1. How many tables are there in our database?	8	(=COUNTA(C4:J4))						
2. What are the names of these tables?	Tables_in_md_water_services (varchar)	SHOW TABLES;						
	data_dictionary	employee	global_water_access	location	visits	water_quality	water_source	well_pollution
3. Once you've identified the tables, write a SELECT statement to retrieve the first five records from each table.	1. For the "data_dictionary" table: SELECT * FROM data_dictionary LIMIT 5;	2. For the "employee" table: SELECT * FROM employee LIMIT 5;	3. For the "global_water_access" table: SELECT * FROM global_water_access LIMIT 5;	4. For the "location" table: SELECT * FROM location LIMIT 5;	5. For the "water_quality" table: SELECT * FROM water_quality LIMIT 5;	6. For the "visits" table: SELECT * FROM visits LIMIT 5;	7. For the "water_source" table: SELECT * FROM water_source LIMIT 5;	8. For the "well_pollution" table: SELECT * FROM well_pollution LIMIT 5;

a. data_dictionary

*table_name (varchar 255)	*column_name (varchar 255)	description (text)	datatype (varchar 50)	related_to (varchar 255)
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)	

b. employee

assigned_employee_id INT	employee_name VARCHAR(255)	phone_number VARCHAR(15)	email VARCHAR(255)	address VARCHAR(255)	town_name VARCHAR(255)	province_name VARCHAR(255)	position VARCHAR(255)
0	Amara Jengo	+99637993287		36 Pwani Mchangani Road	Sokoto	Ilanga	Field Surveyor
1	Bello Azibo	+99643864786		129 Ziwa La Kioo Road	Kilimani	Rural	Field Surveyor
2	Bakari Iniko	+99222599041		18 Mlima Tazama Avenue	Hawassa	Rural	Field Surveyor
3	Malachi Mavuso	+99945849900		100 Mogadishu Road	Akatsi	Lusaka	Field Surveyor
4	Cheche Buhle	+99381679640		1 Savanna Street	Akatsi	Rural	Field Surveyor

c. global_water_access

name VARCHAR(255)	region VARCHAR(255)	year INT	pop_n FLOAT	pop_u FLOAT	wat_bas_n FLOAT	wat_lim_n FLOAT	wat_unimp_n FLOAT	wat_sur_n FLOAT	wat_bas_r FLOAT	wat_lim_r FLOAT	wat_unimp_r FLOAT	wat_sur_r FLOAT	wat_bas_u FLOAT	wat_lim_u FLOAT	wat_unimp_u FLOAT	wat_sur_u FLOAT
Afghanistan	South Asia	2015	34413.6	24.803	61.3398	3.5112	22.1688	12.9802	52.9885	3.86114	26.5533	16.5971	86.6589	2.45027	8.87604	2.01475
Afghanistan	South Asia	2020	38928.3	26.026	75.0914	1.44754	14.5603	8.90078	66.3279	1.95682	19.6829	12.0323	100	0	0	0
Albania	Europe & Central Asia	2015	2890.52	57.434	93.3943	3.62638	2.97929	0	90.6273	5.26317	4.10955	0	95.4451	2.41331	2.14162	0
Albania	Europe & Central Asia	2020	2877.8	62.112	95.068	1.88466	3.04731	0	94.0914	2.30526	3.60338	0	95.6638	1.62809	2.7081	0
Algeria	Middle East & North Africa	2015	39728	70.848	93.4096	5.15778	1.27546	0.157193	88.3527	8.68575	2.58043	0.381108	95.4903	3.70612	0.73851	0.0650579

d. location

location_id VARCHAR(255)	address VARCHAR(255)	province_name VARCHAR(255)	town_name VARCHAR(255)	location_type VARCHAR(255)
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
AkHa00004	17 Addis Ababa Road	Akatsi	Harare	Urban

e. visits

record_id INT	location_id VARCHAR(255)	source_id VARCHAR(510)	time_of_record DATETIME	visit_count INT	time_in_queue INT	assigned_employee_id INT
0	Soll32582	Soll32582224	2021-01-01 9:10:0	1	15	12
1	KiRu28935	KiRu28935224	2021-01-01 9:17:0	1	0	46
2	HaRu19752	HaRu19752224	2021-01-01 9:36:0	1	62	40
3	AkLu01628	AkLu01628224	2021-01-01 9:53:0	1	0	1
4	AkRu03357	AkRu03357224	2021-01-01 10:11:	1	28	14

f. water_quality

record_id INT	subjective_quality_score INT	visit_count INT
0	0	1
1	1	1
2	5	1
3	10	1
4	4	1

g. water_source

source_id INT	type_of_water_source INT	Number_of_people_served INT
AkHa00000224	tap_in_home	956
AkHa00001224	tap_in_home_broken	930
AkHa00002224	tap_in_home_broken	486
AkHa00003224	well	364
AkHa00004224	tap_in_home_broken	942

h. well_pollution

source_id VARCHAR(258)	date DATETIME	description VARCHAR(255)	pollutant_ppm FLOAT	biological FLOAT	results VARCHAR(255)
KiRu28935224	2021-01-04 9:17:0	Bacteria: Giardia Lamblia	0	495.898	Contaminated: Biological
AkLu01628224	2021-01-04 9:53:0	Bacteria: E. coli	0	6.09608	Contaminated: Biological
HaZa21742224	2021-01-04 10:37:	Inorganic contaminants: Zinc, Zinc, Lead, Cadmium	2.715	0	Contaminated: Chemical
HaRu19725224	2021-01-04 11:04:	Clean	0.0288593	0.0000956996	Clean
SoRu35703224	2021-01-04 11:29:	Bacteria: E. coli	0	22.5009	Contaminated: Biological

NOTE:

Formulae used

Columns

Tables

type_of_water_source (varchar 255)	SELECT DISTINCT type_of_water_source FROM water_source;
tap_in_home	
tap_in_home_broken	
well	
shared_tap	
river	

A. Write an SQL query to retrieve records with a time_in_queue greater than 500 minutes

```
SELECT *
FROM visits
WHERE time_in_queue > 500;
```

record_id INT	location_id VARCHAR(255)	source_id VARCHAR(510)	time_of_record DATETIME	visit_count INT	time_in_queue INT	assigned_employee_id INT
899	SoRu35083	SoRu35083224	2021-01-16 10:1	6	515	28
2304	SoKo33124	SoKo33124224	2021-02-06 7:53:	5	512	16
2315	KiRu26095	KiRu26095224	2021-02-06 14:3	3	529	8
3206	SoRu38776	SoRu38776224	2021-02-20 15:0	5	509	46
3701	HaRu19601	HaRu19601224	2021-02-27 12:5	3	504	0
4154	SoRu38869	SoRu38869224	2021-03-06 10:4	2	533	24
5483	AmRu14089	AmRu14089224	2021-03-27 18:1	4	509	12
9177	SoRu37635	SoRu37635224	2021-05-22 18:4	2	515	1
9648	SoRu36096	SoRu36096224	2021-05-29 11:2	2	533	3
11631	AkKi00881	AkKi00881224	2021-06-26 6:15:	6	502	32
11647	HaRu17137	HaRu17137224	2021-06-26 13:2	6	506	28
11654	AmRu13488	AmRu13488224	2021-06-26 17:4	6	530	5
13468	AkLu02523	AkLu02523224	2021-07-24 6:15:	5	534	0
14367	AkRu02691	AkRu02691224	2021-08-07 9:12:	2	503	48
14376	HaRu19006	HaRu19006224	2021-08-07 14:5	4	514	1
14861	AmDa12121	AmDa12121224	2021-08-14 9:56:	4	515	5
15334	AkRu03816	AkRu03816224	2021-08-21 12:5	4	518	38
15763	SoRu35918	SoRu35918224	2021-08-28 6:36:	3	504	6
15767	KiRu25613	KiRu25613224	2021-08-28 9:53:	4	529	48
16312	AmPw12313	AmPw12313224	2021-09-04 13:3	6	516	10
16319	KiZu31371	KiZu31371224	2021-09-04 17:3	3	510	32
18070	KiRu25908	KiRu25908224	2021-10-02 6:41:	5	512	46
18075	AkKi01265	AkKi01265224	2021-10-02 9:15:	2	520	1
18541	AkRu06612	AkRu06612224	2021-10-09 11:4	2	523	40
18959	KiZu31006	KiZu31006224	2021-10-16 10:1	4	507	28
20372	KiZu31117	KiZu31117224	2021-11-06 9:37:	3	537	10
20384	KiRu30071	KiRu30071224	2021-11-06 15:4	2	534	28
22205	AkRu05296	AkRu05296224	2021-12-04 9:41:	2	520	1
22220	AmRu14449	AmRu14449224	2021-12-04 17:0	4	502	1
22637	AmAs10109	AmAs10109224	2021-12-11 6:46:	6	521	32
22641	AkRu03262	AkRu03262224	2021-12-11 8:52:	5	531	30
22646	SoRu36726	SoRu36726224	2021-12-11 12:2	6	533	1
22654	AkRu08016	AkRu08016224	2021-12-11 16:1	5	516	46
22656	SoRu39427	SoRu39427224	2021-12-11 18:3	4	515	38
23074	AmBe11134	AmBe11134224	2021-12-18 7:25:	5	505	42
23077	AkRu08167	AkRu08167224	2021-12-18 9:34:	4	508	34
23078	KiRu25391	KiRu25391224	2021-12-18 10:1	4	516	36
23545	AkRu04807	AkRu04807224	2021-12-25 10:2	4	501	32
23547	KiRu30657	KiRu30657224	2021-12-25 11:4	5	531	3
24023	KiRu27098	KiRu27098224	2022-01-01 7:53:	5	510	4
24489	HaRu19412	HaRu19412224	2022-01-08 14:2	2	528	18
25405	AkRu07801	AkRu07801224	2022-01-22 9:09:	3	522	12
25408	KiRu25672	KiRu25672224	2022-01-22 10:2	3	502	36
25411	SoRu36934	SoRu36934224	2022-01-22 11:2	2	518	24
26289	KiZu31252	KiZu31252224	2022-02-05 7:40:	3	513	8
27639	AmAs10315	AmAs10315224	2022-02-26 6:37:	3	503	5
27653	AkRu06817	AkRu06817224	2022-02-26 15:5	2	522	18
28110	KiRu25801	KiRu25801224	2022-03-05 6:54:	3	513	30
28537	HaRu20458	HaRu20458224	2022-03-12 12:4	5	506	2
29020	AmBe11184	AmBe11184224	2022-03-19 9:46:	3	510	40
29039	SoRu38394	SoRu38394224	2022-03-19 18:4	5	532	36
29525	SoRu36063	SoRu36063224	2022-03-26 6:26:	2	513	1
29541	KiRu30348	KiRu30348224	2022-03-26 16:1	3	530	16
30007	AmRu14612	AmRu14612224	2022-04-02 8:55:	2	539	8
30017	KiRu29325	KiRu29325224	2022-04-02 14:3	5	503	26
30849	KiRu30332	KiRu30332224	2022-04-16 6:18:	3	507	34
31310	SoRu37865	SoRu37865224	2022-04-23 6:01:	2	535	40
32297	KiRu27914	KiRu27914224	2022-05-07 8:04:	3	528	48
32307	KiRu28894	KiRu28894224	2022-05-07 13:3	6	525	1

	32317	AkRu03567	AkRu03567224	2022-05-07 18:3	5	526	1
	33206	SoRu37676	SoRu37676224	2022-05-21 16:4	4	504	7
	33647	SoRu34770	SoRu34770224	2022-05-28 11:2	4	520	38
	33650	KiRu29348	KiRu29348224	2022-05-28 12:5	2	537	10
	34101	KiRu25700	KiRu25700224	2022-06-04 10:5	3	509	7
	34585	AkRu08590	AkRu08590224	2022-06-11 16:3	4	525	30
	34587	AkRu04180	AkRu04180224	2022-06-11 17:2	4	504	1
	35102	KiRu25640	KiRu25640224	2022-06-18 8:21	6	529	38
	35109	AmAs10285	AmAs10285224	2022-06-18 10:0	2	517	24
	35120	AkRu08660	AkRu08660224	2022-06-18 15:1	4	533	5
	35642	AkRu05208	AkRu05208224	2022-06-25 13:4	6	527	46
	36083	AkRu05784	AkRu05784224	2022-07-02 9:42	4	518	5
	36608	SoRu37419	SoRu37419224	2022-07-09 16:0	5	529	1
	38012	KiRu25441	KiRu25441224	2022-07-30 12:1	5	525	28
	38944	KiIs24249	KiIs24249224	2022-08-13 8:56	4	505	12
	38947	SoRu38095	SoRu38095224	2022-08-13 13:4	6	535	30
	39394	SoRu36116	SoRu36116224	2022-08-20 16:0	5	528	10
	40293	AmBe11056	AmBe11056224	2022-09-03 11:0	6	501	32
	42140	KiZu31033	KiZu31033224	2022-10-01 14:0	5	513	30
	42609	HaRu20141	HaRu20141224	2022-10-08 14:0	6	506	32
	45307	AkRu05702	AkRu05702224	2022-11-19 9:47	6	508	8
	45310	KiRu26540	KiRu26540224	2022-11-19 10:5	5	528	2
	45317	HaRu20126	HaRu20126224	2022-11-19 14:2	6	538	16
	45326	AkRu04497	AkRu04497224	2022-11-19 18:5	4	508	42
	46212	SoRu35296	SoRu35296224	2022-12-03 9:06	4	521	40
	47164	HaRu17375	HaRu17375224	2022-12-17 12:5	4	532	48
	47171	SoRu35979	SoRu35979224	2022-12-17 17:0	3	523	16
	48536	KiRu28510	KiRu28510224	2023-01-07 15:2	4	511	30
	48538	KiZu31337	KiZu31337224	2023-01-07 15:4	6	515	0
	49351	KiRu26060	KiRu26060224	2023-01-21 9:04	6	520	0
	50982	SoRu37646	SoRu37646224	2023-02-18 13:5	3	512	28
	51858	HaRu19538	HaRu19538224	2023-03-04 18:0	3	539	4
	52264	HaRu17383	HaRu17383224	2023-03-11 7:10	5	535	30
	52285	AmRu15810	AmRu15810224	2023-03-11 17:5	3	511	5
	52752	KiZu31086	KiZu31086224	2023-03-18 13:1	3	518	2
	53262	KiRu29005	KiRu29005224	2023-03-25 7:27	2	534	30
	53275	HaRu18687	HaRu18687224	2023-03-25 13:3	3	514	10
	53278	AkRu05704	AkRu05704224	2023-03-25 13:4	2	539	36
	53795	AmBe11043	AmBe11043224	2023-04-01 9:12	5	522	14
	54687	AmAs10779	AmAs10779224	2023-04-15 10:3	2	522	30
	56081	HaRu20810	HaRu20810224	2023-05-06 18:1	6	526	40

B. Write an SQL query to retrieve water source details

```
SELECT *
FROM water_source
 WHERE source_id IN
('AkKi00881224','AkLu01628224','AkRu05234224','HaRu19601224','HaZa21742224','SoRu36096224','SoRu37635224','SoRu38776224');
```

source_id INT	type_of_water_source INT	Number_of_people_served INT
AkKi00881224	shared_tap	3398
AkLu01628224	well	210
AkRu05234224	tap_in_home_broken	496
HaRu19601224	shared_tap	3322
HaZa21742224	well	308
SoRu36096224	shared_tap	3786
SoRu37635224	shared_tap	3920
SoRu38776224	shared_tap	3180

Write an SQL query to find records where the subject_quality_score is 10 and the source was visited a second time. This query will help identify cases where home taps have received a perfect score and were revisited.

```
SELECT *
  FROM
water_quality
  WHERE
subjective_quali
ty_score = 10
    AND
visit_count = 2;
```

record_id INT	subjective_quality_score INT	visit_count INT
59	10	2
137	10	2
269	10	2
363	10	2
378	10	2
618	10	2
752	10	2
801	10	2
819	10	2
850	10	2
1176	10	2
1372	10	2
1496	10	2
1746	10	2
1753	10	2
2298	10	2
2702	10	2
3203	10	2
3489	10	2
4750	10	2
4906	10	2
5068	10	2
5072	10	2
5279	10	2
5343	10	2
5621	10	2
5656	10	2
5918	10	2
6074	10	2
6620	10	2
6731	10	2
7251	10	2
7271	10	2
7284	10	2
7711	10	2
7747	10	2
7761	10	2
7980	10	2
8076	10	2
8515	10	2
8810	10	2
9154	10	2
9287	10	2
9632	10	2
9760	10	2
9815	10	2
10129	10	2
10139	10	2
10385	10	2
10553	10	2
11161	10	2
11896	10	2
12296	10	2
12488	10	2
13282	10	2
13463	10	2
13748	10	2
13768	10	2
13776	10	2
14671	10	2
15642	10	2
16035	10	2
16211	10	2
16447	10	2
16459	10	2
16577	10	2
17058	10	2
17534	10	2
17967	10	2

	18073	10	2								
	18075	10	2								
	18220	10	2								
	18713	10	2								
	18826	10	2								
	19658	10	2								
	19693	10	2								
	19759	10	2								
	19871	10	2								
	20009	10	2								
	20336	10	2								
	21186	10	2								
	21238	10	2								
	21277	10	2								
	21962	10	2								
	22197	10	2								
	22875	10	2								
	23050	10	2								
	23318	10	2								
	23367	10	2								
	23483	10	2								
	23555	10	2								
	23999	10	2								
	24165	10	2								
	24260	10	2								
	24320	10	2								
	24332	10	2								
	24476	10	2								
	24790	10	2								
	24869	10	2								
	26032	10	2								
	26213	10	2								
	26304	10	2								
	26583	10	2								
	26616	10	2								
	26662	10	2								
	26694	10	2								
	26988	10	2								
	28268	10	2								
	28269	10	2								
	28347	10	2								
	28461	10	2								
	28991	10	2								
	29525	10	2								
	30208	10	2								
	30777	10	2								
	31021	10	2								
	31115	10	2								
	31231	10	2								
	31291	10	2								
	31979	10	2								
	32036	10	2								
	32733	10	2								
	33048	10	2								
	33407	10	2								
	33935	10	2								
	33938	10	2								
	34050	10	2								
	34465	10	2								
	34515	10	2								
	34560	10	2								
	34589	10	2								
	34894	10	2								
	34907	10	2								
	34967	10	2								
	35384	10	2								
	35640	10	2								
	35966	10	2								
	36078	10	2								
	36753	10	2								
	36970	10	2								
	37044	10	2								
	37243	10	2								
	37250	10	2								
	37326	10	2								
	37691	10	2								
	37967	10	2								

	38601	10	2								
	39356	10	2								
	39625	10	2								
	39793	10	2								
	39817	10	2								
	40050	10	2								
	40584	10	2								
	40624	10	2								
	41038	10	2								
	41224	10	2								
	41812	10	2								
	41890	10	2								
	42072	10	2								
	42144	10	2								
	42480	10	2								
	42728	10	2								
	42922	10	2								
	43284	10	2								
	43302	10	2								
	43436	10	2								
	43694	10	2								
	43745	10	2								
	45215	10	2								
	45482	10	2								
	46974	10	2								
	47151	10	2								
	47167	10	2								
	47483	10	2								
	47571	10	2								
	47871	10	2								
	47907	10	2								
	48339	10	2								
	48774	10	2								
	48791	10	2								
	48821	10	2								
	48880	10	2								
	49309	10	2								
	49362	10	2								
	49618	10	2								
	49809	10	2								
	50084	10	2								
	50106	10	2								
	50290	10	2								
	50545	10	2								
	50836	10	2								
	50979	10	2								
	50988	10	2								
	51253	10	2								
	51295	10	2								
	51651	10	2								
	52578	10	2								
	52627	10	2								
	52727	10	2								
	53068	10	2								
	53250	10	2								
	53280	10	2								
	54046	10	2								
	54154	10	2								
	54530	10	2								
	55036	10	2								
	56461	10	2								
	56545	10	2								
	56808	10	2								
	56950	10	2								
	57153	10	2								
	57155	10	2								
	57238	10	2								
	57522	10	2								
	58664	10	2								
	58728	10	2								
	59733	10	2								
	59779	10	2								

5.A. Print the first few rows of the table:
- Write an SQL query to retrieve the first few rows from the "well_pollution" table.

```
SELECT * FROM well_pollution LIMIT 5;
```

source_id VARCHAR(258)	date DATETIME	description VARCHAR(255)	pollutant_ppm FLOAT	biological FLOAT	results VARCHAR(255)
KiRu28935224	2021-01-04 9:17:	Bacteria: Giardia Lamblia	0	495.898	Contaminated: Biological
AkLu01628224	2021-01-04 9:53:	Bacteria: E. coli	0	6.09608	Contaminated: Biological
HaZa21742224	2021-01-04 10:3	Inorganic contaminants: Zinc, Zinc, Lead, Cadmium	2.715	0	Contaminated: Chemical
HaRu19725224	2021-01-04 11:0	Clean	0.0288593	0.0000956996	Clean
SoRu35703224	2021-01-04 11:2	Bacteria: E. coli	0	22.5009	Contaminated: Biological

5.B. Identify descriptions with the word "Clean" but biological contamination:
- Write an SQL query to identify records where the description mistakenly includes the word "Clean" but the biological column has a value greater than 0.01.

```
SELECT * FROM well_pollution WHERE description LIKE 'Clean%' AND biological > 0.01;
```

source_id VARCHAR(258)	date DATETIME	description VARCHAR(255)	pollutant_ppm FLOAT	biological FLOAT	results VARCHAR(255)
AkRu06489224	2021-01-10 9:44:	Clean Bacteria: Giardia Lamblia	0.0897904	38.467	Clean
KiRu25473224	2021-02-07 15:5	Clean Bacteria: Giardia Lamblia	0.0630094	24.4536	Clean
HaRu17401224	2021-03-01 13:4	Clean Bacteria: Giardia Lamblia	0.0649209	25.8129	Clean
AkRu07137224	2021-03-04 13:4	Clean Bacteria: Giardia Lamblia	0.0656843	18.2978	Clean
KiRu27205224	2021-03-13 14:1	Clean Bacteria: Giardia Lamblia	0.0418018	49.4281	Clean
AkHa00514224	2021-04-11 12:1	Clean Bacteria: Giardia Lamblia	0.0305404	22.0255	Clean
AmAm09776224	2021-05-23 11:2	Clean Bacteria: Giardia Lamblia	0.0963821	13.6574	Clean
SoIl32894224	2021-07-11 11:3	Clean Bacteria: Giardia Lamblia	0.0712408	5.44957	Clean
AkRu07366224	2021-07-23 11:1	Clean Bacteria: Giardia Lamblia	0.0969458	26.0308	Clean
KiHa23443224	2021-09-05 12:3	Clean Bacteria: Giardia Lamblia	0.0828	13.7162	Clean
AkRu06340224	2021-11-01 15:2	Clean Bacteria: Giardia Lamblia	0.051746	36.4594	Clean
AkRu06363224	2021-11-15 13:3	Clean Bacteria: Giardia Lamblia	0.0882158	45.3731	Clean
AkRu08749224	2021-11-19 13:3	Clean Bacteria: E. coli	0.0888194	9.28085	Clean
AkRu07066224	2021-12-10 14:4	Clean Bacteria: E. coli	0.0151035	44.4652	Clean
KiAm21996224	2021-12-16 10:0	Clean Bacteria: E. coli	0.0480415	25.4174	Clean
AkRu05377224	2022-01-01 10:5	Clean Bacteria: E. coli	0.0226614	0.613503	Clean
HaRu20765224	2022-02-07 12:3	Clean Bacteria: E. coli	0.0213831	6.95778	Clean
SoMa34473224	2022-03-10 9:01:	Clean Bacteria: E. coli	0.0745784	13.6458	Clean
SoKo33474224	2022-03-20 16:0	Clean Bacteria: E. coli	0.0687177	10.5712	Clean
AkHa00226224	2022-03-26 11:5	Clean Bacteria: E. coli	0.0424571	39.4978	Clean
AkRu06881224	2022-03-28 13:3	Clean Bacteria: E. coli	0.0202831	33.0692	Clean
AkLu02254224	2022-04-25 12:5	Clean Bacteria: E. coli	0.0635533	16.9482	Clean
AkRu08416224	2022-05-29 15:5	Clean Bacteria: E. coli	0.0406701	27.0805	Clean
SoMa34205224	2022-07-31 13:2	Clean Bacteria: E. coli	0.0726217	23.892	Clean
KiIs23482224	2022-08-15 16:4	Clean Bacteria: E. coli	0.0539137	41.0842	Clean
AmRu13252224	2022-08-20 14:2	Clean Bacteria: E. coli	0.0653589	30.8643	Clean
KiRu27297224	2022-11-21 13:1	Clean Bacteria: E. coli	0.0856842	8.62509	Clean
KiRu26929224	2022-12-18 10:5	Clean Bacteria: E. coli	0.0748808	26.6541	Clean
HaAm16042224	2022-12-22 13:5	Clean Bacteria: E. coli	0.0605767	4.54755	Clean
AkRu03107224	2023-01-01 15:3	Clean Bacteria: E. coli	0.0901101	18.3118	Clean
KiHa23111224	2023-01-12 15:5	Clean Bacteria: E. coli	0.0161869	36.8748	Clean
AkRu03771224	2023-02-26 12:4	Clean Bacteria: E. coli	0.0625017	19.7648	Clean

	AkRu04253224	2023-02-26 13:4	Clean Bacteria: E. coli	0.0273811	3.08854	Clean						
	HaSe20949224	2023-04-03 16:2	Clean Bacteria: E. coli	0.0458311	32.9004	Clean						
	KiAm22477224	2023-05-14 15:0	Clean Bacteria: E. coli	0.0568172	25.6586	Clean						
	AkRu08365224	2023-05-28 14:5	Clean Bacteria: E. coli	0.0826689	19.586	Clean						
	AkRu04026224	2023-06-05 16:5	Clean Bacteria: E. coli	0.0124892	22.9488	Clean						
	KiRu29194224	2023-06-08 9:28	Clean Bacteria: E. coli	0.0502865	8.40146	Clean						

5.C. Before executing UPDATE queries to fix the inconsistencies, create a copy of the "well_pollution" table using the CREATE TABLE AS SELECT statement.

```
CREATE TABLE well_pollution_copy AS (SELECT * FROM well_pollution);
```

5.D. Update descriptions that mistakenly mention "Clean Bacteria: E. coli":
- Write an SQL query to update the descriptions that mistakenly mention "Clean Bacteria: E. coli" to "Bacteria: E. coli".

```
UPDATE well_pollution SET description = REPLACE (description, 'Clean Bacteria: E. coli', 'Bacteria: E. coli') WHERE description LIKE 'Clean Bacteria: E. coli%';
```

5.E. Update descriptions that mistakenly mention "Clean Bacteria: Giardia Lamblia":
- Write an SQL query to update the descriptions that mistakenly mention "Clean Bacteria: Giardia Lamblia" to "Bacteria: Giardia Lamblia".

```
UPDATE well_pollution SET description = REPLACE (description, 'Clean Bacteria: Giardia Lamblia', 'Bacteria: Giardia Lamblia') WHERE description LIKE 'Clean Bacteria: Giardia Lamblia%';
```

5.F. Update the results column to "Contaminated: Biological" where the biological column has a value greater than 0.01 and the current results are "Clean":
- Write an SQL query to update the results column to "Contaminated: Biological" in the records where the biological column has a value greater than 0.01 and the current results are "Clean". Explanation: These queries help fix the identified inconsistencies by updating the descriptions and results columns in the "well_pollution" table.

```
UPDATE well_pollution SET results = 'Contaminated: Biological' WHERE biological > 0.01 AND results = 'Clean';
```

5.G. Verify the description updates:
- Write an SQL query to check if the updates for the descriptions were successful. Explanation: These queries help verify if the updates made to the descriptions columns in the "well_pollution" table were successful.

```
SELECT * FROM well_pollution WHERE description LIKE 'Clean%';
```

source_id VARCHAR(258)	date DATETIME	description VARCHAR(255)	pollutant_ppm FLOAT	biological FLOAT	results VARCHAR(255)
HaRu19725224	2021-01-04 11:0	Clean	0.0288593	0.0000956996	Clean
HaSe21346224	2021-01-04 11:5	Clean	0.0140376	0.0000898989	Clean
AkRu05973224	2021-01-04 13:4	Clean	0.084344	0.000062396	Clean
AkHa00706224	2021-01-04 16:1	Clean	0.0862577	0.0000422834	Clean
HaRu20773224	2021-01-07 9:08	Clean	0.0599682	0.0000242289	Clean

5.H. Verify the results updates:
- Write an SQL query to check if the updates for the results column were successful. Explanation: These queries help verify if the updates made to the and results columns in the "well_pollution" table were successful.

SELECT * FROM well_pollution WHERE results = 'Clean' AND biological > 0.01 LIMIT 5;	source_id VARCHAR(258)	date DATETIME	description VARCHAR(255)	pollutant_ppm FLOAT	biological FLOAT	results VARCHAR(255)					

Q1. What is the address of Bello Azibo?

```
SELECT *
FROM employee LIMIT 5;
```

assigned_employee_id INT	employee_name VARCHAR (255)	phone_number VARCHAR(15)	email VARCHAR(255)		address VARCHAR(255)	town_name VARCHAR(255)	province_name VARCHAR(255)	position VARCHAR(255)
0	Amara Jengo	+99637993287			36 Pwani Mchangani Road	Sokoto	Ilanga	Field Surveyor
1	Bello Azibo	+99643864786			129 Ziwa La Kioo Road	Kilimani	Rural	Field Surveyor
2	Bakari Iniko	+99222599041			18 Mlima Tazama Avenue	Hawassa	Rural	Field Surveyor
3	Malachi Mavuso	+99945849900			100 Mogadishu Road	Akatsi	Lusaka	Field Surveyor
4	Cheche Buhle	+99381679640			1 Savanna Street	Akatsi	Rural	Field Surveyor

Q2. What is the name and phone number of our Microbiologist?

```
SELECT *
FROM employee
WHERE position LIKE '%bio%';
```

assigned_employee_id INT	employee_name VARCHAR (255)	phone_number VARCHAR(15)	email VARCHAR(255)		address VARCHAR(255)	town_name VARCHAR(255)	province_name VARCHAR(255)	position VARCHAR(255)
102	Vuyisile Ghadir	+99712584936			104 Kenyatta Street	Akatsi	Rural	Micro Biologist

Q3. What is the source_id of the water source shared by the most number of people? Hint: Use a comparison operator.

```
SELECT *
FROM water_source
ORDER BY Number_of_people_served DESC
LIMIT 10;
```

source_id INT	type_of_water_source INT	Number_of_people_served INT
AkRu05603224	shared_tap	3998
HaRu19509224	shared_tap	3998
AmAs10911224	shared_tap	3996
KiHa22867224	shared_tap	3996
AkRu04862224	shared_tap	3996
KiZu31330224	shared_tap	3994
HaRu19839224	shared_tap	3994
KiZu31415224	shared_tap	3992
KiRu28630224	shared_tap	3992
KiRu26218224	shared_tap	3990

Q4. What is the population of Maji Ndogo?

Hint: Start by searching the data_dictionary table for the word 'population'.

```
SELECT *
FROM global_water_access
WHERE name = 'Maji Ndogo';
```

name VARCHAR(255)	region VARCHAR(255)	year INT	pop_n FLOAT	pop_n (Million) FLOAT	pop_u FLOAT	wat_bas_n FLOAT	wat_lim_n FLOAT	wat_unimp_n FLOAT	wat_sur_n FLOAT	wat_bas_r FLOAT	wat_lim_r FLOAT	wat_unimp_r FLOAT	wat_sur_r FLOAT	wat_bas_u FLOAT	wat_lim_u FLOAT	wat_unimp_u FLOAT	wat_sur_u FLOAT
Maji Ndogo	Sub-Saharan Africa	2015	27628.1	27628100	36	47.6	31.4	12.8	8.2	41.15	36.68	12.67	9.49	58.32	22.57	12.21	6.89
Maji Ndogo	Sub-Saharan Africa	2020	27628.1	27628100	36	47.6	31.4	12.8	8.2	41.15	36.68	12.67	9.49	58.32	22.57	12.21	6.89

Q5. Which SQL query returns records of employees who are Civil Engineers residing in Dahabu or living on an avenue?

```
SELECT *
FROM employee
WHERE position = 'Civil Engineer' AND (province_name = 'Dahabu' OR address LIKE '%Avenue%');
```

assigned_employee_id INT	employee_name VARCHAR (255)	phone_number VARCHAR(15)	email VARCHAR(255)	address VARCHAR(255)	town_name VARCHAR(255)	province_name VARCHAR(255)	position VARCHAR(255)
71	Jengo Rudo	+99317854629		33 Angélique Kidjo Avenue	Amanzi	Dahabu	Civil Engineer
81	Nia Vuyisile	+99273841596		52 Moroni Avenue	Sokoto	Rural	Civil Engineer

Q6. Create a query to identify potentially suspicious field workers based on an anonymous tip. This is the description we are given:

The employee's phone number contained the digits 86 or 11.
The employee's last name started with either an A or an M.
The employee was a Field Surveyor.
Which option is correct?

```
SELECT *
FROM employee
WHERE (phone_number LIKE '%86%' OR phone_number LIKE '%11%')
  AND (employee_name LIKE '%A%' OR employee_name LIKE '%M%')
  AND position = 'Field Surveyor';
```

assigned_employee_id INT	employee_name VARCHAR (255)	phone_number VARCHAR(15)	email VARCHAR(255)	address VARCHAR(255)	town_name VARCHAR(255)	province_name VARCHAR(255)	position VARCHAR(255)
1	Bello Azibo	+99643864786		129 Ziwa La Kioo Road	Kilimani	Rural	Field Surveyor
5	Zuriel Matembo	+99034075111		26 Bahari Ya Faraja Road	Kilimani	Rural	Field Surveyor

Q7. What is the result of the following query? Choose the most appropriate description of the results set.

```
SELECT *
FROM well_pollution
WHERE description LIKE 'Clean_%' OR results = 'Clean' AND biological < 0.01;
```

ANS: 4916 records are returned. This query describes the pollution samples that had an insignificant amount of biological contamination.

Q8. Which query will identify the records with a quality score of 10, visited more than once?

```
SELECT *
FROM water_quality
WHERE visit_count >= 2
AND subjective_quality_score = 10;
```

Q9. You have been given a task to correct the phone number for the employee named 'Bello Azibo'. The correct number is +99643864786. Write the SQL query to accomplish this. Note: Running these queries on the employee table may create issues later, so use the knowledge you have learned to avoid that.

```
UPDATE employee
SET phone_number = '+99643864786'
WHERE employee_name = 'Bello Azibo';
```

Q10. How many rows of data are returned for the following query?

```
SELECT *
FROM well_pollution
WHERE description
IN ('Parasite: Cryptosporidium', 'biologically contaminated')
OR (results = 'Clean' AND biological > 0.01);
```

ANS: 570 rows