

Laporan Praktikum

Mata Kuliah Sistem Manajemen Basis Data

SQL Practice

Dwi Cahya Ramadani

25 April 2022

1 Soal Easy

- Show first name, last name, and gender of patients who's gender is 'M'
Query :

Listing 1: Easy - 1

```
1 SELECT first_name, last_name, gender FROM patients
2 WHERE gender is 'M'
```

Penjelasan :

- Baris 1 : Mengambil data first_name, last_name, dan gender dari tabel patients
- Baris 2 : Dengan kondisi dimana gendernya adalah "M"

Screenshot :

The screenshot shows a computer interface for practicing SQL queries. At the top, there is a toolbar with various icons and tabs. Below the toolbar, there are sections for 'Run', 'Support', and 'Light Mode'. A sidebar on the left lists 'SQL Database' and 'Learning Resources'. The main area contains a code editor with the following SQL query:

```
1 SELECT first_name, last_name, gender FROM patients
2 WHERE gender is 'M'
```

To the right of the code editor, a green box says 'You Got The Query Correct'. Below the code editor, the results of the query are displayed in a table:

first_name	last_name	gender
Darryl	Morris	M
Sam	Platter	M
Benjamin	Brockman	M
Jonathan	Capone	M
Evan	Stone	M
John	Klump	M

At the bottom of the results table, it says 'Showing 1 to 100 of 2468 results'. To the right of the table, another green box says 'You Got It Correct' and 'Congratulations on solving the question.' The status bar at the bottom of the screen shows the date and time.

- Show first name and last name of patients who does not have allergies (null)

Query :

Listing 2: Easy - 2

```
1 SELECT first_name, last_name FROM patients
2 WHERE allergies ISNULL
```

- Baris 1 : Mengambil data first_name dan last_name dari tabel patients
- Baris 2 : Dengan kondisi dimana allergies dari pasien bernilai null (tidak memiliki allergi)

Screenshot :

first_name	last_name
Mel	Johnson
Will	Magnus
Sasha	McGraw
Betty	Lewis
Alan	Gordon
Tristan	Sawyer

- Show first name of patients that start with the letter 'C'

Query :

Listing 3: Easy - 3

```
1 SELECT first_name FROM patients
2 WHERE first_name LIKE 'C%'
```

- Baris 1 : Mengambil data first_name dari tabel patients
- Baris 2 : Dengan kondisi dimana first_name dari pasien diawali dengan huruf "C"

Screenshot :

The screenshot shows a computer desktop with a browser window open to 'sql-practice.com'. The query entered is:

```
1 SELECT first_name FROM patients
2 WHERE first_name LIKE 'C'
```

The results table shows the following data:

first_name
Claudia
Claudia
Caesar
Cordelia
Cassie
Constance

At the bottom right of the application window, there is a green message: "You Got It Correct" followed by "Congratulations on solving the question."

- Show first name and last name of patients that weight within the range of 100 to 120 (inclusive)

Query :

Listing 4: Easy - 4

```
1 SELECT first_name, last_name FROM patients
2 Where weight between 100 AND 120
```

- Baris 1 : Mengambil data first_name dan last_name dari tabel patients
- Baris 2 : Dengan kondisi dimana weight dari pasien bernilai diantara 100 dan 120

Screenshot :

The screenshot shows a computer desktop with a browser window open to 'sql-practice.com'. The query entered is:

```
1 SELECT first_name, last_name FROM patients
2 WHERE weight BETWEEN 100 AND 120
```

The results table shows the following data:

first_name	last_name
Deunan	Knute
Rita	Morrow
Piper	Cooper
Mel	Johnson
Agnes	Eckhart
Will	Magnus

At the bottom right of the application window, there is a green message: "You Got It Correct" followed by "Congratulations on solving the question."

- Update the patients table for the allergies column. If the patient's allergies is null then replace it with 'NKA'

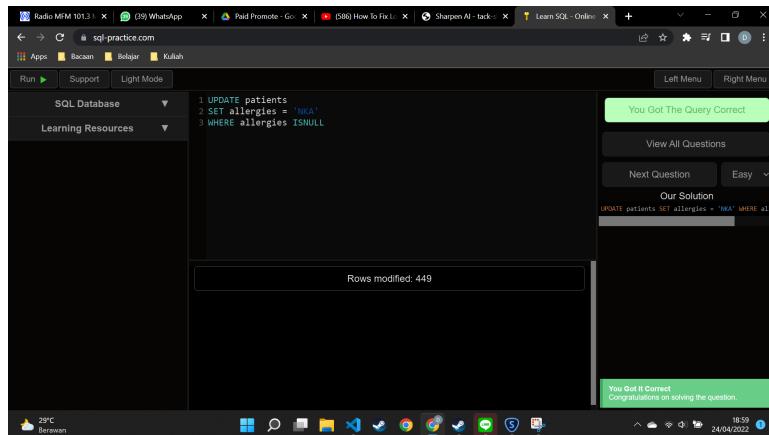
Query :

Listing 5: Easy - 5

```
1 update patients
2 set allergies = 'NKA'
3 where allergies isnull
```

- Baris 1 : Mengupdate tabel patients
- Baris 2 : Kolom allergies akan diisi dengan "NKA"
- Baris 3 : Pada data dimana allergies dari pasien bernilai null (tidak memiliki allergi)

Screenshot :



- Show first name and last name concatenated into one column to show their full name

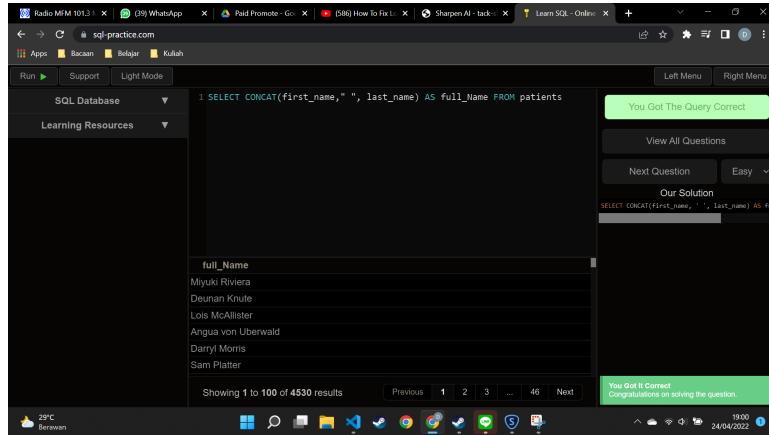
Query :

Listing 6: Easy - 6

```
1 SELECT concat(first_name, " ", last_name) as full_Name FROM
     patients
```

- Baris 1 : Mengambil data gabungan dari first_name dan last_name yang akan disebut sebagai full_name dari data pada tabel patients

Screenshot :



- Show first name, last name, and the full province name of each patient.
Example: 'Ontario' instead of 'ON'
Query :

Listing 7: Easy - 7

```
1 select first_name, last_name, province_name from patients
2 inner join provinces
3 on patients.province_id = provinces.province_id
```

- Baris 1 : Mengambil data first_name, last_name, dan provincename dari tabel patients
- Baris 2 : Menggabungkan tabel provinces sesuai dengan data tabel patients
- Baris 3 : Dengan acuan province_id pada tabel patients merujuk pada province_id pada tabel provinces

Screenshot :

The screenshot shows a browser window with the URL sql-practice.com. The main content area displays a SQL query:

```
1 SELECT first_name, last_name, province_name FROM patients
2 INNER JOIN provinces
3 ON patients.province_id = provinces.province_id
```

Below the query, a table is shown with the following data:

first_name	last_name	province_name
Miyuki	Riviera	Ontario
Deunan	Krute	Ontario
Lois	McAllister	Ontario
Angua	von Uberwald	Ontario
Darryl	Morris	Ontario
Sam	Platter	Ontario

At the bottom of the page, there are navigation links for 'Previous' and 'Next' pages, and a message: "Showing 1 to 100 of 4530 results". On the right side, there are buttons for "View All Questions", "Next Question", and "Easy". A green box at the top right says "You Got The Query Correct". Another green box at the bottom right says "You Got It Correct" and "Congratulations on solving the question".

- Show how many patients have a birth_date with 2010 as the birth year
Query :

Listing 8: Easy - 8

```
1 select count(birth_date) from patients
2 where year(birth_date) = 2010
```

- Baris 1 : Mengambil banyaknya data pada kolom brthdate dari tabel patients
- Baris 2 : Dengan kondisi dimana tahun birth.date dari pasien adalah 2010

Screenshot :

The screenshot shows a browser window with the URL sql-practice.com. The main content area displays a SQL query:

```
1 SELECT COUNT(birth_date) FROM patients
2 WHERE YEAR(birth_date) = 2010
```

Below the query, a table is shown with the following data:

count(birth_date)
57

On the right side, there are buttons for "View All Questions", "Next Question", and "Easy". A green box at the top right says "You Got The Query Correct". A green box at the bottom right says "You Got It Correct" and "Congratulations on solving the question".

- Show the first_name, last_name, and height of the patient with the greatest height

Query :

Listing 9: Easy - 9

```
1 select first_name, last_name, height from patients
2 order by height desc limit 1
```

- Baris 1 : Mengambil data first_name, last_name, dan height dari tabel patients
- Baris 2 : Mengurutkan data berdasarkan height secara descending (dari height terbesar), dan ambil 1 data pertama

Screenshot :

The screenshot shows a web-based SQL practice environment. At the top, there's a navigation bar with tabs like Radio MI!, WhatsApp, Pad Prom..., Sharpen AI, Learn SQL, and translate. Below the navigation bar, there are sections for Apps, Bacaan, Belajar, and Kuliah. A sidebar on the left has buttons for Run, Support, and Light Mode, and dropdown menus for SQL Database and Learning Resources.

In the main area, there's a code editor with the following SQL query:

```
1 SELECT first_name, last_name, height FROM patients
2 ORDER BY height DESC LIMIT 1
```

To the right of the code editor, a green box says "You Got The Query Correct". Below the code editor, there's a table with three columns: first_name, last_name, and height. The data row is:

first_name	last_name	height
Salvatore	Carlyle	215

Further down, under "Our Solution", is the same query. At the bottom right, another green box says "You Got It Correct" and "Congratulations on solving the question."

- Show all columns for patients who have one of the following patient_ids: 1,45,534,879,1000

Query :

Listing 10: Easy - 10

```
1 select * from patients
2 where patient_id in (1,45,534,879,1000)
```

- Baris 1 : Mengambil semua data dari tabel patients
- Baris 2 : Dengan kondisi dimana patient_id nya termasuk diantara 1, 45, 534, 879, 1000

Screenshot :

The screenshot shows a browser window with the URL sql-practice.com. The page displays a SQL query and its results. The query is:

```
1 SELECT * FROM patients  
2 WHERE patient_id IN (1,45,534,879,1000)
```

The results table has columns: patient_id, first_name, last_name, gender, birth_date, and city. The data is:

patient_id	first_name	last_name	gender	birth_date	city
1	Miyuki	Riviera	F	1953-12-06	Hamilton
45	Hestia	Harrison	F	1975-12-10	Delhi
534	Ronnie	John	M	2000-07-02	Hamilton
879	Jay	Hallwell	M	1957-07-10	Hamilton
1000	Sarah	Kelling	F	2000-06-07	Stoney C

On the right side of the page, there are several buttons: 'Left Menu' and 'Right Menu', 'View All Questions', 'Next Question', 'Easy', 'Our Solution', and a message box saying 'You Got The Query Correct'. Below the message box is another message: 'SELECT * FROM patients WHERE patient_id IN (1,45,534,879,1000); You Got It Correct Congratulations on solving the question.'

- Show the total number of admissions

Query :

Listing 11: Easy - 11

```
1 select count(*) as total_admissions from admissions
```

- Baris 1 : Mengambil banyaknya data dari admissions (akan disebut sebagai total_admissions)

Screenshot :

The screenshot shows a browser window with the URL sql-practice.com. The page displays a SQL query and its results. The query is:

```
1 SELECT COUNT(*) AS total_admissions FROM admissions
```

The results table has one column: total_admissions. The data is:

total_admissions
5067

On the right side of the page, there are several buttons: 'Left Menu' and 'Right Menu', 'View All Questions', 'Next Question', 'Easy', 'Our Solution', and a message box saying 'You Got The Query Correct'. Below the message box is another message: 'SELECT COUNT(*) AS total_admissions FROM admissions; You Got It Correct Congratulations on solving the question.'

2 Soal Medium

- Show unique birth years from patients and order them by ascending Query :

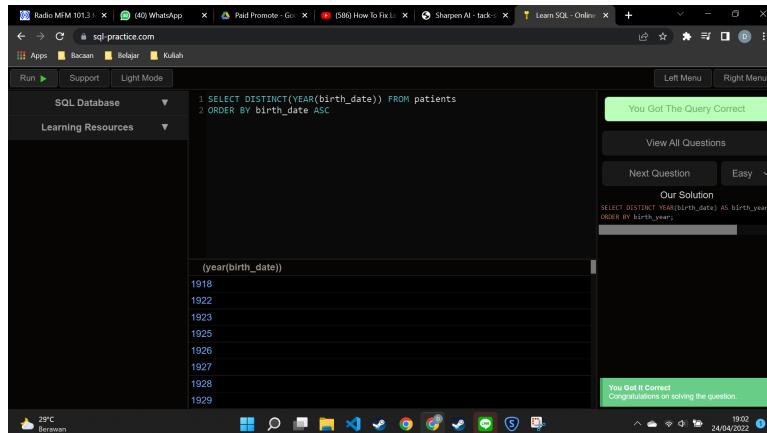
Listing 12: Medium - 1

```
1 select distinct(year(birth_date)) from patients
2 order by birth_date ASC
```

Penjelasan :

- Baris 1 : Mengambil data tahun yang unik (distinct) pada kolom birth_date dari tabel patients
- Baris 2 : Mengurutkan secara ascending berdasarkan birth_date

Screenshot :



- Show unique first names from the patients table which only occurs once in the list. For example, if two or more people are named 'John' in the first_name column then don't include their name in the output list. If only 1 person is named 'Leo' then include them in the output.

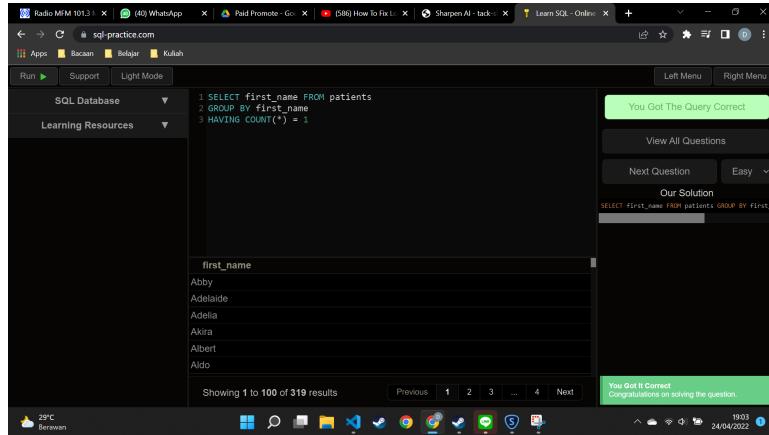
Query :

Listing 13: Medium - 2

```
1 select first_name from patients
2 group by first_name
3 having count(*) = 1
```

- Baris 1 : Mengambil data first_name dari tabel patients
- Baris 2 : Mengelompokkan data berdasarkan first_name
- Baris 3 : Mengambil data dengan jumlah data dalam kelompok adalah 1

Screenshot :



- Show patient_id and first_name from patients where their first_name start and ends with 's' and is atleast 5 characters long.

Query :

Listing 14: Medium - 3

```
1 select patient_id, first_name from patients
2 where first_name Like "s%s"
```

- Baris 1 : Mengambil data patient_id dan first_name dari tabel patients
- Baris 2 : Dengan kondisi dimana first_name dari pasien diawali dengan huruf "s" yang diikuti minimal 3 karakter dan diakhiri dengan huruf "s"

Screenshot :

The screenshot shows a web-based SQL editor interface. At the top, there's a navigation bar with tabs like 'Run', 'Support', and 'Light Mode'. Below the navigation is a dropdown menu for 'SQL Database' and 'Learning Resources'. The main area contains a code editor with the following SQL query:

```
1 SELECT patient_id, first_name FROM patients
2 WHERE first_name LIKE "%s%"
```

Below the code editor is a table with the following data:

patient_id	first_name
221	Stanislaus
475	Spiros
764	Spiros
1673	Spiros
1890	Stanislaus
2032	Seras
2037	Seamus
2382	Stanislaus

To the right of the table, there's a message box saying 'You Got The Query Correct'. Below the table, there's another message box with 'Our Solution' containing the same SQL query. At the bottom right, there's a message box saying 'You Got It Correct' with the text 'Congratulations on solving the question.'

- Show patient_id, first_name, last_name from patients whos primary_diagnosis is 'Dementia'. Primary diagnosis is stored in the admissions table

Query :

Listing 15: Medium - 4

```
1 select patients.patient_id, first_name, last_name from
2     patients
3 inner join admissions
4 on patients.patient_id = admissions.patient_id
5 where primary_diagnosis is "Dementia"
```

- Baris 1 : Mengambil data patient_id, first_name dan last_name dari tabel patients
- Baris 2 : Menggabungkan tabel admissions sesuai data dari tabel patients
- Baris 3 : Dengan acuan patient_id pada tabel patients merujuk pada patient_id pada tabel admissions
- Baris 4 : Mengambil data dengan kondisi dimana primary_diagnosis dari pasien adalah "Dementia"

Screenshot :

The screenshot shows a computer screen with a browser window open to a SQL practice site. The query entered is:

```
1 SELECT patients.patient_id, first_name, last_name FROM patients
2 INNER JOIN admissions
3 ON patients.patient_id = admissions.patient_id
4 WHERE primary_diagnosis IS "Dementia"
```

The results table shows patient information:

patient_id	first_name	last_name
1	Miyuki	Riviera
22	Suzanne	Lovegood
695	Joyce	Beck
846	Avon	Sanderson
1139	Daniel	Cunningham
1280	Thora	Page
1330	Sophie	Spencer
1359	Charlie	Wateman

On the right side of the interface, there are several buttons and messages:

- "You Got The Query Correct"
- "View All Questions"
- "Next Question" and "Easy"
- "Our Solution" button
- "SELECT patients.patient_id, first_name, last_name
INNER JOIN admissions
ON patients.patient_id = admissions.patient_id
WHERE primary_diagnosis = "Dementia";"
- "You Got It Correct" message
- "Congratulations on solving the question."

- Show patient_id, first_name, last_name from the patients table. Order the rows by the first_name ascending and then by the last_name descending Query :

Listing 16: Medium - 5

```
1 select patient_id, first_name, last_name from patients
2 order by first_name ASC, last_name DESC
```

- Baris 1 : Mengambil data patient_id, first_name, dan last_name dari tabel patients
- Baris 2 : Mengurutkan berdasarkan first_name secara ascending lalu mengurutkan berdasarkan last_name secara descending

Screenshot :

The screenshot shows a computer screen with a browser window open to a SQL practice site. The query entered is:

```
1 SELECT patient_id, first_name, last_name FROM patients
2 ORDER BY first_name ASC, last_name DESC
```

The results table shows patient information:

patient_id	first_name	last_name
3933	Abby	Townsend
4465	Abe	Polin
2969	Abe	Owens
555	Abe	Lynley
2871	Abe	Carver
556	Abe	Brown

On the right side of the interface, there are several buttons and messages:

- "You Got The Query Correct"
- "View All Questions"
- "Next Question" and "Easy"
- "Our Solution" button
- "SELECT patient_id, first_name, last_name FROM patients
ORDER BY first_name ASC, last_name DESC;"
- "Showing 1 to 100 of 4530 results"
- "You Got It Correct" message
- "Congratulations on solving the question."

- Show the total amount of male patients and the total amount of female patients in the patients table

Query :

Listing 17: Medium - 6

```
1 select count(case when gender = 'M' then 1 end) as male_count,
      count(case when gender = 'F' then 2 end) as female_count
     from patients
```

- Baris 1 : Mengambil banyaknya data yang memiliki gender ”M” (selanjutnya akan disebut male_count) dan banyaknya data yang memiliki gender ”F” (selanjutnya akan disebut female_count) dari tabel patients

Screenshot :

	male_count	female_count
	2468	2062

- Show first and last name, allergies from patients which have allergies to either 'Penicillin' or 'Morphine'. Show results ordered ascending by allergies then by first_name then by last_name.

Query :

Listing 18: Medium - 7

```
1 select first_name, last_name, allergies from patients
2 where allergies = "Penicillin" OR allergies = "Morphine"
3 order by allergies ASC, first_name ASC, last_name ASC
```

- Baris 1 : Mengambil data first_name, last_name, dan allergies dari tabel patients
- Baris 2 : Dengan kondisi dimana allergies yang dimiliki pasien adalah ”Penicillin” atau ”Morphine”

- Baris 3 : Mengurutkan data berdasarkan allergies secara ascending, lalu mengurutkan data berdasarkan first_name secara ascending, dan terakhir mengurutkan data berdasarkan last_name secara ascending

Screenshot :

The screenshot shows a browser window with a SQL query editor and a results table. The query is:

```

1 SELECT first_name, last_name, allergies FROM patients
2 WHERE allergies = "Penicillin" OR allergies = "Morphine"
3 ORDER BY allergies ASC, first_name ASC, last_name ASC

```

The results table has columns: first_name, last_name, and allergies. The data includes:

first_name	last_name	allergies
Adam	Cooper	Morphine
Anthony	Tony	Morphine
Cash	Fox	Morphine
Dotty	McClary	Morphine
Jack	Osborne	Morphine
Jane	Foster	Morphine

Showing 1 to 100 of 1104 results

On the right, there's a 'Solution' section with the same query and a message 'You Got It Correct'.

- Show patient_id, primary_diagnosis from admissions. Find patients admitted multiple times for the same primary_diagnosis
- Query :

Listing 19: Medium - 8

```

1 select patient_id, primary_diagnosis from admissions
2 group by patient_id, primary_diagnosis
3 having count(*) > 1

```

- Baris 1 : Mengambil data patient_id dan primary_diagnosis dari tabel admissions
- Baris 2 : Mengelompokkan berdasarkan patient_id, lalu Mengelompokkan berdasarkan primary_diagnosis
- Baris 3 : Mengambil data dalam kelompok yang memiliki anggota lebih dari 1 (pasien yang sama terkena penyakit yang sama lebih dari sekali)

Screenshot :

The screenshot shows a computer screen with a browser window open to 'sql-practice.com'. The main content area displays a SQL query:

```
1 SELECT patient_id, primary_diagnosis FROM admissions
2 GROUP BY patient_id, primary_diagnosis
3 HAVING COUNT(*) > 1
```

Below the query is a table with the following data:

patient_id	primary_diagnosis
393	Fractured Tibia
396	Congestive Heart Failure
859	Appendicitis
1153	Abdominal Pain
1313	Congestive Heart Failure
1362	Appendicitis
1378	Congestive Heart Failure
1588	Severe Pain + Tenderness In Left Calf

To the right of the table, there is a green box with the text 'You Got The Query Correct' and a button labeled 'View All Questions'. Below the table, another green box says 'Our Solution' and shows the same query. At the bottom right, there is a message 'You Got It Correct' and 'Congratulations on solving the question.'

- Show the city and the total number of patients in the city in the order from most to least patients

Query :

Listing 20: Medium - 9

```
1 select city, count(*) as num_patients from patients
2 group by city
3 order by num_patients DESC
```

- Baris 1 : Mengambil data city dan banyaknya data (akan disebut sebagai num_patients) dari tabel patients
- Baris 2 : Mengelompokkan data berdasarkan city
- Baris 3 : Mengurutkan data berdasarkan num_patients secara descending

Screenshot :

The screenshot shows a web browser window with multiple tabs open. The active tab is on 'sql-practice.com' and displays a SQL query and its results. The query is:

```
1 SELECT city, COUNT(*) AS num_patients FROM patients
2 GROUP BY city
3 ORDER BY num_patients DESC
```

The results table shows the following data:

city	num_patients
Hamilton	1938
Toronto	317
Burlington	276
Brantford	147
Ancaster	117
Stoney Creek	107
Dundas	79
Cambridge	79

On the right side of the screen, there are several green boxes with messages: 'You Got The Query Correct', 'Our Solution' (showing the same query), and 'You Got It Correct' with the message 'Congratulations on solving the question.' Below the browser window, the Windows taskbar is visible.

- Show first name, last name and role of every person that is either patient or physician. The roles are either "Patient" or "Physician"

Query :

Listing 21: Medium - 10

```
1 alter table patients
2 add Role varchar(100);
3 Alter table physicians
4 add Role varchar(100);
5 update patients
6 set Role = "Patient";
7 Update physicians
8 set Role = "Physician";
9 insert into patients (first_name, last_name, Role, Gender)
10 select first_name, last_name, Role, Role from physicians;
11 select first_name, last_name, Role from patients
12 order by first_name ASC, last_name ASC
13
14 'CARA LAMBAT'
```

- Baris 1 : Mengambil tabel patients
- Baris 2 : Menambahkan kolom Role sebagai tipe data varchar
- Baris 3 : Mengambil tabel physicians
- Baris 4 : Menambahkan kolom Role sebagai tipe data varchar
- Baris 5 : Mengupdate tabel patients
- Baris 6 : Mengubah nilai Role pada tabel patients dengan "Patient"
- Baris 7 : Mengupdate tabel patients
- Baris 8 : Mengubah nilai Role pada tabel physicians dengan "Physician"

- Baris 9 - 10 : Menambahkan nilai dari tabel physicians ke tabel patients
- Baris 11 : Mengambil data first_name, last_name, dan Role dari tabel patients
- Baris 12 : Mengurutkan berdasarkan first_name secara ascending, lalu mengurutkan berdasarkan last_name secara ascending

Screenshot :

The screenshot shows a browser window with several tabs open, including "Radio FM 101.3", "WhatsApp", "Paid Promote - Go", "586 How to Fix...", "Sharpen AI - task", and "Learn SQL - Online". The main content area is a SQL editor with the following code:

```

1 ALTER TABLE patients
2 ADD Role varchar(100);
3 ALTER TABLE physicians
4 ADD Role varchar(100);
5 UPDATE patients
6 SET Role = "Patient";
7 SET Role = "Physician";
8 SET Role = "Physician";
9 INSERT INTO patients(first_name, last_name, Role, Gender)
10 SELECT first_name, last_name, Role, Role FROM physicians;
11 SELECT first_name, last_name, Role FROM patients
12 ORDER BY first_name ASC, last_name ASC
  
```

Below the code, there is a table with the following data:

first_name	last_name	Role
Abby	Townsend	Patient
Abe	Brown	Patient
Abe	Carver	Patient
Abe	Lyntley	Patient
Abe	Owens	Patient
Abe	Polin	Patient

The message "You Got The Query Correct" is displayed above the table, and "Congratulations on solving the question." is at the bottom.

- Show all allergies ordered by popularity. Remove 'NKA' and NULL values from query

Query :

Listing 22: Medium - 11

```

1 select allergies, Count(*) as total_diagnose from patients
2 where not (allergies isNULL OR allergies is "NKA")
3 group by allergies
4 order by count(*) DESC
  
```

- Baris 1 : Mengambil data allergies dan banyaknya data sebagai total_diagnose dari tabel patients
- Baris 2 : Dengan kondisi dimana allergies yang dimiliki adalah selain NKA dan tidak bernilai null
- Baris 3 : Mengelompokkan data berdasarkan allergies
- Baris 4 : Mengurutkan data berdasarkan banyaknya data dari tiap kelompok secara descending

Screenshot :

The screenshot shows a browser window with the URL sql-practice.com. The page displays a SQL query editor and a results table. The query is:

```
1 SELECT allergies, COUNT(*) AS total_diagnose FROM patients
2 WHERE NOT (allergies ISNULL OR allergies IS "NKA")
3 GROUP BY allergies
4 ORDER BY COUNT(*) DESC
```

The results table shows the following data:

allergies	total_diagnose
Penicillin	1082
Codeine	305
Sulfa	157
ASA	99
Sulfa Drugs	71
Peanuts	52
Iodine	48
Tylenol	42

On the right side of the screen, there is a sidebar with buttons for 'View All Questions', 'Next Question', 'Easy', and 'Our Solution'. The 'Our Solution' section contains the same SQL query as the editor. A green box at the bottom right says 'You Got It Correct'.

- Show all patient's first_name, last_name, and birth_date who were born in the 1970s decade. Sort the list starting from the earliest birth_date.

Query :

Listing 23: Medium - 12

```
1 select first_name, last_name, height from patients
2 where year(birth_date) % 1970 < 10
3 order by birth_date asc
```

- Baris 1 : Mengambil data first_name, last_name, dan height dari tabel patients
- Baris 2 : Dengan kondisi dimana modulus dari tahun lahir dengan 1970 adalah kurang dari 10 (1970 - 1979)
- Baris 3 : Mengurutkan data berdasarkan birth_date secara ascending

Screenshot :

The screenshot shows a web-based SQL editor interface. The URL is sql-practice.com. The query entered is:

```
1 SELECT first_name, last_name, birth_date FROM patients
2 WHERE YEAR(birth_date) % 1970 < 10
3 ORDER BY birth_date ASC
```

A tooltip on the right side of the interface provides the following information:

Show all patient's first_name, last_name, and birth_date who were born in the 1970s decade. Sort the list starting from the earliest birth_date.

The results table displays the following data:

first_name	last_name	birth_date
Carol	Cunningham	1970-01-14
Martha	Jones	1970-01-18
Vin	Gonzales	1970-01-26
Hyman	Roth	1970-02-03
Emma	Scribbs	1970-02-11
Jean-Baptiste	Adamsberg	1970-02-17

Below the table, it says "Showing 1 to 100 of 621 results".

- We want to display each patient's full name in a single column. Their last_name in all upper letters must appear first, then first_name in all lower case letters. Separate the last_name and first_name with a comma. Order the list by the first_name in descending order. EX: SMITH,jane
Query :

Listing 24: Medium - 13

```
1 select concat(Upper(last_name), ", ", lower(first_name)) as
   new_name_format from patients
2 order by first_name desc
```

- Baris 1 : Mengambil data gabungan dari last_name yang diubah uppercase, tanda ”,”, dan first_name yang diubah lowercase (akan disebut sebagai new_name_format) dari tabel patients
- Baris 2 : Mengurutkan data berdasarkan first_name secara descending

Screenshot :

The screenshot shows a browser window with the URL sql-practice.com. The page displays a SQL query and its execution results. The query is:

```
1 SELECT CONCAT(UPPER(last_name)," ",LOWER(first_name)) AS new_name_format
2 ORDER BY first_name DESC
```

The results table has a single column named "new_name_format" and contains the following data:

new_name_format
HERIOT.zoe
DAVID.ziva
BRENNAN.zenigata
HELM.zenigata
MOSSES.zen
THOMAS.zelda

Below the table, it says "Showing 1 to 100 of 4530 results". To the right of the results, there are buttons for "View All Questions", "Next Question", "Medium", and "Our Solution". The "Our Solution" button is highlighted. At the bottom right, there is a green box saying "You Got It Correct" and "Congratulations on solving the question".

- Show the cities where the patient's average weight, rounded-up, is less than 70kg. Sort the list by highest to lowest avg_weight.

Query :

Listing 25: Medium - 14

```
1 select city, ceil(avg(weight)) as avg_weight from patients
2 group by city
3 having ceil(avg(weight)) < 70
4 order by avg_weight desc
```

- Baris 1 : Mengambil data city, pembulatan keatas dari rata-rata weight (akan disebut sebagai avg_weight) dari tabel patients
- Baris 2 : Mengelompokkan data berdasarkan city
- Baris 3 : Dengan kondisi dimana hasil pembulatan keatas dari rata-rata weight kurang dari 70 kg
- Baris 4 : Mengurutkan data berdasarkan avg_weight secara descending

Screenshot :

The screenshot shows a web browser window for 'sql-practice.com'. The URL bar has 'sql-practice.com' and a search term 'Kuliah'. The page title is 'Bacan'. The main content area displays a SQL query and its results. The query is:

```
1 SELECT city, CEIL( AVG(weight) ) AS avg_weight FROM patients
2 GROUP BY city
3 HAVING CEIL( AVG(weight) ) < 70
4 ORDER BY avg_weight DESC
```

The results table shows the following data:

city	avg_weight
Carlisle	69
Carlisle	68
Elmwood	67
Winona	65
Kapuskasing	65
Chicoutimi	65
Brandon	65
Whitby	64

On the right side of the page, there is a 'Our Solution' section with the same query and results.

- Show the province_id(s) where the total sum of its patient's height is greater than or equal to 7,000.

Query :

Listing 26: Medium - 15

```
1 select province_id, sum(height) as sum_height from patients
2 group by province_id
3 having sum_height > 7000
```

- Baris 1 : Mengambil data province_id, jumlah dari height (akan disebut sebagai sum_height) dari tabel patients
- Baris 2 : Mengelompokkan data berdasarkan province_id
- Baris 3 : Dengan kondisi dimana sum_height lebih dari 7000

Screenshot :

The screenshot shows a web browser window for 'sql-practice.com'. The URL bar has 'sql-practice.com' and a search term. The page title is 'sql-practice.com'. The main content area displays a SQL query and its results. The query is:

```
1 SELECT province_id, SUM(height) AS sum_height
2 GROUP BY province_id
3 HAVING sum_height > 7000
```

The results table shows the following data:

province_id	sum_height
AB	7059
BC	7709
NS	9889
ON	679693

On the right side of the results, there is a green box saying 'You Got The Query Correct.' Below it are buttons for 'View All Questions', 'Next Question', and 'Medium'. Underneath the results, there is a 'Our Solution' section with the same query code.

- Show the difference between the largest weight and smallest weight for patients with the last name 'Maroni'

Query :

Listing 27: Medium - 16

```
1 select max(weight) - min(weight) as weight_delta from patients
2 where last_name is "Maroni"
```

- Baris 1 : Mengambil data selisih dari nilai maximal dengan nilai minimal dari kolom weight (akan disebut sebagai weight_delta) dari tabel patients
- Baris 2 : Dengan kondisi dimana last_name dari pasien adalah "Maroni"

Screenshot :

The screenshot shows a browser window for sql-practice.com. In the left panel, under 'SQL Database', there is a code editor with the following SQL query:

```
1 SELECT MAX(weight) - MIN(weight) AS weight_delta FROM patients
2 WHERE last_name IS "Haroni"
```

The results table shows one row with the column 'weight_delta' containing the value '99'. On the right side of the screen, there is a sidebar with buttons for 'View All Questions', 'Next Question', and 'Our Solution'. The 'Our Solution' section displays the same query. A green message box at the bottom right says 'You Got It Correct'.

- Based on the cities that our patients live in, show unique cities that are in province_id 'NS'?

Query :

Listing 28: Medium - 17

```
1 select distinct(city) from patients
2 where province_id is "NS"
```

- Baris 1 : Mengambil data yang unik pada kolom city dari tabel patients
- Baris 2 : Dengan kondisi dimana province_id dari pasien adalah "NS"

Screenshot :

The screenshot shows a browser window for sql-practice.com. In the left panel, under 'SQL Database', there is a code editor with the following SQL query:

```
1 SELECT DISTINCT(city) FROM patients
2 WHERE province_id IS "NS"
```

The results table shows four rows with the column 'city' containing 'Inverness', 'Port Hawkesbury', and 'Halifax'. On the right side of the screen, there is a sidebar with buttons for 'View All Questions', 'Next Question', and 'Our Solution'. The 'Our Solution' section displays the same query. A green message box at the bottom right says 'You Got It Correct'.

3 Soal Hard

- Show all of the patients grouped into weight groups. Show the total amount of patients in each weight group. Order the list by the weight group descending. For example, if they weight 100 to 109 they are placed in the 100 weight group, $110-119 = 110$ weight group, etc

Query :

Listing 29: Hard - 1

```
1 alter table patients
2 add weight_group int;
3 update patients
4 set weight_group =
5 case
6 when (weight >= 0 and weight < 10) then 0
7 when (weight >= 10 and weight < 20) then 10
8 when (weight >= 20 and weight < 30) then 20
9 when (weight >= 30 and weight < 40) then 30
10 when (weight >= 40 and weight < 50) then 40
11 when (weight >= 50 and weight < 60) then 50
12 when (weight >= 60 and weight < 70) then 60
13 when (weight >= 70 and weight < 80) then 70
14 when (weight >= 80 and weight < 90) then 80
15 when (weight >= 90 and weight < 100) then 90
16 when (weight >= 100 and weight < 110) then 100
17 when (weight >= 110 and weight < 120) then 110
18 when (weight >= 120 and weight < 130) then 120
19 when (weight >= 130 and weight < 140) then 130
20 end;
21 select count(*) as patients_in_group, weight_group from
patients
22 group by weight_group
23 order by weight_group DESC
24
25 'CARA LAMBAT'
```

- Baris 1 : Mengambil data table patients
- Baris 2 : Menambahkan kolom weight_group sebagai tipe data integer pada table patients
- Baris 3 : Mengupdate table patients
- Baris 4 - 20 : Mengubah nilai weight_group sesuai dengan berat pasien (menurut aturan dari soal)
- Baris 21 : Mengambil banyaknya data (akan disebut sebagai patients.in_group) dan weight_group dari tabel patients
- Baris 22 : Mengelompokkan data berdasarkan weight_group
- Baris 23 : Mengurutkan data berdasarkan weight_group secara descending

Screenshot :

The screenshot shows a computer screen with a browser window open to a SQL practice website. The URL is sql-practice.com. The page has a navigation bar with tabs for 'Run', 'Support', and 'Light Mode'. Below the navigation bar, there are sections for 'SQL Database' and 'Learning Resources'. A code editor window displays the following SQL code:

```
1 ALTER TABLE patients
2 ADD weight_group int;
3 UPDATE patients
4 SET weight_group =
5 CASE
6 WHEN (weight>= 0 AND weight < 10) THEN 0
7 WHEN (weight>= 10 AND weight < 20) THEN 10
8 WHEN (weight>= 20 AND weight < 30) THEN 20
9 WHEN (weight>= 30 AND weight < 40) THEN 30
10 WHEN (weight>= 40 AND weight < 50) THEN 40
11 WHEN (weight>= 50 AND weight < 60) THEN 50
12 WHEN (weight>= 60 AND weight < 70) THEN 60
13 WHEN (weight>= 70 AND weight < 80) THEN 70
14 WHEN (weight>= 80 AND weight < 90) THEN 80
15 WHEN (weight>= 90 AND weight < 100) THEN 90
```

Below the code is a table with two columns: 'patients_in_group' and 'weight_group'. The data is as follows:

patients_in_group	weight_group
53	130
207	120
445	110
510	100
396	90
490	80
624	70
727	60

To the right of the table, there is a 'Solution' section with the following text:

```
SELECT COUNT(*) AS patients_in_group, weight/100
GROUP BY weight_group
ORDER BY weight_group DESC
-- because weight is a integer, dividing by 10
-- at weight uses a decimal number we could solve
```

At the bottom right of the screen, there is a status bar showing the date and time: 19:05 24/04/2022.

- Show patient_id, weight, height, isObese from the patients table. Display isObese as a boolean 0 or 1. Obese is defined as $\text{weight}(\text{kg})/(\text{height}(\text{m})^2) \geq 30$. weight is in units kg. height is in units cm

Query :

Listing 30: Hard - 2

```
1 select patient_id, weight, height, (cast(weight as float) /
power((cast(height as float)/100),2) >=30) as isObese from
patients
```

- Baris 1 : Mengambil data patient_id, weight, height, dan $(\text{weight}/(\text{height}/100)^2) \geq 30$ (akan disebut sebagai isObese) dari tabel patients

Screenshot :

The screenshot shows a computer screen with a browser window open to a SQL practice website. The URL is sql-practice.com. The page displays a SQL query and its execution results.

SQL Query:

```
1 SELECT patient_id, weight, height, (CAST(weight AS float) / CAST(POWER((G
```

Execution Result:

patient_id	weight	height	isObese
1	93	181	0
2	100	172	1
3	31	101	1
4	80	176	0
5	83	168	0
6	71	144	1

Showing 1 to 100 of 4530 results

Our Solution:

```
SELECT patient_id, weight, height,  
CASE  
    WHEN weight/(POWER(height/100,0.25)) >= 30  
    THEN 1  
    ELSE 0  
END AS isObese  
FROM patients;
```

You Got The Query Correct

You Got It Correct

Congratulations on solving the question.

- Show patient_id, first_name, last_name, and attending physician's specialty. Show only the patients who has a primary_diagnosis as 'Dementia' and the physician's first name is 'Lisa'. Check patients, admissions, and physicians tables for required information.

Query :

Listing 31: Hard - 3

```
1 select patients.patient_id, patients.first_name, patients.  
      last_name, physicians.specialty  
2 from ((admissions  
3 inner join patients  
4 on admissions.patient_id = patients.patient_id)  
5 Inner join physicians  
6 on admissions.attending_physician_id = physicians.physician_id  
    )  
7 where primary_diagnosis is "Dementia" and physicians.  
      first_name is "Lisa"
```

- Baris 1 : Mengambil data patient_id, first_name, last_name, dan specialty
- Baris 2 - 4 : Dari tabel admissions yang digabung dengan tabel patients dengan acuan patient_id pada tabel admissions merujuk ke patient_id pada tabel patients
- Baris 5 - 6 : Lalu digabung lagi dengan tabel physicians dengan acuan attending_physician_id pada tabel admissions merujuk ke physician_id pada tabel physicians
- Baris 7 : Dengan kondisi dimana primary_diagnosis dari pasien adalah "Dementia" dan first_name dari physician adalah "Lisa"

Screenshot :

The screenshot shows a computer screen with a browser window open to 'sql-practice.com'. The main content area displays a SQL query and its execution results. The query is:

```
SELECT p.patient_id, patient.first_name, patient.last_name, physician.specialty
FROM admissions A
JOIN patients p ON A.patient_id = p.patient_id
JOIN physicians ph ON A.attending_physician_id = ph.physician_id
WHERE primary_diagnosis = 'Dementia' AND physician.first_name = 'Lisa'
```

The results table shows two rows:

patient_id	first_name	last_name	specialty
1	Miyuki	Riviera	Obstetrician/Gynecologist
3994	Sam	Parkinson	Obstetrician/Gynecologist

On the right side of the interface, there are several buttons: 'Run', 'Support', 'Light Mode', 'Left Menu', 'Right Menu', 'You Got The Query Correct', 'View All Questions', 'Next Question', 'Easy', 'Our Solution', and a message box stating 'You Got It Correct' with the message 'Congratulations on solving the question.' Below the message box is a timestamp: '19:07 24/04/2022'.

- All patients who have gone through admissions, can see their medical documents on our site. Those patients are given a temporary password after their first admission. Show the patient_id and temp_password. The password must be the following, in order: 1. patient_id 2. the numerical length of patient's last_name 3. year of patient's birth_date

Query :

Listing 32: Hard - 4

```
1 select distinct(admissions.patient_id), concat(admissions.
    patient_id,LEN(patients.last_name),year(patients.
    birth_date)) as temp_password
2 from admissions
3 inner join patients
4 on admissions.patient_id = patients.patient_id
```

- Baris 1 : Mengambil data unik dari kolom patient_id, gabungan dari patient_id, panjang last_name dan tahun lahir (akan disebut sebagai temp_password)
- Baris 2 : Dari tabel admissions
- Baris 3 : Menggabungkan tabel patients sesuai dengan data pada tabel admissions
- Baris 4 : Dengan acuan patient_id pada tabel admissions merujuk ke patient_id pada tabel patients

Screenshot :

The screenshot shows a computer screen with a browser window open to a SQL practice website. The URL is sql-practice.com. The page displays a SQL query editor with the following code:

```

1 SELECT DISTINCT(admissions.patient_id), CONCAT(admissions.patient_id,LENK)
2 FROM admissions
3 INNER JOIN patients
4 ON admissions.patient_id = patients.patient_id
    
```

Below the code is a table with two columns: patient_id and temp_password. The data is as follows:

patient_id	temp_password
1	171953
3	3102011
6	671989
7	761929
8	861978
9	961963

At the bottom of the table, it says "Showing 1 to 100 of 3382 results". To the right of the table, there are navigation buttons for "Previous", "1", "2", "3", "...", "34", and "Next".

On the right side of the screen, there are several status indicators and a system tray at the bottom.

- Each admission costs 50 dollars for patients without insurance, and 10 dollars for patients with insurance. All patients with an even patient_id have insurance. Give each patient a 'Yes' if they have insurance, and a 'No' if they don't have insurance. Add up the admission.total cost for each has_insurance group.

Query :

Listing 33: Hard - 5

```

1 alter table admissions
2 add column [has_insurance] varchar(3);
3 alter table admissions
4 add column [cost_after_insurance] int;
5 update admissions
6 set has_insurance =
7 (case
8 when patient_id % 2 = 0 then "Yes"
9 when patient_id % 2 != 0 then "No"
10 end),
11 cost_after_insurance =
12 (case
13 when patient_id % 2 = 0 then 10
14 when patient_id % 2 != 0 then 50
15 end);
16 select has_insurance, sum(cost_after_insurance) from
17 admissions
18 group by has_insurance
19 'CARA LAMBAT'
    
```

- Baris 1 : Mengambil data tabel admissions
- Baris 2 : Menambahkan kolom has_insurance sebagai tipe data varchar

- Baris 3 : mengambil data tabel admissions
- Baris 4 : Menambahkan kolom cost_after_insurance sebagai tipe data integer
- Baris 5 : Mengupdate tabel admissions
- Baris 6 - 10 : Mengubah nilai dari has_insurance sesuai dengan patient_id, ketika patient_id genap maka bernilai "Yes", ketika patient_id ganjil maka bernilai "No"
- Baris 11 - 15 : Mengubah nilai dari cost_after_insurance sesuai dengan patient_id, ketika patient_id genap maka bernilai 10, ketika patient_id ganjil maka bernilai 50
- Baris 16 : Mengambil data has_insurance dan jumlah dari cost_after_insurance dari tabel admissions
- Baris 17 : Mengelompokkan data berdasarkan has_insurance

Screenshot :

The screenshot shows a Windows desktop with several open windows. In the center is a SQL editor window titled 'SQL Database'. The code input area contains:

```

1 ALTER TABLE admissions
2 ADD COLUMN [has_insurance] varchar(1);
3 ALTER TABLE admissions
4 ADD COLUMN [cost_after_insurance] int;
5 UPDATE admissions
6 SET has_insurance =
7 CASE
8 WHEN patient_id % 2 = 0 THEN "Yes"
9 WHEN patient_id % 2 != 0 THEN "No"
10 END;
11 cost_after_insurance =
12 CASE
13 WHEN patient_id % 2 = 0 THEN 10
14 WHEN patient_id % 2 != 0 THEN 50
15 END;
16
17 has_insurance      sum(cost_after_insurance)
18 No                  127800
19 Yes                 25110

```

To the right of the code, there's a 'Our Solution' section with the following SQL code:

```

SELECT
CASE WHEN patient_id % 2 = 0 THEN
    'Yes'
ELSE
    'No'
END AS has_insurance,
SUM(CASE WHEN patient_id % 2 = 0 THEN
    cost_after_insurance
ELSE
    0
END) AS cost_after_insurance
FROM admissions
GROUP BY has_insurance;

```

At the bottom right of the editor, a green box says 'You Got It Correct! Congratulations on solving the question.'

- Show the province that has more patients identified as 'M' than 'F'. Must only show full province_name

Query :

Listing 34: Hard - 6

```

1 select provinces.province_name from patients
2 inner join provinces
3 on patients.province_id = provinces.province_id
4 group by patients.province_id
5 having count(case when gender is "M" then 1 end) > count(case
when gender is "F" then 1 end)

```

- Baris 1 : Mengambil data province_name dari tabel patients

- Baris 2 : Menggabungkan data pada tabel provinces sesuai dengan data pada tabel patients
- Baris 3 : Dengan acuan province_id pada tabel patients merujuk ke province_id pada tabel provinces
- Baris 4 : Mengelompokkan data berdasarkan province_id
- Baris 5 : Dengan kondisi dimana banyaknya data dengan gender ”M” lebih banyak dibandingkan data dengan gender ”F”

Screenshot :

```

1 SELECT provinces.province_name FROM patients
2 INNER JOIN provinces
3 ON patients.province_id = provinces.province_id
4 GROUP BY patients.province_id
5 HAVING COUNT(CASE WHEN gender IS "M" THEN 1 END) > COUNT(CASE WHEN gender IS "F" THEN 1 END)
  
```

province_name
Alberta
British Columbia
Manitoba
Newfoundland and Labrador
Nova Scotia
Ontario
Saskatchewan

- We are looking for a specific patient. Pull all columns for the patient who matches the following criteria:
 - First_name contains an ’r’ after the first two letters.
 - Identifies their gender as ’F’
 - Born in February, May, or December
 - Their weight would be between 60kg and 80kg
 - Their patient_id is an odd number
 - They are from the city ’Halifax’

Query :

Listing 35: Hard - 7

```

1 select * from patients
2 where (first_name like "__r%")
3 and (gender is "F")
4 AND (month(birth_date) in (02,03,12))
5 and (weight between 60 and 80)
6 and (patient_id % 2 != 0)
7 and (city is "Halifax")
  
```

- Baris 1 : Mengambil semua data dari tabel patients
- Baris 2 : Dengan kondisi dimana first_name yang mengandung huruf "r" pada huruf ketiga
- Baris 3 : Dan bergender "F"
- Baris 4 : Dan memiliki bulan lahir yang termasuk dalam (2, 3, 12)
- Baris 5 : Dan memiliki weight diantara 60 dan 80
- Baris 6 : Dan memiliki patient_id ganjil
- Baris 7 : Dan berasal dari kota "Halifax"

Screenshot :

The screenshot shows a SQL practice interface. On the left, there's a code editor with the following SQL query:

```

1 SELECT * FROM patients
2 WHERE (first_name LIKE "%_r%")
3 AND (gender IS "F")
4 AND (MONTH(birth_date) IN (02,03,12))
5 AND (weight BETWEEN 60 AND 80)
6 AND (patient_id % 2 = 1)
7 AND (city IS "Halifax")
  
```

On the right, the results of the query execution are displayed in a table:

patient_id	first_name	last_name	gender	birth_date	city
989	Sarah	Hamilton	F	1994-02-08	Halifax

Below the table, the application displays "You Got The Query Correct" and "Our Solution" with the same query. At the bottom right, it says "You Got It Correct" and "Congratulations on solving the question."

- Show the percent of patients that have 'M' as their gender. Round the answer to the nearest hundredth number and in percent form.
- Query :

Listing 36: Hard - 8

```

1 select concat(Round(cast(count(case when gender is "M" then 1
end) as float) / cast(count(*) as float) * 100,2), "%") as
percent_of_male_patients from patients
  
```

- Baris 1 : Mengambil data gabungan dari pembulatan pada 2 angka dibelakang koma dari perhitungan banyaknya data bergender "M" dibagi total banyaknya data dikali 100 (perhitungan persentase), dan tanda "%" (akan disebut sebagai percent_of_male_patients) dari tabel patients

Screenshot :

The screenshot shows a computer screen with a Windows desktop environment. A browser window is open at sql-practice.com, specifically on the 'Bacan' page. The main content area displays a SQL query:

```
1 SELECT CONCAT(ROUND(CAST(COUNT(CASE WHEN gender IS "M" THEN 1 END) AS float), 2)) AS percent_of_male_patients
FROM patients;
```

The result of the query is shown in a table:

percent_of_male_patients
54.48%

A green notification bar at the top right says "You Got The Query Correct". To the right of the result table, there are buttons for "View All Questions", "Next Question", "Hard", and "Our Solution". The status bar at the bottom of the browser window shows the time as 21:38 and the date as 24/04/2022.