```
%load_ext sql
```

```
#connection to my database
%sql mysql+pymysql://root:123456789@localhost:3306/md_water_services
```

∨ 1. Get to Know Our Data

```
%%sql
SHOW TABLES;
```

⇥▾   * mysql+pymysql://root:***@localhost:3306/md_water_services
     8 rows affected.

Tables_in_md_water_services
data_dictionary
employee
global_water_access
location
visits
water_quality
water_source

```
%%sql

select*

from md_water_services.data_dictionary

limit 5;
```

⇥▾   * mysql+pymysql://root:***@localhost:3306/md_water_services
     5 rows affected.

table_name	column_name	description	datatype	related_to
employee	assigned_employee_id	Unique ID assigned to each employee	INT	visits
employee	employee_name	Name of the employee	VARCHAR(255)	
employee	phone_number	Contact number of the employee	VARCHAR(15)	
employee	email	Email address of the employee	VARCHAR(255)	
employee	address	Residential address of the employee	VARCHAR(255)	

```
%%sql

select*

from md_water_services.employee

limit 5;
```

⇥▾   * mysql+pymysql://root:***@localhost:3306/md_water_services
     5 rows affected.

assigned_employee_id	employee_name	phone_number	email	address	province_name	town_name	position
0	Amara Jengo	+99637993287	None	36 Pwani Mchangani Road	Sokoto	Ilanga	Field Surveyor
1	Bello Azibo	+99643864786	None	129 Ziwa La Kioo Road	Kilimani	Rural	Field Surveyor
2	Bakari Iniko	+99222599041	None	18 Mlima Tazama Avenue	Hawassa	Rural	Field Surveyor
3	Malachi Mavuso	+99945849900	None	100 Mogadishu Road	Akatsi	Lusaka	Field Surveyor

```
%%sql

select*

from md_water_services.location

limit 5;
```

⇥▾   * mysql+pymysql://root:***@localhost:3306/md_water_services
     5 rows affected.

location_id	address	province_name	town_name	location_type
AkHa00000	2 Addis Ababa Road	Akatsi	Harare	Urban
AkHa00001	10 Addis Ababa Road	Akatsi	Harare	Urban
AkHa00002	9 Addis Ababa Road	Akatsi	Harare	Urban
AkHa00003	139 Addis Ababa Road	Akatsi	Harare	Urban

```
%%sql

select*

from md_water_services.visits

limit 5;
```

> * mysql+pymysql://root:***@localhost:3306/md_water_services
>   5 rows affected.

record_id	location_id	source_id	time_of_record	visit_count	time_in_queue	assigned_employee_id
0	Soll32582	Soll32582224	2021-01-01 09:10:00	1	15	12
1	KiRu28935	KiRu28935224	2021-01-01 09:17:00	1	0	46
2	HaRu19752	HaRu19752224	2021-01-01 09:36:00	1	62	40
3	AkLu01628	AkLu01628224	2021-01-01 09:53:00	1	0	1

Yeah, so this is a list of location_id, source_id, record_id, and a date and time, so it makes sense that someone (assigned_employee_id) visited some location (location_id) at some time (time_of_record) and found a 'source' there (source_id). Often the "_id" columns are related to another table. In this case, the source_id in the visits table refers to source_id in the water_sourc table. This is what we call a foreign key, but we'll get more into this next time.

```
%%sql

select*

from md_water_services.water_source

limit 5;
```

> * mysql+pymysql://root:***@localhost:3306/md_water_services
>   5 rows affected.

source_id	type_of_water_source	number_of_people_served
AkHa00000224	tap_in_home	956
AkHa00001224	tap_in_home_broken	930
AkHa00002224	tap_in_home_broken	486
AkHa00003224	well	364

Unsupported Cell Type. Double-Click to inspect/edit the content.

∨ 2. Let's Dive into Water_Source table.

Unsupported Cell Type. Double-Click to inspect/edit the content.

```
%%sql

SELECT
    distinct(type_of_water_source)
FROM
    water_source;
```

> * mysql+pymysql://root:***@localhost:3306/md_water_services
>   5 rows affected.

type_of_water_source
tap_in_home
tap_in_home_broken
well
shared_tap
river

Unsupported Cell Type. Double-Click to inspect/edit the content.

```
%%sql
SELECT
    *
FROM
    md_water_services.visits
WHERE
    time_in_queue>500
LIMIT
    10;
```

```
 *  mysql+pymysql://root:***@localhost:3306/md_water_services
  10 rows affected.
```

record_id	location_id	source_id	time_of_record	visit_count	time_in_queue	assigned_employee_id
899	SoRu35083	SoRu35083224	2021-01-16 10:14:00	6	515	28
2304	SoKo33124	SoKo33124224	2021-02-06 07:53:00	5	512	16
2315	KiRu26095	KiRu26095224	2021-02-06 14:32:00	3	529	8
3206	SoRu38776	SoRu38776224	2021-02-20 15:03:00	5	509	46
3701	HaRu19601	HaRu19601224	2021-02-27 12:53:00	3	504	0
4154	SoRu38869	SoRu38869224	2021-03-06 10:44:00	2	533	24
5483	AmRu14089	AmRu14089224	2021-03-27 18:15:00	4	509	12
9177	SoRu37635	SoRu37635224	2021-05-22 18:48:00	2	515	1
9648	SoRu36096	SoRu36096224	2021-05-29 11:24:00	2	533	3
11631	AkKi00881	AkKi00881224	2021-06-26 06:15:00	6	502	32

Unsupported Cell Type. Double-Click to inspect/edit the content.

```
%%sql
SELECT
    *
FROM
    md_water_services.visits
LIMIT 5;
```

```
 *  mysql+pymysql://root:***@localhost:3306/md_water_services
  5 rows affected.
```

record_id	location_id	source_id	time_of_record	visit_count	time_in_queue	assigned_employee_id
0	SoIl32582	SoIl32582224	2021-01-01 09:10:00	1	15	12
1	KiRu28935	KiRu28935224	2021-01-01 09:17:00	1	0	46
2	HaRu19752	HaRu19752224	2021-01-01 09:36:00	1	62	40
3	AkLu01628	AkLu01628224	2021-01-01 09:53:00	1	0	1
4	AkRu03357	AkRu03357224	2021-01-01 10:11:00	1	28	14

Unsupported Cell Type. Double-Click to inspect/edit the content.

```
%%sql
SELECT
    *
FROM
    MD_WATER_SERVICES.WATER_SOURCE
WHERE
    SOURCE_ID IN ('AkRu05234224','HaZa21742224')
```

```
 *  mysql+pymysql://root:***@localhost:3306/md_water_services
  2 rows affected.
```

source_id	type_of_water_source	number_of_people_served
AkRu05234224	tap_in_home_broken	496
HaZa21742224	well	308

˅ ACCESS THE QUALITY OF THE WATER SOURCES.

```
%%sql
SELECT
    *
FROM
    md_water_services.water_quality
LIMIT
    10;
```

```
 *  mysql+pymysql://root:***@localhost:3306/md_water_services
  10 rows affected.
```

record_id	subjective_quality_score	visit_count
0	0	1
1	1	1
2	5	1
3	10	1
4	4	1
5	0	1
6	9	1
7	10	1
8	2	1
9	10	1

```
# Surveyers assigned a score to each source from 1, being terrible, to 10 for a good, clean water source in a home.
```

Unsupported Cell Type. Double-Click to inspect/edit the content.

```
# QUESTION
```

Unsupported Cell Type. Double-Click to inspect/edit the content.

Unsupported Cell Type. Double-Click to inspect/edit the content.

∨ Investigate pollution issues:

```
%%sql
SHOW TABLES;
```

⮯   * mysql+pymysql://root:***@localhost:3306/md_water_services
     8 rows affected.

Tables_in_md_water_services
data_dictionary
employee
global_water_access
location
visits
water_quality
water_source
well_pollution

```
%%sql
SELECT
    *
FROM
    md_water_services.well_pollution
LIMIT 10;
```

⮯   * mysql+pymysql://root:***@localhost:3306/md_water_services
     10 rows affected.

source_id	date	description	pollutant_ppm	biological	results
KiRu28935224	2021-01-04 09:17:00	Bacteria: Giardia Lamblia	0.0	495.898	Contaminated: Biological
AkLu01628224	2021-01-04 09:53:00	Bacteria: E. coli	0.0	6.09608	Contaminated: Biological
HaZa21742224	2021-01-04 10:37:00	Inorganic contaminants: Zinc, Zinc, Lead, Cadmium	2.715	0.0	Contaminated: Chemical
HaRu19725224	2021-01-04 11:04:00	Clean	0.0288593	9.56996e-05	Clean
SoRu35703224	2021-01-04 11:29:00	Bacteria: E. coli	0.0	22.5009	Contaminated: Biological
AkHa00070224	2021-01-04 11:42:00	Inorganic contaminants: Cadmium	5.46739	0.0	Contaminated: Chemical
HaSe21346224	2021-01-04 11:52:00	Clean	0.0140376	8.98989e-05	Clean
HaYa21468224	2021-01-04 12:03:00	Inorganic contaminants: Chromium, Barium, Chromium, Lead	6.05137	0.0	Contaminated: Chemical
SoRu36278224	2021-01-04 12:24:00	Parasite: Cryptosporidium	0.0	485.162	Contaminated: Biological
AkLu02155224	2021-01-04 12:29:00	Inorganic contaminants: Selenium, Arsenic	7.64106	0.0	Contaminated: Chemical

Unsupported Cell Type. Double-Click to inspect/edit the content.

Unsupported Cell Type. Double-Click to inspect/edit the content.

∨ QUESTIONS: write a query that checks if the results is Clean but the biological column is > 0.01.

```
%%sql
SELECT
    *
FROM
    md_water_services.well_pollution
WHERE
    results='clean'
AND
    biological>0.01;
```

⮯   * mysql+pymysql://root:***@localhost:3306/md_water_services
     0 rows affected.

source_id date description pollutant_ppm biological results

```
%%sql
SELECT
    COUNT(*)
FROM
    md_water_services.well_pollution
WHERE
```

```
    description LIKE 'clean_%'
```

> * mysql+pymysql://root:***@localhost:3306/md_water_services
> 1 rows affected.

COUNT(*)

0

These are the records that mistakenly have the word Clean in the description. And they Toa\tal to 38 records.

```
%%sql
SELECT
    *
FROM
    md_water_services.well_pollution
WHERE
    description LIKE 'clean_%'
```

> * mysql+pymysql://root:***@localhost:3306/md_water_services
> 0 rows affected.

source_id date description pollutant_ppm biological results

Unsupported Cell Type. Double-Click to inspect/edit the content.

```
%%sql
CREATE TABLE
    md_water_services.well_pollution_copy
AS (
SELECT
*
FROM
md_water_services.well_pollution
);
```

> * mysql+pymysql://root:***@localhost:3306/md_water_services
> 17383 rows affected.
> []

```
%%sql
UPDATE
    well_pollution_copy
SET
    description = 'Bacteria: E. coli'
WHERE
    description = 'Clean Bacteria: E. coli';
UPDATE
    well_pollution_copy
SET
    description = 'Bacteria: Giardia Lamblia'
WHERE
    description = 'Clean Bacteria: Giardia Lamblia';
UPDATE
    well_pollution_copy
SET
    results = 'Contaminated: Biological'
WHERE
    biological > 0.01 AND results = 'Clean';
```

> * mysql+pymysql://root:***@localhost:3306/md_water_services
> 0 rows affected.
> 0 rows affected.
> 0 rows affected.
> []

```
%%sql
SELECT
    *
FROM
    well_pollution_copy
WHERE
    description LIKE "Clean_%"
OR (results = "Clean" AND biological > 0.01)
;
```

```
 *  mysql+pymysql://root:***@localhost:3306/md_water_services
    0 rows affected.
```

source_id date description pollutant_ppm biological results

```python
# Then if we're sure it works as intended, we can change the table back to the well_pollution and delete the well_pollution_copy table.
```

```
%%sql

UPDATE
    well_pollution
SET
    description = 'Bacteria: E. coli'
WHERE
    description = 'Clean Bacteria: E. coli';
UPDATE
    well_pollution
SET
    description = 'Bacteria: Giardia Lamblia'
WHERE
    description = 'Clean Bacteria: Giardia Lamblia';
UPDATE
    well_pollution
SET
    results = 'Contaminated: Biological'
WHERE
    biological > 0.01 AND results = 'Clean';

DROP TABLE
    md_water_services.well_pollution_copy;
```

```
 *  mysql+pymysql://root:***@localhost:3306/md_water_services
    0 rows affected.
    0 rows affected.
    0 rows affected.
    0 rows affected.
    []
```

```
%%sql
SELECT
    *
FROM
    well_pollution
WHERE
    description LIKE "Clean_%"
;
```

```
 *  mysql+pymysql://root:***@localhost:3306/md_water_services
    0 rows affected.
```

source_id date description pollutant_ppm biological results

Please note that the Database is already updated based on my Findings.