slow

3

VARIETY

- One of the greatest challenges
- Various data types and structures
- Difficult and important to organize them

4

VARIABILITY

- Different than variability
- A coffee shop has 6 different coffee varieties. If each day you purchase the same one but it tastes different, is called variability.
- Affects the data homogeneity.

7 V's of Big Data

5

VERACITY

- Ensures data accuracy
- Helps tackle the "garbage in, garbage out" problem

6

7

7 V's of Big Data

5

VERACITY

- Ensures data accuracy
- Helps tackle the "garbage in, garbage out" problem

6

VISUALIZATION

- Plots and diagrams that are more helpful than reports full of numbersç

7

7 V's of Big Data

5 **VERACITY**

- Ensures data accuracy
- Helps tackle the "garbage in, garbage out" problem

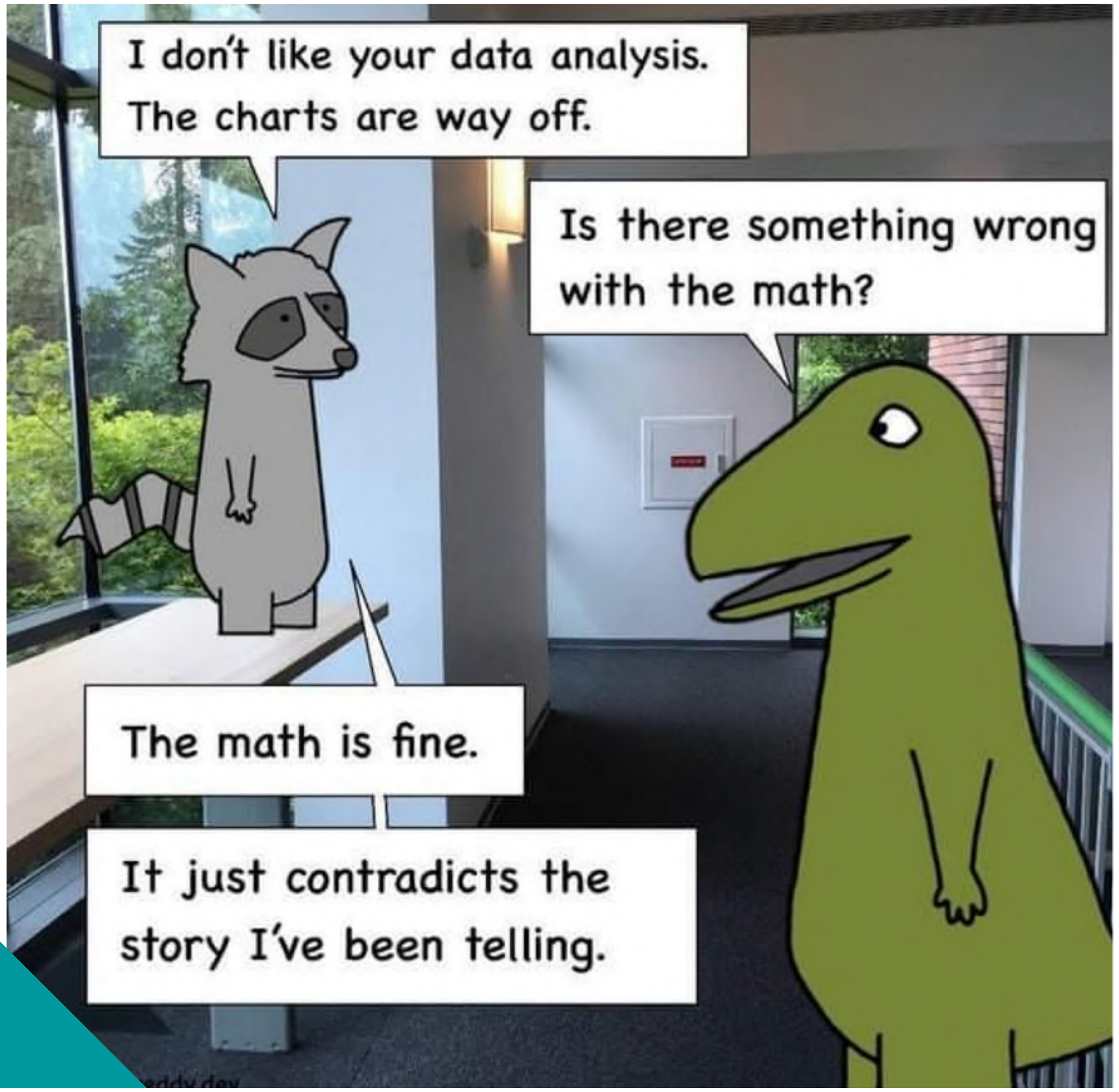
6 **VISUALIZATION**

- Plots and diagrams that are more helpful than reports full of numbers

7 **VALUE**

- Final purpose and target
- After all the above are realized, a value/profit should be the result of the data processing.

Big Data Processing

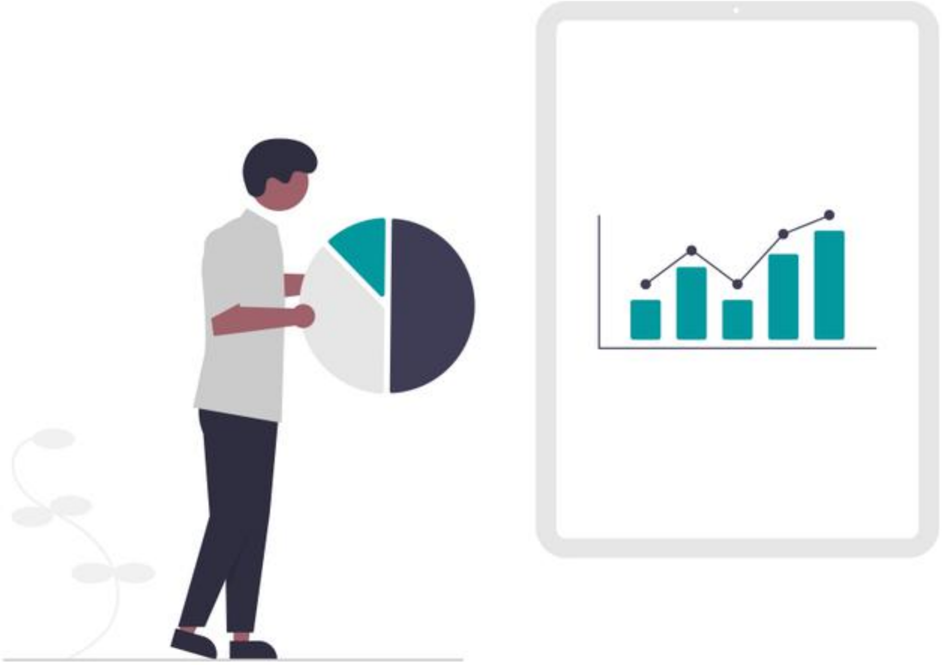

I don't like your data analysis. The charts are way off.
Is there something wrong with the math?
The math is fine.
It just contradicts the story I've been telling.

Step 1st – Collection

- Data **collection** for various courses
- **Elimination** of false data
- Proper labels and **categorization**
- The basic step for further proper processing

2nd Step – Conversion

- **Convert the data type** e.g. clustering
- **Normalization**
- **Convert** from unstructured to structured

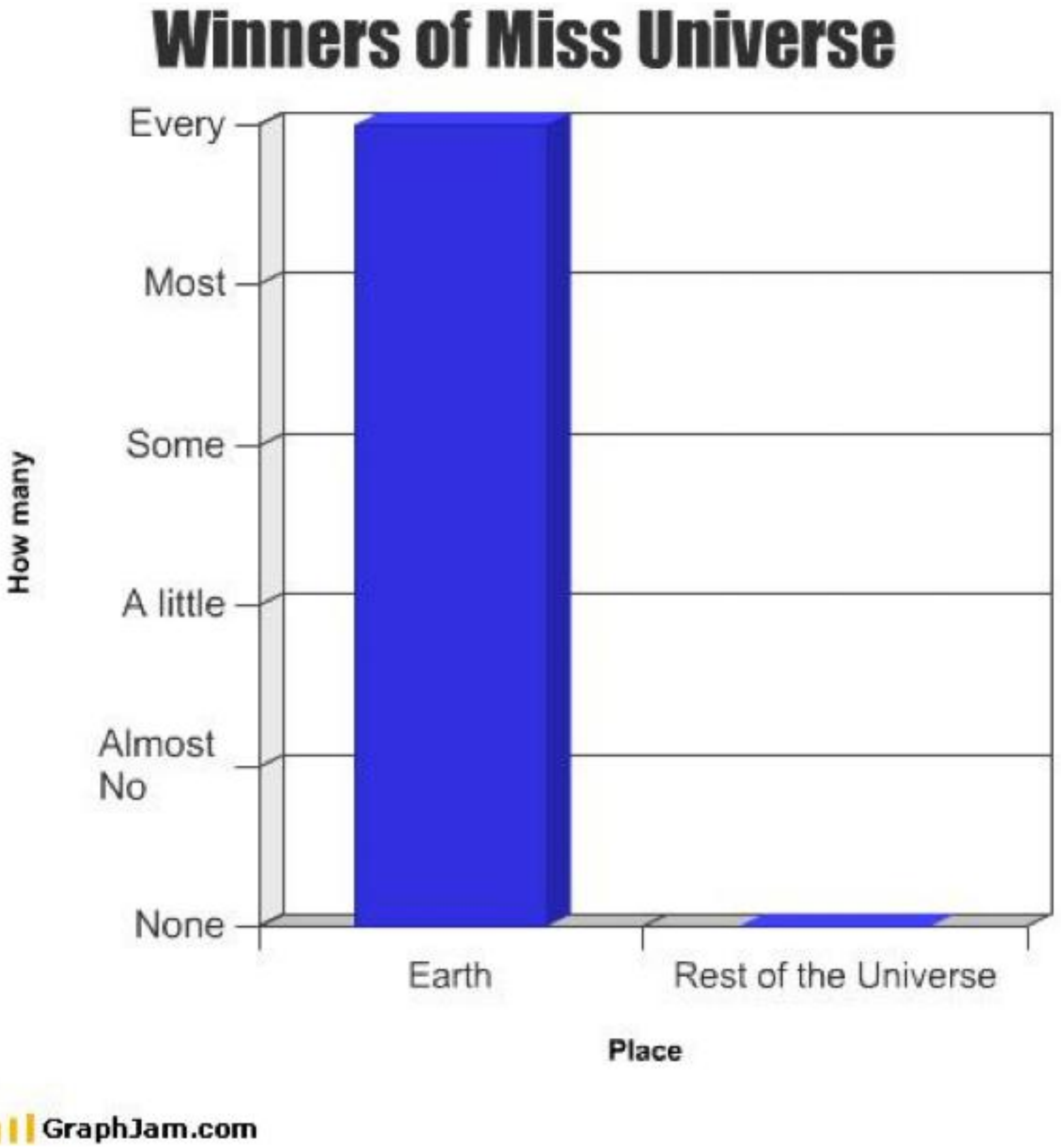


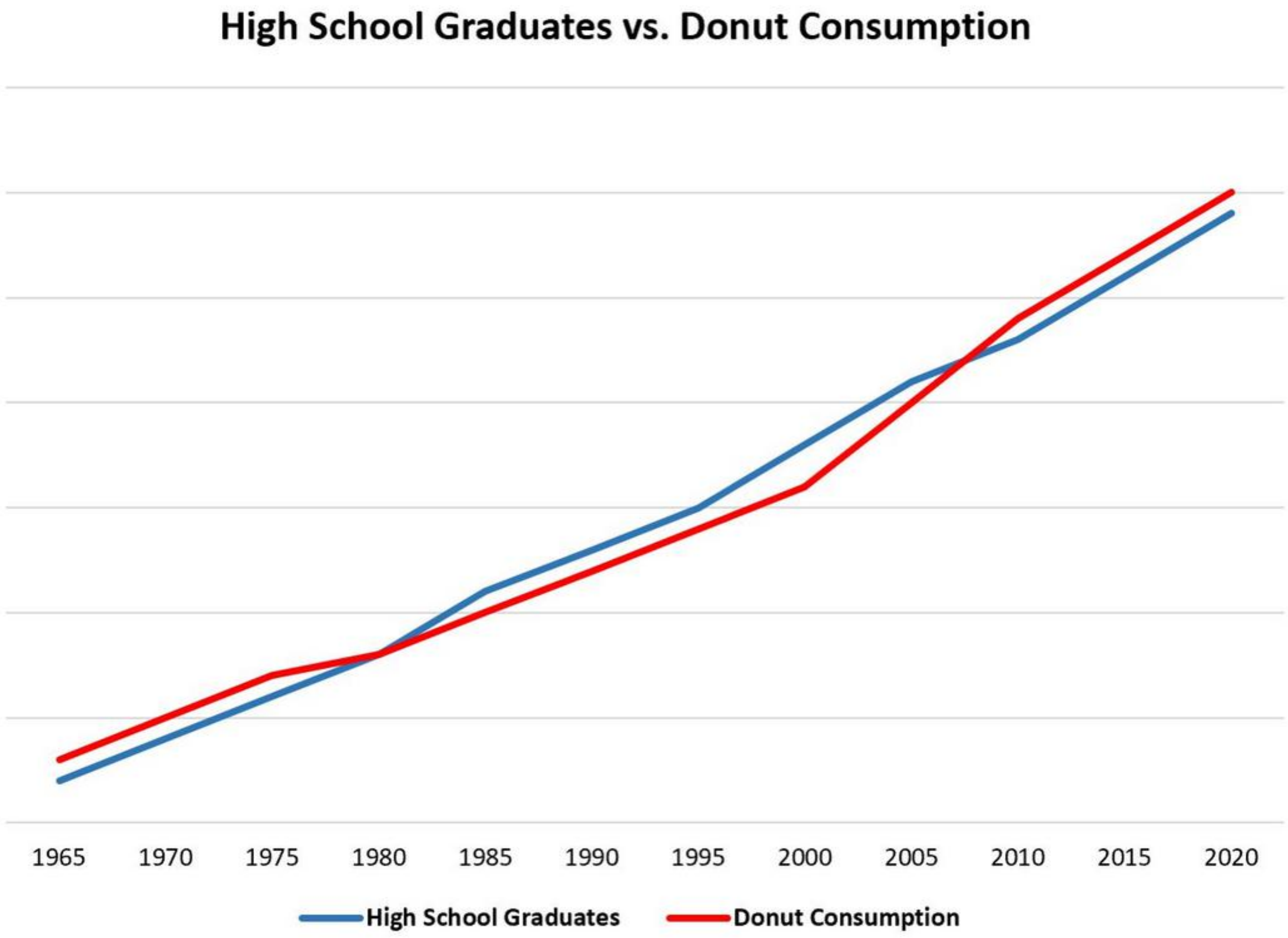
3rd Step – Loasd

"**Upload**" data to the main database

I II III IV

4th Step – Visualization and Analytics





I II III IV V

5th Step – Application of Machine Learning

!

Machine learning is a subtotal of **Artificial Intelligence (AI)**

Computers are taught to learn from data and **improve through experience** – instead of being explicitly programmed to do it.

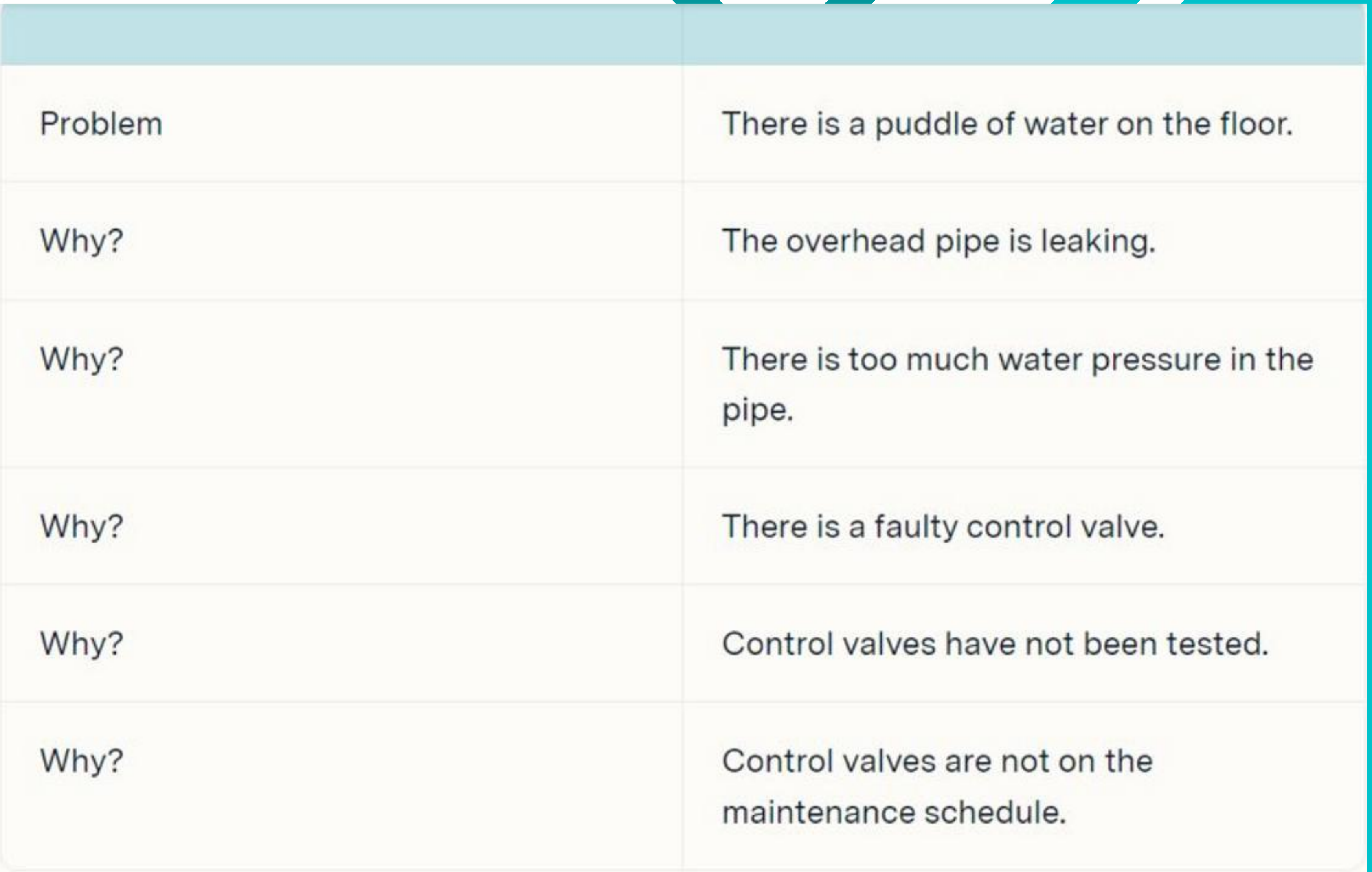
- Create **models** that evolve with new inputs
- **Learn** from the data
- Finds **patterns** and makes evolution predictions with no human interference

What is the "favorite" application of those that work on Machine Learning?

The 5 "Why's" method

?

- Tool to analyze the **causes of a fact**
- Enables locating the causes of a problem by successively asking "**Why?**"
- Creates a **cause-outcome chain** that leads to the initial cause
- Developed by Sakichi Toyoda, founder of **Toyota** and it is still **the basis of its scientific approach**



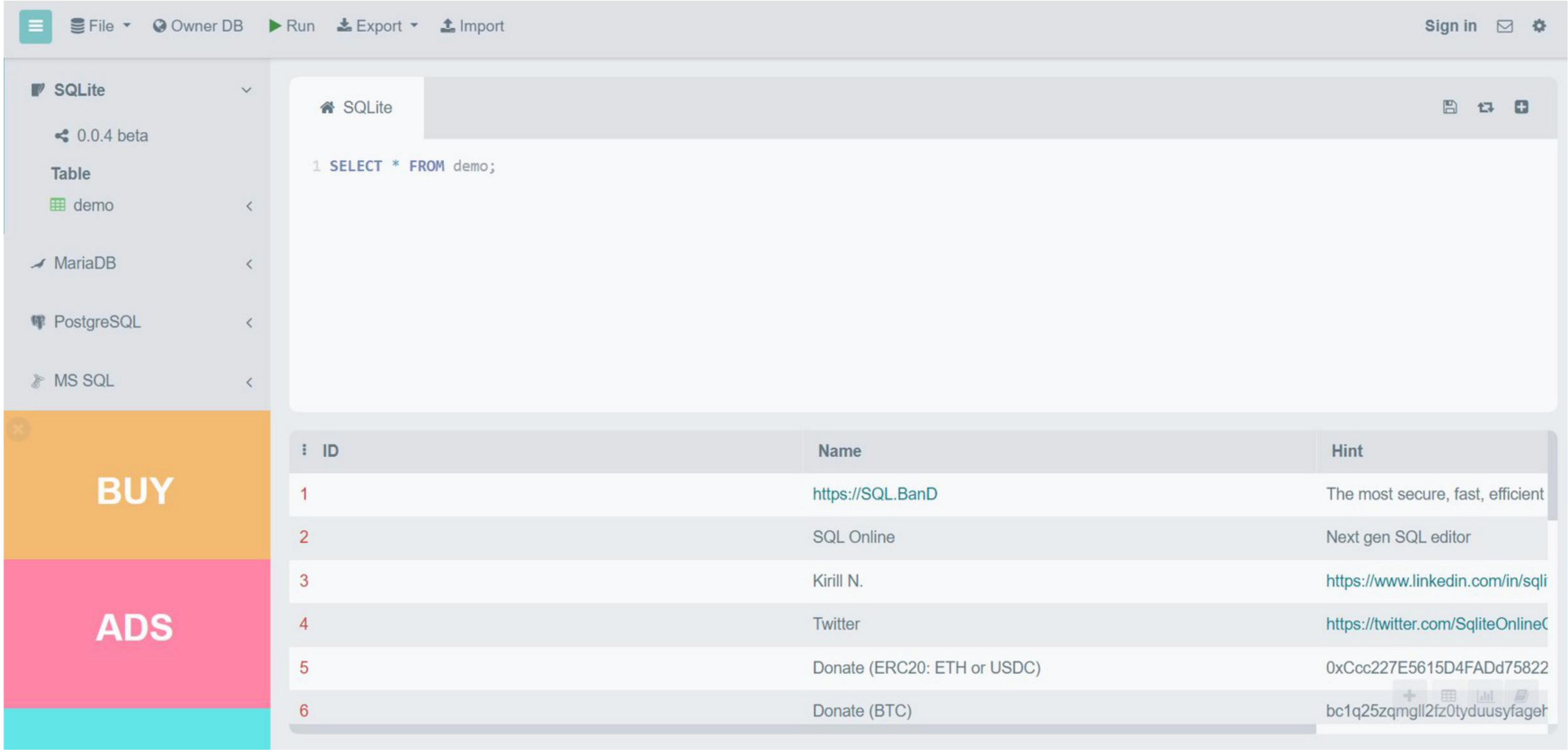
Problem	There is a puddle of water on the floor.
Why?	The overhead pipe is leaking.
Why?	There is too much water pressure in the pipe.
Why?	There is a faulty control valve.
Why?	Control valves have not been tested.
Why?	Control valves are not on the maintenance schedule.

Introduction to SQL

Structured Query Language

- **Programming language** to manage data in a relative database management system.
- SQL includes **data retrieval and update capabilities**, plot and relative matrices creation and modification but also **access control** to the data.
- SQL was developed at **IBM** by Andrew Richardson, Donald C. Messerly, and Raymond F. Boyce, in the early **1970's**.

SQLITEONLINE

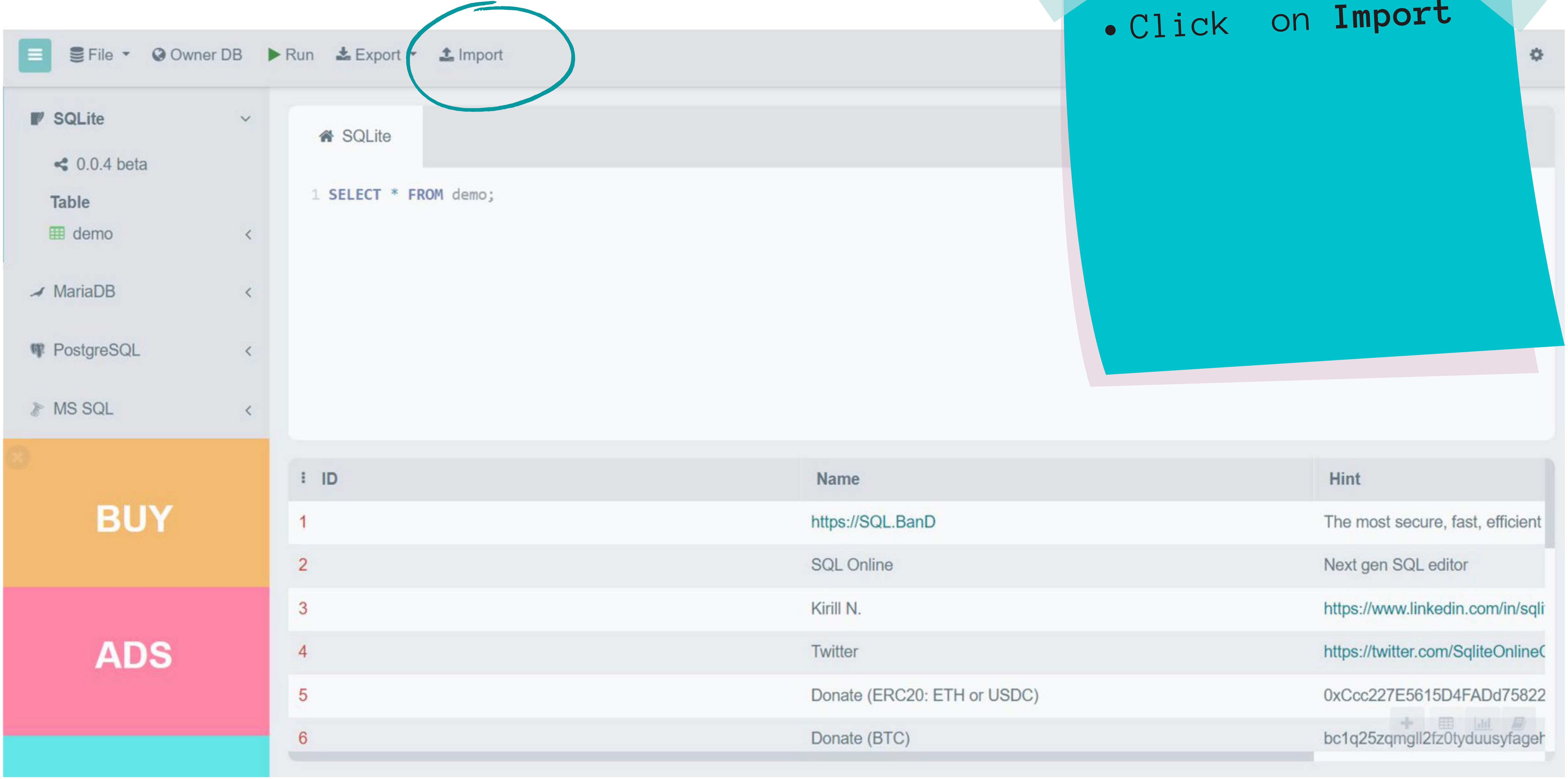


```
SELECT * FROM demo;
```

ID	Name	Hint
1	https://SQL.BanD	The most secure, fast, efficient
2	SQL Online	Next gen SQL editor
3	Kirill N.	https://www.linkedin.com/in/sqli
4	Twitter	https://twitter.com/SqliteOnline(
5	Donate (ERC20: ETH or USDC)	0xCcc227E5615D4FADd75822
6	Donate (BTC)	bc1q25zqmgll2fz0tyduusyfageh

SQLITEONLINE

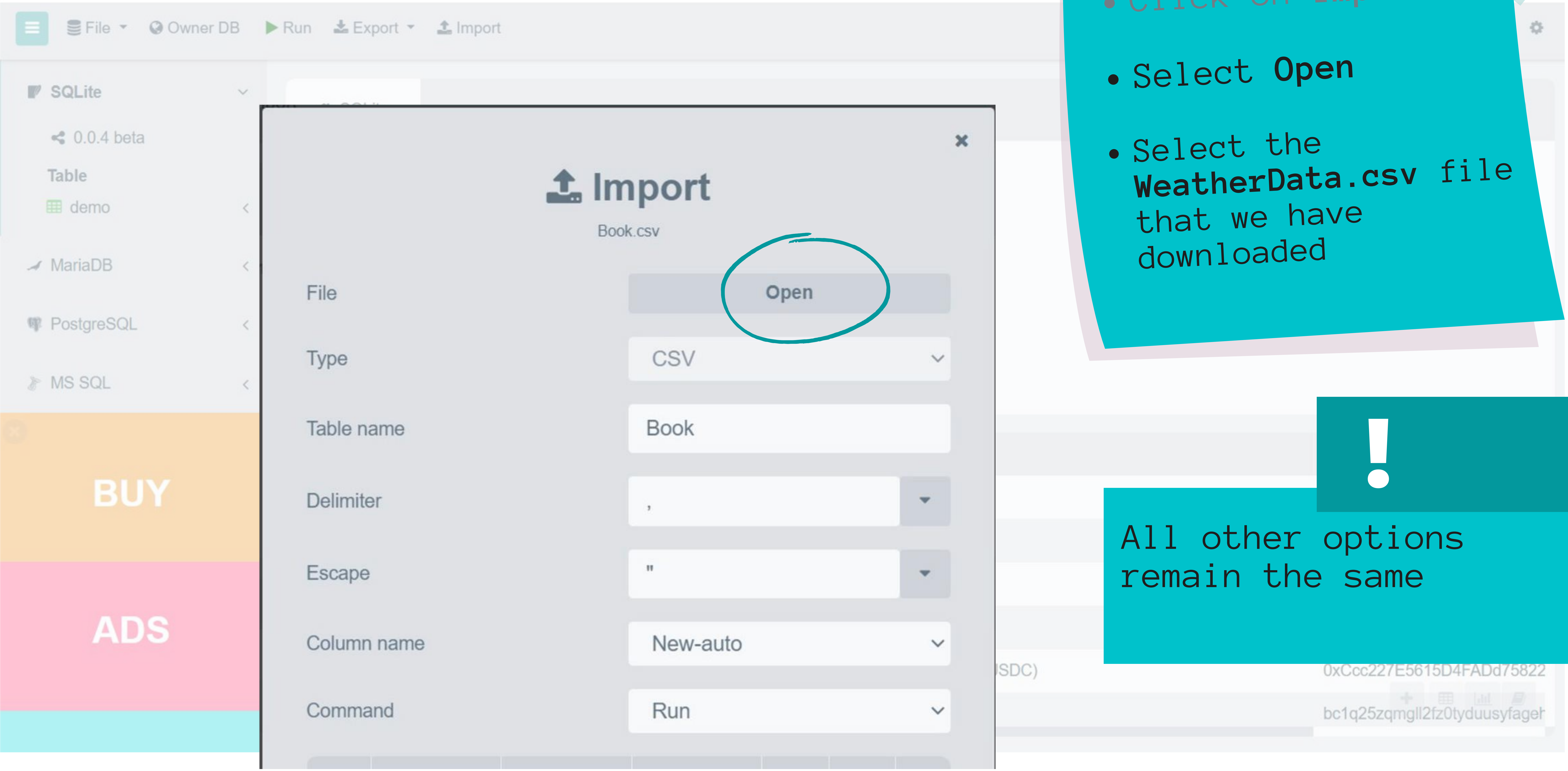
- Click on **Import**



```
SELECT * FROM demo;
```

ID	Name	Hint
1	https://SQL.BanD	The most secure, fast, efficient
2	SQL Online	Next gen SQL editor
3	Kirill N.	https://www.linkedin.com/in/sqli
4	Twitter	https://twitter.com/SqliteOnline(
5	Donate (ERC20: ETH or USDC)	0xCcc227E5615D4FADd75822
6	Donate (BTC)	bc1q25zqmgll2fz0tyduusyfageh

SQLITEONLINE



Import

Book.csv

File: Open

Type: CSV

Table name: Book

Delimiter: ,

Escape: "

Column name: New-auto

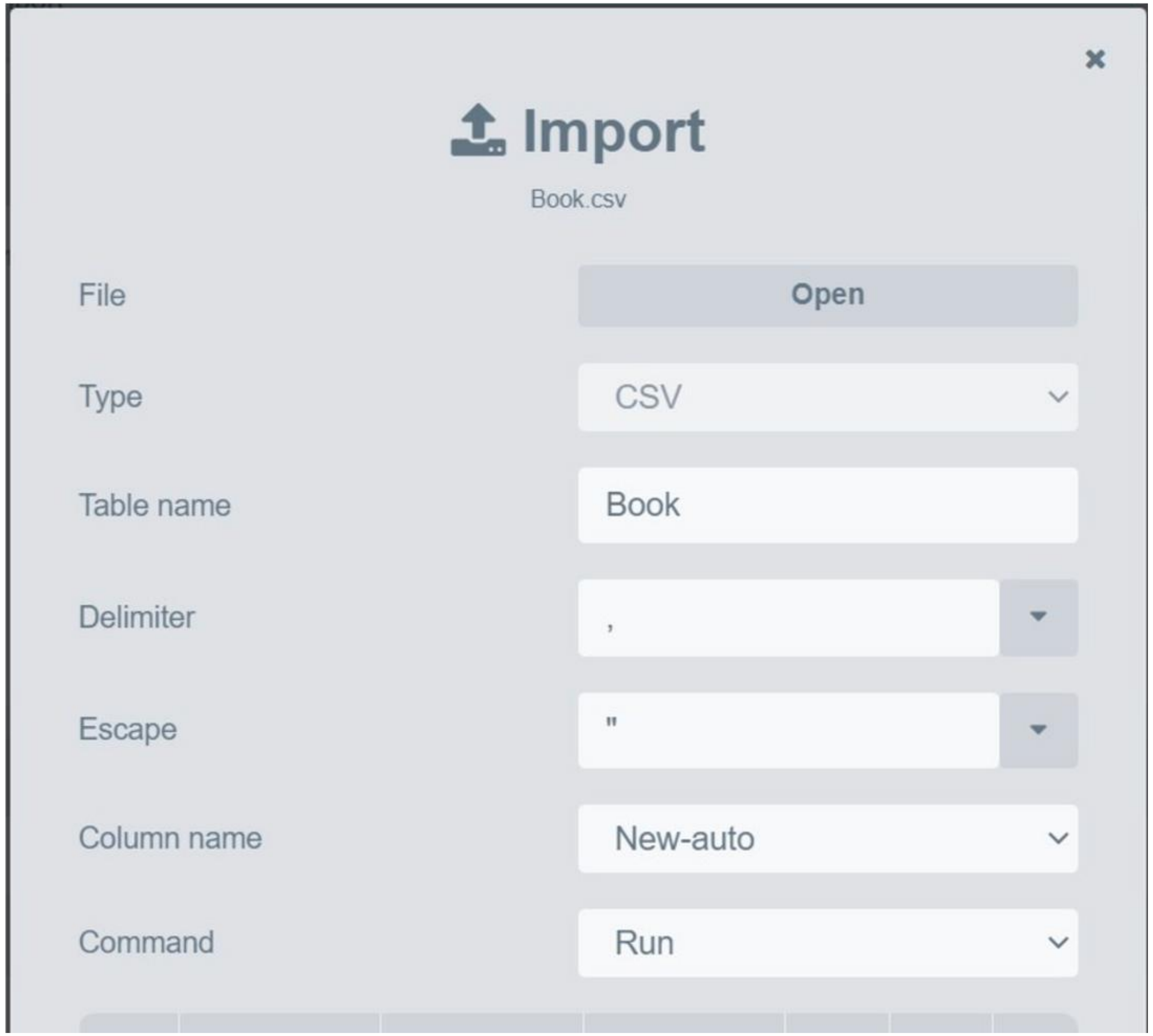
Command: Run

- Click on **Import**
- Select **Open**
- Select the **WeatherData.csv** file that we have downloaded

!

All other options remain the same

SQLITEONLINE



Import

Book.csv

File: Open

Type: CSV

Table name: Book

Delimiter: ,

Escape: "

Column name: New-auto

Command: Run

- Name the database as we wish e.g. MyDB
- Select the 2nd option **First Line**

All other options remain the same

SELECT...FROM

How do I read data from the data base?

!

SELECT...FROM

How do I read data from the data base?

• • • • • • • • • • • • • • •

SELECT the column(s) from which to extract the data.

FROM which matrix the column(s) will be selected from. The column(s) should be part of the matrix.

SELECT...FROM

How do I read data from the data base?

```
SELECT *
FROM WeatherDataThessaloniki;
```

LIMIT

If we only want to process the first few line of the matrix.
Faster than loading the entire dataset.
It is **ALWAYS** the last instruction.

Examples for the first 10 lines of a matrix:

```
SELECT *
FROM WeatherDataThessaloniki
LIMIT 10;
```

ORDER BY - 1

Order the results using one column's data.

It has **temporary action**, in contrast to e.g. sort in Excel.

Therefore, for the next search (query), data will be **unordered**.

Always after SELECT and FROM but before LIMIT.

ORDER BY - 1

```
SELECT *
FROM WeatherDataThessaloniki
ORDER BY Max_Temperature
LIMIT 10;
```

ORDER BY - 1

Pro tip

*Including **DESC** after the column in the ORDER BY instruction, classifies the data in a **descending** order. Increasing sorting is the default option.*

```
SELECT *
FROM WeatherDataThessaloniki
ORDER BY Max_Temperature
LIMIT 10;
```

ORDER BY - 2

When the **ORDER BY** instruction includes more than one columns, sorting is initially done according to the left-most column, then the one next to it, etc.

This can be reversed using **DESC**.

```
SELECT *
FROM WeatherDataThessaloniki
ORDER BY Max_Temperature DESC
LIMIT 10;
```

WHERE

Common symbol used in a **WHERE** clause:

- > (greater than)
- < (less than)
- >= (greater or equal to)
- <= (less or equal to)
- = (equal to)
- != (different than)

WHERE

> (greater than)
< (less than)
>= (greater or equal to)
<= (less or euqal to)
= (equal to)
!= (different than)

```
SELECT *
FROM WeatherDataThessaloniki
WHERE Avg_Temperature > 20
ORDER BY Max_Temperature
LIMIT 100;
```

ARITHEMETIC OPERATORS

Creating a new column as a combination of other columns is known as **the computed or the produced column**.

Usually assigned a name to it using the committed work **AS**.

It is **temporary** and no longer exists in the next query.

If the new column has been produced using some **mathematical expression**:

- * (Multiplication)
- + (Addition)
- – (Subtraction)
- / (Division)

ARITHEMETIC OPERATORS

Creating a new column as a combination of other columns is known as **the computed or the produced column**.

Usually assigned a name to it using the committed work **AS**.

It is **temporary** and no longer exists in the next query.

If the new column has been produced using some **mathematical expression**:

- * (Multiplication)
- + (Addition)
- – (Subtraction)
- / (Division)

```
SELECT Date, ((Max_Temperature +
Min_Temperature)/2) AS Avg_Temperature
FROM WeatherDataThessaloniki
LIMIT 50;
```

LOGICAL OPERATORS

LIKE

Useful when we process **text files**.

Used in a **WHERE** clause.

Often used with the **%** symbol.

The % indicates that we might want any amount of numbers or characters till we locate a certain piece or the part after it.

Single or double quotation marks
for the case of characters as **'T'**
is not the same as **'t'**.

LOGICAL OPERATORS

LIKE

Useful when we process **text files**.

Used in a **WHERE** clause.

Often used with the % symbol.

The % indicates that we might want any amount of numbers or characters till we locate a certain piece or the part after it.

Single or double quotation marks for the case of characters as **'T'** is not the same as **'t'**.

```
SELECT *
FROM WeatherDataThessaloniki
WHERE Weather_Description LIKE
'%cold%';
```

LOGICAL OPERATORS

IN

Useful where columns create both numbers and characters.

Allows the use of = but for more than one object of a particular column.

They can control one, two, or more values of a column.

Pro tip

Single or double quotation marks for the case of characters as 'T' is not the same as 't'.
Double if there is an apostrophe in the text.

```
                             SELECT *
         FROM WeatherDataThessaloniki
WHERE Weather_Description IN (30,35);
```

LOGICAL OPERATORS

NOT

Used wit the previous tow operators **IN** and **LIKE**.

Using **NOT LIKE** or **NOT IN**, we can extract all the columns that do not fit a certain criterion.

LOGICAL OPERATORS

NOT

Used wit the previous tow operators **IN** and **LIKE**.

Using **NOT LIKE** or **NOT IN**, we can extract all the columns that do not fit a certain criterion.

```
SELECT *
FROM WeatherDataThessaloniki
WHERE Weather_Description NOT IN
(30,35)
ORDER BY Date
```

LOGICAL OPERATORS

AND & BETWEEN

The **AND** operator is used in a **WHERE** clause to consider more than one logical factors.

The **column** that we are interested in should be mentioned.
We can connect as many clauses as we want.

It can be **combined** with all operators that we have seen so far (logical and arithmetic).

LIKE, **IN** and **NOT** µcan also be connected using the AND operator.

LOGICAL OPERATORS

AND & BETWEEN

The **AND** operator is used in a **WHERE** clause to consider more than one logical factors.

The **column** that we are interested in should be mentioned.
We can connect as many clauses as we want.

It can be **combined** with all operators that we have seen so far (logical and arithmetic).

LIKE, **IN** and **NOT** can also be connected using the AND operator.

```
SELECT *
FROM WeatherDataThessaloniki
WHERE Avg_Humidity >= 80 AND
Avg_Humidity <= 100
ORDER BY Date
```

LOGICAL OPERATORS

AND & BETWEEN

When the same column is used for different parts of the AND statement, the BETWEEN statement helps make a more "presentable" statement.

For example, instead of:
WHERE column >= 6 AND column <= 10

We can write:
WHERE column BETWEEN 6 AND 10

LOGICAL OPERATORS

AND & BETWEEN

When the same column is used for different parts of the AND statement, the BETWEEN statement helps make a more "presentable" statement.

For example, instead of::
WHERE column >= 6 AND column <= 10

We can write:
WHERE column BETWEEN 6 AND 10

```
                                   SELECT *
           FROM WeatherDataThessaloniki
WHERE Avg_Humidity BETWEEN 80 AND 90
                              ORDER BY Date
```

LOGICAL OPERATORS

OR

The OR operator can combine multiple statements.

The column we want to access needs to be mentioned.

We can combine as many statements as we want.

It can be combined with all the operators we have seen so far (logical and arithmetic).

LIKE, IN, NOT, AND, and BETWEEN can also be connected using the OR operator.

LOGICAL OPERATORS

OR

The OR operator can combine multiple statements.

The column we want to access needs to be mentioned.

We can combine as many statements as we want.

It can be combined with all the operators we have seen so far (logical and arithmetic).

LIKE, IN, NOT, AND, and BETWEEN can also be connected using the OR operator.

```
                  SELECT Date,
              Max_Temperature,
                 Avg_Humidity,
               Avg_Wind_Speed,
                 Avg_Pressure
 FROM WeatherDataThessaloniki
WHERE Avg_Temperature > 30 OR
         Avg_Humidity >80 OR
         Avg_Wind_Speed > 20
```

Pro tip

When combining multiple such statements, it's good to use parentheses!

LOGICAL OPERATORS

COUNT & NULL

NULL is a data type that shows there are no data.

In addition functions they are usually ignored.

Count the number of rows in a matrix.

LOGICAL OPERATORS

COUNT & NULL

NULL is a data type that shows there are no data.

In aggregation functions they are usually ignored.

Count the number of rows in a matrix.

✱

```
SELECT COUNT(*)
FROM WeatherDataThessaloniki;
```

SUM

Instead of COUNT, SUM can only be used in arithmetic data.

It will ignore the NULL values.

Aggregation Reminder

Aggregations are only performed vertically - values of one column.

Aggregating in a specific row can be performed arithmetically.

LOGICAL OPERATORS

SUM

Instead of COUNT, SUM can only be used in arithmetic data.

It will ignore the NULL values.

Aggregation Reminder
Aggregations are only performed vertically - values of one column.

Aggregating in a specific row can be performed arithmetically.

```
SELECT SUM(Avg_Temperature)/COUNT(*)
                        AS avg_temp,
    SUM(Avg_Wind_Speed)/COUNT(*) AS
                           avg_wind,
      SUM(Avg_Pressure)/COUNT(*) AS
                           avg_press
        FROM WeatherDataThessaloniki
```

LOGICAL OPERATORS

MIN & MAX

MIN and MAX also ignore NULL values.

LOGICAL OPERATORS

AVG

Returns the **average** of all the data, (sum of all data in a column divided by their number).

Also ignores **NULL** values both at the nominator as well as at the denominator.

If we want to consider **NULLs** as zeros, SUM, and COUNT functions should be used.

This is not a good idea if NULL values represent unknown values for our data.

Pro tip

The median metric may be more appropriate to find the central value of the data but it is more difficult to compute.

```
SELECT AVG(Avg_Temperature) AS
                 temperature_avg,
            AVG(Avg_Humidity) AS
                   humidity_avg,
AVG(Avg_Wind_Speed) AS wind_avg
 FROM WeatherDataThessaloniki
```

MIN & MAX

MIN and MAX also ignore NULL values.

Pro tip

Work similarly with COUNT as they can also be used in columns with non-arithmetic data.

Depending on the column type, *MIN will return the smallest number, the furthest date or the character closest to "A".*

MAX does the opposite. It returns the greatest number, the closest date or the character closest to "Z".

```
SELECT MIN(Avg_Temperature) AS
                       avg_temp_min,
       MIN(Min_Temperature) AS
                       min_temp_min,
          MIN(Min_Humidity) AS
                   min_humidity_min,
       MAX(Max_Temperature) AS
                       max_temp_max,
          MAX(Max_Humidity) AS
                        max_hum_max,
MAX(Max_Wind_Speed) AS wind_max
  FROM   WeatherDataThessaloniki
```

AVG

Returns the **average** of all the data, (sum of all data in a column divided by their number).

Also ignores **NULL** values both at the nominator as well as at the denominator.

If we want to consider **NULLs** as zeros, SUM, and COUNT functions should be used.

This is not a good idea if NULL values represent unknown values for our data.

LOGICAL OPERATORS

GROUP BY

GROUP BY command can be used to aggregate data in a **sub-total**.

For example, calculate the average temperature for one month.

If a column is not included in the aggregation, it should be included in a **GROUP BY** statement.

Always between the **WHERE** and **ORDER BY**.

```
   SELECT Max_Temperature,
                 COUNT(*)
FROM WeatherDataThessaloniki
    GROUP BY Max_Temperature
    ORDER BY Max_Temperature
```

LOGICAL OPERATORS

GROUP BY

We can use it in **multiple columns** at the same time.

The order of the columns in the **ORDER BY** statement affects their ranking. It is from left to right.

```
SELECT Max_Temperature,
MAX(Max_Humidity) AS Hum
FROM WeatherDataThessaloniki
GROUP BY Max_Temperature
ORDER BY Max_Temperature, Hum
```

LOGICAL OPERATORS

DECLARATION OF CASE

Always before the **SELECT** statement.

It must include: **WHEN**, **THEN**, and **END**. ELSE statement is optional.

We can perform any logical checj between the **WHEN** and **THEN**. For example, mulriple **AND** and **OR**.

We can use multiple **WHEN** statements as well as an **ELSE** statement for the unwanted cases.

LOGICAL OPERATORS

DECLARATION OF CASE

Always before the **SELECT** statement.

It must include: **WHEN**, **THEN**, and **END**. ELSE statement is optional.

We can perform any logical checj between the **WHEN** and **THEN**. For example, mulriple **AND** and **OR**.

We can use multiple **WHEN** statements as well as an **ELSE** statement for the unwanted cases.

```
SELECT date,
max_temperature
max_humidity,
CASE WHEN max_temperature > 30
AND max_humidity > 80 THEN 'Too hot
weather!'
WHEN max_temperature < 10 AND
max_humidity > 80 THEN 'Too cold
Weather!'
END AS total_group
FROM WeatherDataThessaloniki
```

Examples

Considering the WeatherData matrix we have seen so far, we are going to write an SQL code where:

- We want to print the 10 days with the highest temperature, highest humidity, and lowest wind speed in 2022 in Thessaloniki.

- We want to compute the number of days where the minimum temperature in Thessaloniki was lower than 0oC and the wind was very strong.

- We want to print the days when it was hot or very hot (regardless of humidity or wind) only during the summer months (01/06/2022 to 31/08/2022) and rank these days initially in descending order using their maximum temperature. If some days have the same maximum temperature, they should be ranked in descending order according to the humidity.

Examples – Solutions

```
SELECT date,
       max_temperature,
       max_humidity,
       min_wind_speed
FROM WeatherDataThessaloniki
ORDER BY max_temperature DESC, max_humidity DESC, min_wind_speed
LIMIT 10
```

```
SELECT COUNT(date) AS num_of_very_cold_days
FROM WeatherDataThessaloniki
WHERE min_temperature <= 0 AND max_wind_speed > 20
```

```
SELECT date,
       max_temperature,
       max_humidity,
       weather_description
FROM WeatherDataThessaloniki
WHERE ((date like '%-06-%') OR (date like '%-07-%') OR (date like
'%-08-%')) AND weather_description LIKE 'hot%'
ORDER BY max_temperature DESC, max_humidity DESC
```

Data Visualization

SQL allows the user to visualize the data using charts.

```
LINE-SELECT Date, Avg_Temperature
FROM WeatherDataThessaloniki

BAR-SELECT Date, Avg_Temperature
FROM WeatherDataThessaloniki

AREA-SELECT Date, Avg_Temperature
FROM WeatherDataThessaloniki

LINE-SELECT Date,
            Avg_Temperature,
            Avg_Wind_Speed
FROM WeatherDataThessaloniki
```

QUESTIONS

SCIENTIX
The community for science education in Europe
CISCO
STE(A)M PARTNERSHIPS
Education Resilience in Europe
European Schoolnet
SCIENTIX