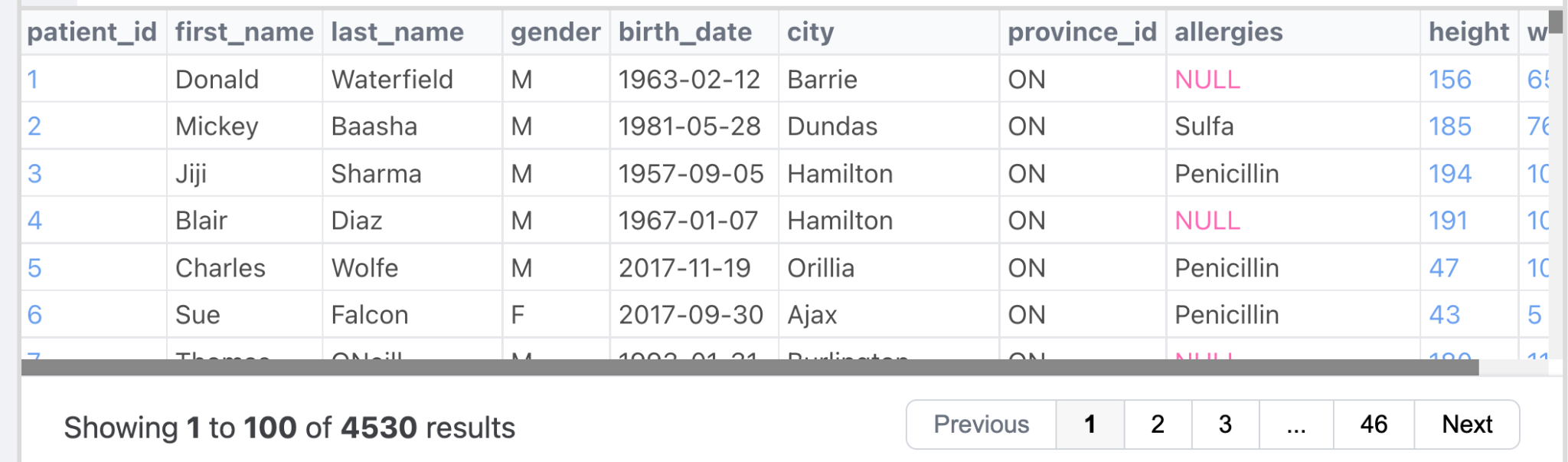
Завдання на практику SQL

Виконуємо на сайті <https://www.sql-practice.com/> використовуючи базу, яка є там. Для виконання кожного завдання необхідно текстом вставити свій SQL запит, який виконує задане завдання та скріншот результату, разом з кількістю вкладок і кількістю результатів (якщо ця інформація вивелась на екран).

Наприклад.

0. вивести всю інформацію про пацієнтів.

SELECT \* FROM patients;



Завдання

1. Виведіть мінімальну і максимальну вагу в наборі даних.

| SELECT  min(weight) as min\_weight,  max(weight) as max\_weight FROM patients ; |
| --- |

1. Виведіть ім'я, прізвище та вагу пацієнта з найбільшою вагою. Результатом має бути лише один рядок.

| SELECT  first\_name,  last\_name,  weight FROM patients order by weight desc limit 1; |
| --- |



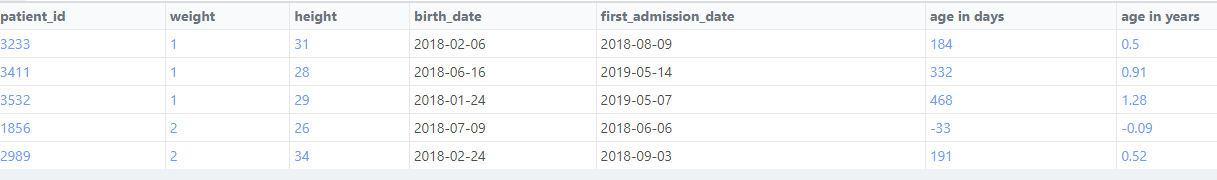
1. Виведіть ідентифікатор пацієнта, вагу, зріст та день народження 5ти пацієнтів з найменшою вагою в наборі даних.

| SELECT  patient\_id,  weight,  height,  birth\_date FROM patients order by weight asc limit 5; |
| --- |

1. Додайте до попереднього результату кількість років пацієнта, округлену до 2 цифр після коми, на момент його звернення в клініку (admission\_date з таблиці admissions). Як обчислити різницю між двома датами в SQLLite можна прочитати [тут](https://learnsql.com/cookbook/how-to-calculate-the-difference-between-two-dates-in-sqlite/).

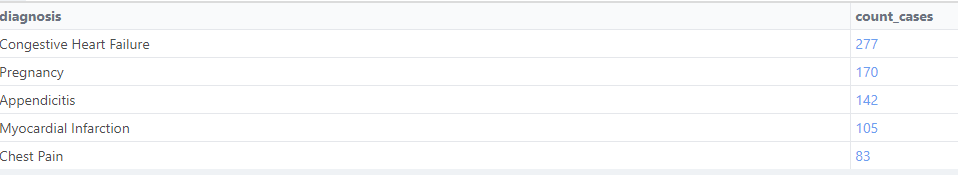
Може вийти так, що вік на момент звернення негативний. Або це баг в даних, або специфіка даних, наприклад, звернулась вагітна жінка - цього ми не дізнаємось.

| SELECT  p.patient\_id,  p.weight,  p.height,  p.birth\_date,  min(a.admission\_date) as first\_admission\_date,  JULIANDAY(min(a.admission\_date)) - JULIANDAY(p.birth\_date) AS 'age in days',  round(  (  JULIANDAY(min(a.admission\_date)) - JULIANDAY(p.birth\_date)  ) / 365,  2  ) AS 'age in years' FROM patients as p  left join admissions as a on a.patient\_id = p.patient\_id group by p.patient\_id order by p.weight asc limit 5; |
| --- |



1. Виведіть TOP 5 діагнозів в базі за кількістю випадків прийомів з ними. Для цього нам треба тільки таблиця admissions. Колонку з кількістю випадків прийомів з діагнозом назвіть count\_cases.

| select  diagnosis,  count(\*) as count\_cases from admissions group by diagnosis order by count\_cases desc limit 5; |
| --- |



1. Виведіть всю інформацію з таблиці admissions для останнього в базі прийому (визначаємо за admission\_date) з найчастішим діагнозом. Тут треба перевикористати запит з попереднього завдання з невеликими змінами в нього. Вписати просто текстом назву діагнозу - в цьому завданні не можна. Треба обрати найчастіший діагноз з допомогою запиту і за ним відфільтрувати дані. В результаті має вийти один рядок.

| select a.\* from admissions as a  inner join (  select diagnosis  from admissions a1  group by a1.diagnosis  order by count(\*) desc  limit 1  )as top\_diagnos on a.diagnosis = top\_diagnos.diagnosis  order by a.admission\_date desc   limit 1; |
| --- |

Очікуваний результат:



1. Напишіть запит, щоб знайти список імен, прізвищ та алергій пацієнтів з міста Hamilton, у яких є алергії (тобто значення allergies не є пустим).

| select  first\_name,  last\_name,  allergies from patients where  city = 'Hamilton'  and allergies is not null; |
| --- |

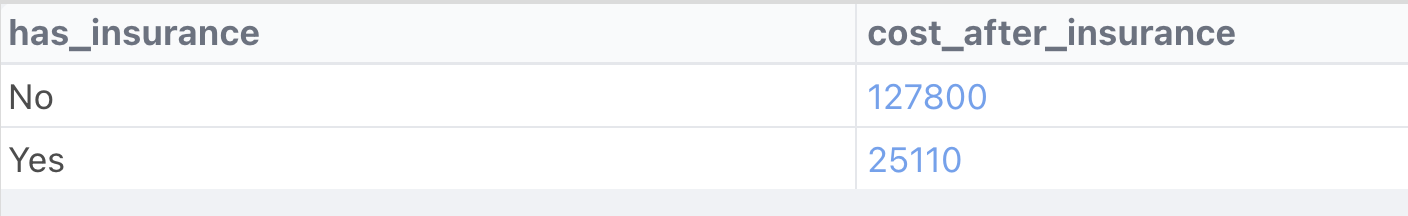
1. Кожен прийом коштує 50 доларів для пацієнтів без страхування та 10 доларів для пацієнтів зі страхуванням. Усі пацієнти з **парним** ідентифікатором пацієнта мають страховку.

Поставте кожному пацієнту «Yes», якщо він має страхування, і «No», якщо він не має страхування, назвіть цю колонку has\_insurance. Загрегуйте дані за значенням has\_insurance та додайте загальну вартість прийомів (admissions) для кожної групи has\_insurance.

Підказка: рішення можна побудувати з використанням CASE або з використанням UNION.

| select  has\_insurance,  case  when has\_insurance = 'Yes' then count(\*) \* 10  else count(\*) \* 50  end as cost\_after\_insurance from admissions a  inner join (  select  patient\_id,  case  when patient\_id % 2 = 0 then 'Yes'  else 'No'  end as has\_insurance  from  patients  ) as insurance on insurance.patient\_id = a.patient\_id group by has\_insurance; |
| --- |

Очікуваний результат:



1. Вивести унікальні імена пацієнтів, які починаються і закінчуються на «s», разом з кількістю таких пацієнтів. Кількість носіїв імен відсортувати за спаданням (від найбільшої кількості, до найменшої).

| select  first\_name,  count(\*) as count\_names from patients where first\_name like 's%s' group by first\_name order by count\_names desc; |
| --- |

1. Виведіть кількість пацієнтів чоловіків та пацієнток жінок за провінціями (province\_name) так, аби кількість чоловіків була в одній колонкці, а кількість жінок - в іншій. Відсортуйте дані за спаданням кількості чоловіків та відфільтруйте дані так, аби увійшли до вибірки лише ті провінції, за якими є більше 20ти пацієнтів (чоловіків і жінок сумарно).

| SELECT  pn.province\_name,  SUM(CASE WHEN p.gender = 'M' THEN 1 ELSE 0 END) AS male\_count,  SUM(CASE WHEN p.gender = 'F' THEN 1 ELSE 0 END) AS female\_count  FROM patients as p  inner join province\_names pn on pn.province\_id = p.province\_id  GROUP BY pn.province\_name   having male\_count + female\_count>20  order by male\_count desc; |
| --- |

Очікуваний результат:

