Form ENCS-SAS (03/04)

Faculty of Engineering and Computer Science
Expectations of Originality

This form has been created to ensure that all students in the Faculty of Engineering and Computer Science comply with principles of academic integrity prior to submitting coursework to their instructors for evaluation: namely reports, assignments, lab reports and/or software. All students should become familiar with the University's Code of Conduct (Academic) located at http://web2.concordia.ca/Legal_Counsel/policies/english/AC/Code.html

Please read the back of this document carefully before completing the section below. This form must be attached to the front of all coursework submitted to instructors in the Faculty of Engineering and Computer Science.

Course Number: COMP 353 **Instructor:** Prof Khaled Jababo

Type of Submission (Please check off reponses to both a & b)

a. ✓ Report __ Assignment __ Lab Report __ Software

b. __ Individual submission ✓ Group Submission (All members of the team must sign below)

Having read both sides of this form, I certify that I/we have conformed to the Faculty's expectations of originality and standards of academic integrity.

Name: Marwan Al-Ghaziri (please print clearly) ID No: 40126554 Signature:  Date: Jul 23 2023

Name: Fahad Abdul Rahman (please print clearly) ID No: 40157997 Signature:  Date: Jul 23 2023

Name: Derek James (please print clearly) ID No: 22688964 Signature: Derek James Date: Jul 23, 2023

Name: Samdarshi Tiwari (please print clearly) ID No: 40113256 Signature: Samdarshi Tiwari Date: Jul 23,2023

Name: _____ (please print clearly) ID No: _____ Signature: _____ Date: _____

Name: _____ (please print clearly) ID No: _____ Signature: _____ Date: _____

Do Not Write in this Space – Reserved for Instructor

COMP 353
Warm Up Project Report

Group Account: ddc353_1

Derek James 22688964

Fahad Abdul Rahman 40157997

Marwan Al-Ghaziri 40126554

Samdarshi Tiwari 40113256

Part 1.Design

1.1 Summary of the system

The Education Personnel Status Tracking System (EPSTS) is an application aimed at helping the Ministry of Education monitor the health status of its personnel and students during the COVID-19 pandemic. The system will maintain crucial information related to infections and vaccinations for each individual associated with the ministry.

For personnel, the system will record details of infections, including the date and type of infection. It will also track vaccination information, such as vaccination date, type, and dose number. The application will handle information about different facilities, including management and educational facilities.

The system will maintain comprehensive details for employees and students, such as their personal information, contact details, Medicare card numbers, and citizenship. It will ensure that everyone is registered with the public health care system, and no two people can have the same Medicare card number.

The system will keep track of each employee's start and end dates for each facility they work at. A student can be registered at only one educational facility at a time, but they may switch between different facilities at various times. The system will maintain the start and end dates for each student's registration and also track their current education level, such as elementary 3 or secondary 2.

In summary, the EPSTS application aims to provide the Ministry of Education with a comprehensive database system to track the health status of its personnel and students during the COVID-19 pandemic. The system will enable effective monitoring and risk reduction across various facilities operated by the ministry.

1.2 E/R diagram of the design

The figure 1 E/R diagram was built using an free online tool called “SmartDraw”, while the figure 2 E/R diagram was built using MySQL Workbench’s reverse engineering feature.

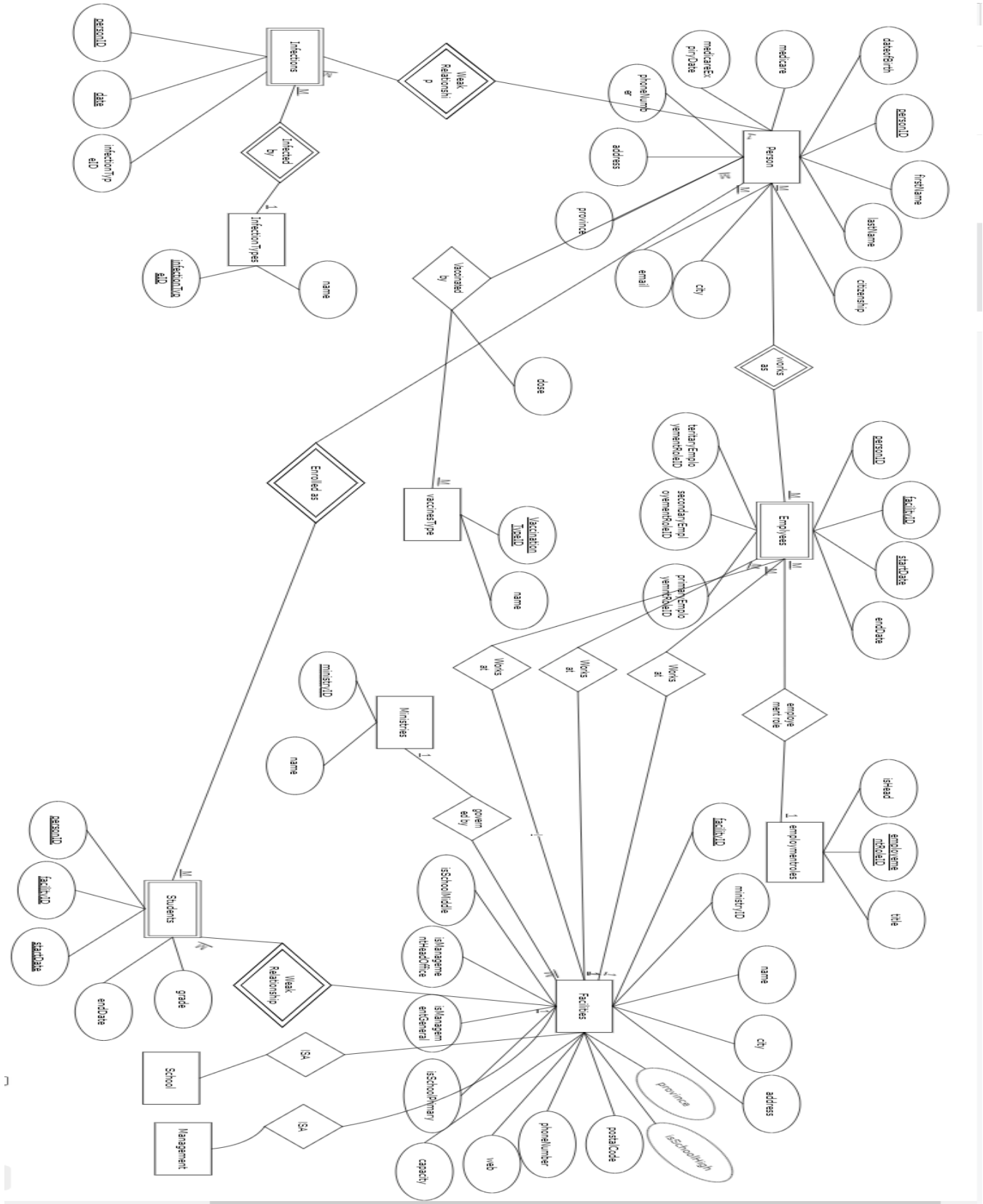


Figure 1 E/R Diagram built using SmartDraw

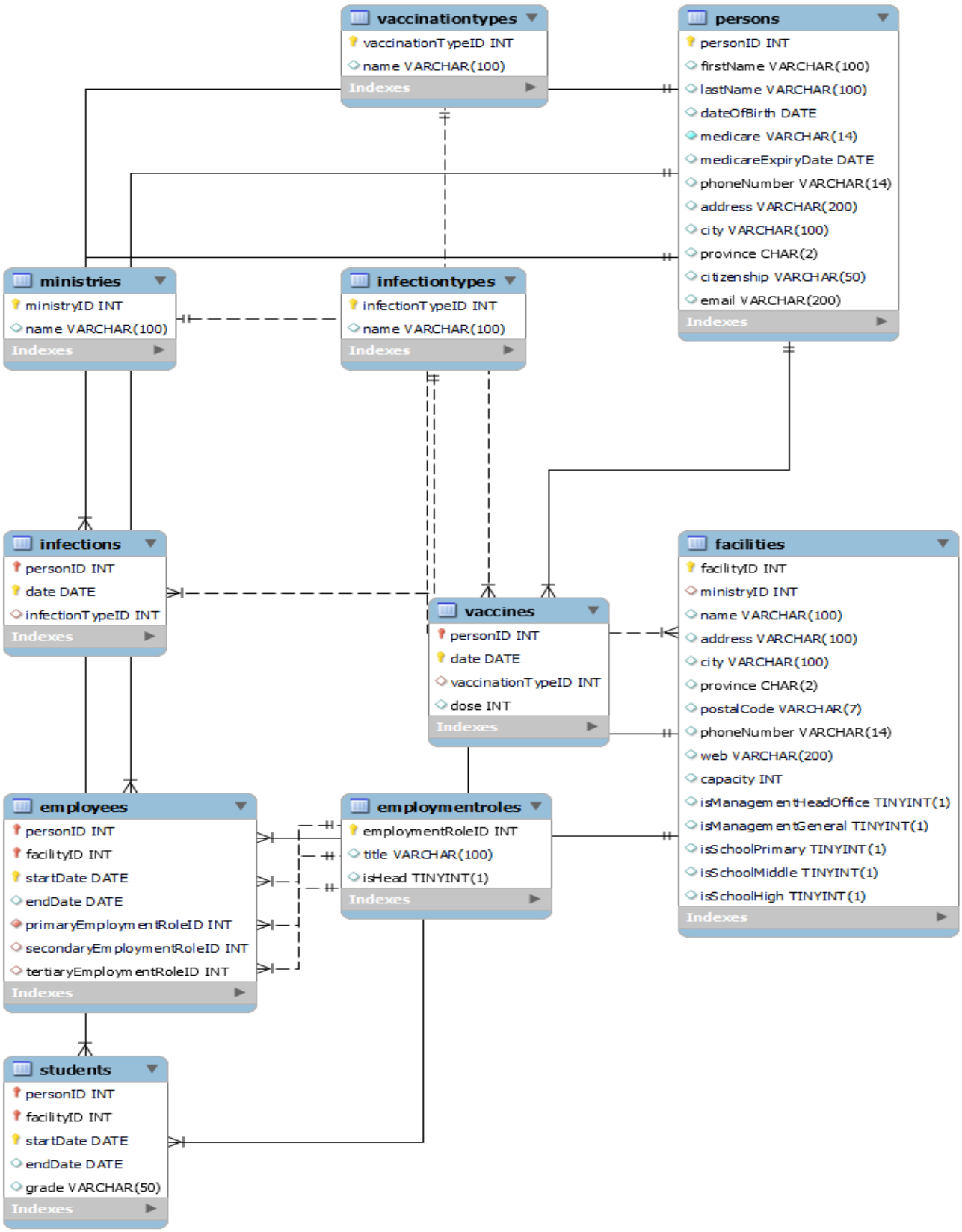


Figure 2 E/R diagram built using MySQL Workbench reverse engineering

The E/R diagram was converted into the following relations:

- Ministries(ministryID, name)
- Facilities(facilityID, ministryID, name, address, city, province, postalCode, phoneNumber, web, capacity, isManagementHeadOffice, isManagementGeneral, isSchoolPrimary, isSchoolMiddle, isSchoolHigh)
- Persons(personID, firstName, lastName, dateOfBirth, medicare, medicareExpiryDate, phoneNumber, address, city, province, citizenship, email)
- EmploymentRoles(employmentRoleID, title, isHead)
- Employees(personID, facilityID, startDate, endDate, primaryEmploymentRoleID, secondaryEmploymentRoleID, tertiaryEmploymentRoleID)
- Students(personID, facilityID, startDate, endDate, grade)
- InfectionTypes(infectionTypeID, name)
- Infections(personID, date, infectionTypeID)
- VaccinationTypes(vaccinationTypeID, name)
- Vaccines(personID, date, vaccinationTypeID, dose)

1.3 Design decisions

The Education Personnel Status Tracking System (EPSTS) project's schema includes tables for managing information related to persons, facilities, employment roles, students, infections, and vaccinations. Here's a summary of the tables and their key attributes:

The `Ministries` table stores information about different ministries, with each ministry having a unique `ministryID` and a `name`

The `Facilities` table stores information about different facilities, with each facility having a unique `facilityID`, a `name`, and other attributes such as `address`, `city`, `province`, `postalCode`, `phoneNumber`, `web`, and `capacity`. Each facility is also associated with a ministry through the foreign key `ministryID` that references the primary key `ministryID` in the `Ministries` table.

The `Persons` table stores information about different persons, with each person having a unique `personID` and attributes such as `firstName`, `lastName`, `dateOfBirth`, `medicare`, `medicareExpiryDate`, `phoneNumber`, `address`, `city`, `province`, `citizenship`, and `email`

The `EmploymentRoles` table stores information about different employment roles, with each role having a unique `employmentRoleID`, a `title`, and a boolean column to indicate whether the role is a head role

The `Employees` table stores information about the employment of persons at different facilities. Each record in the table represents an employment of a person at a facility for a specific period of

time (`startDate` to `endDate`). The primary key of the table is a composite key consisting of the columns `personID`, `facilityID`, and `startDate`

The `Students` table stores information about the enrollment of persons at different facilities. Each record in the table represents an enrollment of a person at a facility for a specific period of time (`startDate` to `endDate`). The primary key of the table is a composite key consisting of the columns `personID`, `facilityID`, and `startDate`.

The `InfectionTypes` table stores information about different types of infections, with each type having a unique `infectionTypeID` and a `name`.

The `Infections` table stores information about infections of persons. Each record in the table represents an infection of a person on a specific date. The primary key of the table is a composite key consisting of the columns `personID` and `date`.

The `VaccinationTypes` table stores information about different types of vaccinations, with each type having a unique `vaccinationTypeID` and a `name`.

The `Vaccines` table stores information about vaccinations of persons. Each record in the table represents a vaccination of a person on a specific date. The primary key of the table is a composite key consisting of the columns `personID` and `date`.

Part 2: Statements to Create Database

```
CREATE TABLE Ministries (
	ministryID int PRIMARY KEY,
	name varchar(100)
);

CREATE TABLE Facilities (
	facilityID int PRIMARY KEY,
	ministryID int,
	name varchar(100),
	address varchar(100),
	city varchar(100),
	province char(2),
	postalCode varchar(7),
	phoneNumber varchar(14),
	web varchar(200),
	capacity int,
	isManagementHeadOffice bool DEFAULT false,
	isManagementGeneral bool DEFAULT false,
	isSchoolPrimary bool DEFAULT false,
	isSchoolMiddle bool DEFAULT false,
	isSchoolHigh bool DEFAULT false,
	FOREIGN KEY (ministryID) REFERENCES Ministries(ministryID)
);

CREATE TABLE Persons (
	personID int PRIMARY KEY,
	firstName varchar(100),
	lastName varchar(100),
	dateOfBirth date,
	medicare varchar(14) NOT NULL UNIQUE , #example:  BOUF 1234 5678
	medicareExpiryDate date,
	phoneNumber varchar(14),    #(514) 262-2822
	address varchar(200),
	city varchar(100),
	province char(2),
	citizenship varchar(50),
	email varchar(200)
);

CREATE TABLE EmploymentRoles (
	employmentRoleID int PRIMARY KEY,
	title varchar(100),
	isHead bool DEFAULT false
);

CREATE TABLE Employees (
	personID int,
	facilityID int,
	startDate date,
	endDate date,
	primaryEmploymentRoleID int NOT NULL,
	secondaryEmploymentRoleID int,
```

```
	tertiaryEmploymentRoleID int,
	PRIMARY KEY (personID, facilityID, startDate),
	FOREIGN KEY (personID) REFERENCES Persons(personID),
	FOREIGN KEY (facilityID) REFERENCES Facilities(facilityID),
	FOREIGN KEY (primaryEmploymentRoleID) REFERENCES
EmploymentRoles(employmentRoleID),
	FOREIGN KEY (secondaryEmploymentRoleID) REFERENCES
EmploymentRoles(employmentRoleID),
	FOREIGN KEY (tertiaryEmploymentRoleID) REFERENCES
EmploymentRoles(employmentRoleID)
);

CREATE TABLE Students (
	personID int,
	facilityID int,
	startDate date,
	endDate date,
	grade varchar(50), #eg: secondary 2
	PRIMARY KEY (personID, facilityID, startDate),
	FOREIGN KEY (personID) REFERENCES Persons(personID),
	FOREIGN KEY (facilityID) REFERENCES Facilities(facilityID)
);



CREATE TABLE InfectionTypes(
	infectionTypeID int PRIMARY KEY,
	name varchar(100)
);

CREATE TABLE Infections (
	personID int,
	date date,
	infectionTypeID int,
	PRIMARY KEY (personID, date),
	FOREIGN KEY (infectionTypeID) REFERENCES InfectionTypes(infectionTypeID),
	FOREIGN KEY (personID) REFERENCES Persons(personID)
);

CREATE TABLE VaccinationTypes (
	vaccinationTypeID int PRIMARY KEY,
	name varchar(100)
);

CREATE TABLE Vaccines (
	personID int,
	date date,
	vaccinationTypeID int,
	dose int,
	PRIMARY KEY (personID, date),
	FOREIGN KEY (personID) REFERENCES Persons(personID),
	FOREIGN KEY (vaccinationTypeID) REFERENCES VaccinationTypes(vaccinationTypeID)
);
```

Part 3: SQL Statements To Express The Required Queries

Query 1

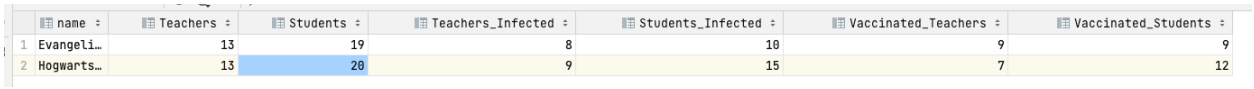
```
SELECT HeadOffices.name, HeadOffices.ministryID as HeadOfficeMinistryID,
HeadOffices.Province, HeadOffices.firstName, HeadOffices.lastName,
FacilitiesPerMinistry.count as Facilities, EmployeesPerMinistry.count as Employees,
StudentsPerMinistry.count as Students
FROM
(SELECT Ministries.ministryID, Ministries.name, Facilities.province as Province,
Persons.firstName, Persons.lastName
       FROM Facilities, Ministries, Persons, Employees, EmploymentRoles
       WHERE Facilities.ministryID = Ministries.ministryID
       AND Employees.facilityID = Facilities.facilityID
       AND Persons.personID = Employees.personID
       AND Employees.primaryEmploymentRoleID = EmploymentRoles.employmentRoleID
       AND Facilities.isManagementHeadOffice=true
       AND EmploymentRoles.isHead = true)
AS HeadOffices,
(SELECT Facilities.ministryID, COUNT(Facilities.facilityID) as count
       FROM Facilities
       GROUP BY Facilities.ministryID)
AS FacilitiesPerMinistry,
(SELECT Ministries.ministryID, COUNT(Employees.personID) as count
       FROM Ministries, Employees, Facilities
       WHERE Employees.facilityID = Facilities.facilityID
       AND Facilities.ministryID = Ministries.ministryID
       AND Employees.endDate IS NULL
       GROUP BY Ministries.ministryID)
AS EmployeesPerMinistry,
(SELECT Ministries.ministryID, COUNT(Students.personID) as count
       FROM Ministries, Students, Facilities
       WHERE Students.facilityID = Facilities.facilityID
       AND Facilities.ministryID = Ministries.ministryID
       AND Students.endDate IS NULL
       GROUP BY Ministries.ministryID)
AS StudentsPerMinistry
WHERE FacilitiesPerMinistry.ministryID = HeadOffices.ministryID
AND EmployeesPerMinistry.ministryID = HeadOffices.ministryID
AND StudentsPerMinistry.ministryID = HeadOffices.ministryID
GROUP BY HeadOffices.ministryID
ORDER BY Facilities DESC;
```

	name	HeadOfficeMinistryID	Province	firstName	lastName	Facilities	Employees	Students
1	Ministry of Education Quebec	1	QC	Bernanrd	Drainville	19	261	327
2	Ministry of Education Ontario	2	ON	Joe	Ontario	11	105	167
3	Ministry of Education Manitoba	5	MB	Stephanie	Miller	10	142	199
4	Ministry of Education Nova Scotia	7	NS	Jennifer	Clark	10	159	162
5	Ministry of Education PEI	8	PE	Emily	Campbell	10	163	179
6	Ministry of Education Saskatchewan	9	SK	Sophia	Murphy	10	147	185
7	Ministry of Education British Columbia	4	BC	Michael	Taylor	9	140	121

Query 2

```
SELECT Facilities.name, Teachers.count as Teachers, StudentsPerSchool.count as
Students, Covid19Teachers.infections as Teachers_Infected, Covid19Students.infections
As Students_Infected, VaccinatedTeachers.count as Vaccinated_Teachers,
VaccinatedStudents.count as Vaccinated_Students
FROM Facilities,
(SELECT Facilities.facilityID,count(Employees.personID) as count
       FROM Facilities,Employees
       WHERE
       Employees.facilityID = Facilities.facilityID
       AND (Employees.primaryEmploymentRoleID = 11 OR primaryEmploymentRoleID = 12)
       AND Employees.endDate IS NULL
       GROUP BY Facilities.facilityID)
as Teachers,
(SELECT Facilities.facilityID, count(Students.personID) as count
       FROM Facilities, Students
       WHERE
       Students.facilityID = Facilities.facilityID
       AND Students.endDate IS NULL
       GROUP BY Facilities.facilityID)
as StudentsPerSchool,
(SELECT DISTINCT Facilities.facilityID, COUNT(Employees.personID) as infections
       FROM Facilities, Employees, Infections
       WHERE Employees.facilityID = Facilities.facilityID
       AND Employees.endDate IS NULL
       AND (Employees.primaryEmploymentRoleID = 11 OR primaryEmploymentRoleID = 12)
       AND Infections.personID = Employees.personID
       GROUP BY Facilities.facilityID)
AS Covid19Teachers,
(SELECT DISTINCT Facilities.facilityID, COUNT(Students.personID) as infections
       FROM Facilities, Students, Infections
       WHERE Students.facilityID = Facilities.facilityID
       AND Students.endDate IS NULL
       AND Infections.personID = Students.personID
       GROUP BY Facilities.facilityID)
AS Covid19Students,
(SELECT Facilities.facilityID,COUNT(Employees.personID) as count
       FROM Facilities,Employees,Vaccines
       WHERE
       Employees.facilityID = Facilities.facilityID
       AND Vaccines.personID = Employees.personID
       AND Vaccines.dose = 1
       AND (Employees.primaryEmploymentRoleID = 11 OR primaryEmploymentRoleID = 12)
       AND Employees.endDate IS NULL
       GROUP BY Facilities.facilityID)
AS VaccinatedTeachers,
(SELECT Facilities.facilityID,COUNT(Students.personID) as count
       FROM Facilities,Students,Vaccines
       WHERE
       Students.facilityID = Facilities.facilityID
       AND Vaccines.personID = Students.personID
       AND Vaccines.dose = 1
       AND Students.endDate IS NULL
```

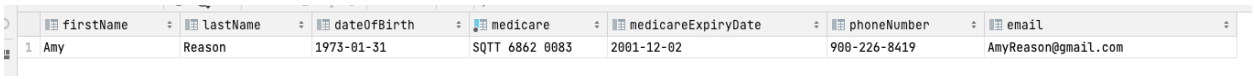
```
        GROUP BY Facilities.facilityID)
AS VaccinatedStudents
WHERE
        Facilities.facilityID = Teachers.facilityID
        AND Facilities.facilityID = StudentsPerSchool.facilityID
        AND Facilities.facilityID = Covid19Teachers.facilityID
        AND Facilities.facilityID = Covid19Students.facilityID
        AND Facilities.facilityID = VaccinatedTeachers.facilityID
        AND Facilities.facilityID = VaccinatedStudents.facilityID
        AND (Facilities.isSchoolPrimary = TRUE OR Facilities.isSchoolMiddle = TRUE OR
Facilities.isSchoolHigh = TRUE)
        AND Facilities.city = 'Montréal'
        GROUP BY Facilities.name
        ORDER BY Facilities.name;
```



	name	Teachers	Students	Teachers_Infected	Students_Infected	Vaccinated_Teachers	Vaccinated_Students
1	Evangeli…	13	19	8	10	9	9
2	Hogwarts…	13	20	9	15	7	12

Query 3

```
SELECT
       P.firstName,
       P.lastName,
       P.dateOfBirth,
       P.medicare,
       P.medicareExpiryDate,
       P.phoneNumber,
       P.email
FROM
       Persons P, Students S, Infections I
WHERE S.personID = P.personID
       AND I.personID = P.personID
       AND S.endDate IS NULL
       AND S.facilityID = 4
       AND P.medicareExpiryDate < CURDATE()
       AND I.infectionTypeID = 1;
```



	firstName	lastName	dateOfBirth	medicare	medicareExpiryDate	phoneNumber	email
1	Amy	Reason	1973-01-31	SQTT 6862 0083	2001-12-02	900-226-8419	AmyReason@gmail.com

Query 4

```
WITH RecentInfections AS (
       SELECT date, name as InfectionType, personID
       FROM Infections, InfectionTypes
       WHERE Infections.infectionTypeID = InfectionTypes.infectionTypeID
       AND date >= DATE_SUB(NOW(), INTERVAL 2 WEEK)
),
LavalStudents AS (
       SELECT Students.personID
       FROM Facilities, Students
       WHERE Facilities.facilityID = Students.facilityID
       AND Students.endDate IS NULL
       AND Facilities.city = 'Laval')
SELECT firstName, lastName, dateOfBirth, InfectionType, RecentInfections.date AS
InfectionDate, phoneNumber, email
FROM RecentInfections,LavalStudents, Persons
WHERE RecentInfections.personID = LavalStudents.personID
       AND LavalStudents.personID = Persons.personID
       AND LavalStudents.personID IN
       (SELECT personID FROM RecentInfections GROUP BY personID HAVING COUNT(*)>=2)
ORDER BY firstName, lastName, RecentInfections.date;
```

	firstName	lastName	dateOfBirth	InfectionType	InfectionDate	phoneNumber	email
1	Catherine	Desai	1998-02-12	COVID-19	2023-07-18	866-934-3065	CatherineDesai@gmail.com
2	Catherine	Desai	1998-02-12	SARS-Cov-2 Variant	2023-07-20	866-934-3065	CatherineDesai@gmail.com
3	Lucille	Quilliams	1962-07-12	COVID-19	2023-07-17	931-443-7850	LucilleQuilliams@gmail.com
4	Lucille	Quilliams	1962-07-12	SARS-Cov-2 Variant	2023-07-19	931-443-7850	LucilleQuilliams@gmail.com
5	Lucille	Quilliams	1962-07-12	Other	2023-07-21	931-443-7850	LucilleQuilliams@gmail.com

Query 5

```
SELECT
    M.name AS ministry_name,
    F.name AS facility_name,
    F.city AS facility_city,
    P.firstName,
    P.lastName,
    P.email
FROM
    Persons P
JOIN
    Employees E ON P.personID = E.personID AND E.endDate IS NULL
JOIN
    Facilities F ON E.facilityID = F.facilityID
JOIN
    Ministries M ON F.ministryID = M.ministryID
WHERE
    P.personID IN (
        SELECT
            I.personID
        FROM
            Infections I
        WHERE
```

```
            I.infectionTypeID = 1
    )
    AND P.personID NOT IN (
        SELECT
            V.personID
        FROM
            Vaccines V
    )
ORDER BY
    M.name,
    F.city,
    F.name;
```

	ministry_name	facility_name	facility_city	firstName	lastName	email
1	Ministry of Education Alberta	Banff Elementary School	Banff	Stephen	Fenstermaker	StephenFenstermaker@gmail.com
2	Ministry of Education Alberta	Banff Elementary School	Banff	Kevin	Han	KevinHan@gmail.com
3	Ministry of Education Alberta	Calgary Elementary School	Calgary	Cody	Caudel	CodyCaudel@gmail.com
4	Ministry of Education Alberta	Grande Prairie Elementary School	Grande Prairie	Ronald	Biondo	RonaldBiondo@gmail.com
5	Ministry of Education Alberta	Grande Prairie Elementary School	Grande Prairie	Stephen	Mixson	StephenMixson@gmail.com
6	Ministry of Education Alberta	Grande Prairie Elementary School	Grande Prairie	Lloyd	Bayardo	LloydBayardo@gmail.com
7	Ministry of Education Alberta	Lethbridge Middle School	Lethbridge	Ronald	Guajardo	RonaldGuajardo@gmail.com
8	Ministry of Education Alberta	Red Deer High School	Red Deer	Jacob	Eggleston	JacobEggleston@gmail.com
9	Ministry of Education Alberta	Red Deer High School	Red Deer	Perry	Cortez	PerryCortez@gmail.com
10	Ministry of Education British Columbia	Abbotsford Secondary School	Abbotsford	Goldie	Machak	GoldieMachak@gmail.com
11	Ministry of Education British Columbia	Kamloops Elementary School	Kamloops	Camille	Boozer	CamilleBoozer@gmail.com
12	Ministry of Education British Columbia	Nanaimo High School	Nanaimo	John	Johnson	JohnJohnson@gmail.com
13	Ministry of Education British Columbia	Nanaimo High School	Nanaimo	Anastasia	Valencia	AnastasiaValencia@gmail.com
14	Ministry of Education British Columbia	Nanaimo High School	Nanaimo	Audrey	Feliciano	AudreyFeliciano@gmail.com
15	Ministry of Education British Columbia	Prince George Middle School	Prince George	Hollis	Strait	HollisStrait@gmail.com
16	Ministry of Education British Columbia	Prince George Middle School	Prince George	Timothy	Tso	TimothyTso@gmail.com
17	Ministry of Education British Columbia	Vancouver Elementary School	Vancouver	Colton	Mcglinchey	ColtonMcglinchey@gmail.com
18	Ministry of Education British Columbia	Vancouver Secondary School	Vancouver	Grant	Whitt	GrantWhitt@gmail.com
19	Ministry of Education British Columbia	Victoria High School	Victoria	Robert	Brooks	RobertBrooks@gmail.com
20	Ministry of Education British Columbia	Victoria High School	Victoria	Erica	Yanni	EricaYanni@gmail.com
21	Ministry of Education British Columbia	Victoria High School	Victoria	Gerald	Carmack	GeraldCarmack@gmail.com
22	Ministry of Education British Columbia	Victoria High School	Victoria	Nancy	Gundlach	NancyGundlach@gmail.com

This has 103 results…

Query 6

```
SELECT Facilities.name AS FacilityName,
       Ministries.name AS MinistryName,
       Facilities.city,
       CONCAT(Persons.firstName, ' ', Persons.lastName) AS PrincipalName
FROM Facilities, Ministries, Persons, Employees
WHERE  Facilities.ministryID = Ministries.ministryID
    AND (Facilities.isSchoolPrimary = true
        OR  Facilities.isSchoolMiddle = true
        OR Facilities.isSchoolHigh = true)
    AND Facilities.facilityID = Employees.facilityID
    AND Employees.personID = Persons.personID
    AND Employees.primaryEmploymentRoleID = 10
    AND Facilities.facilityID NOT IN (
      SELECT DISTINCT E.facilityID
      FROM Employees E, Infections I
      WHERE E.personID = I.personID
  )
  AND Facilities.facilityID NOT IN (
      SELECT DISTINCT S.facilityID
      FROM Students S, Infections I
      WHERE S.personID = I.personID
  )
```

```
GROUP BY Facilities.facilityID, Facilities.name, Ministries.name, Facilities.city,
Persons.firstName, Persons.lastName
ORDER BY Ministries.name ASC, Facilities.city ASC, Facilities.name ASC;
```

	FacilityName	MinistryName	city	PrincipalName
1	Calgary High School	Ministry of Education Alberta	Calgary	Bessie Payne
2	Yorkton Elementary School	Ministry of Education Saskatchewan	Yorkton	Thomas Gadbois

Query 7

```
SELECT VaccinationTypes.name AS VaccineType, COUNT(Vaccines.personID) AS TotalDoses
FROM VaccinationTypes, Vaccines, Students
WHERE VaccinationTypes.vaccinationTypeID = Vaccines.vaccinationTypeID
    AND Vaccines.personID = Students.personID
GROUP BY VaccinationTypes.name
ORDER BY TotalDoses DESC;
```

	VaccineType	TotalDoses
1	Moderna	392
2	Johnson & Johnson	372
3	AstraZeneca	363
4	Pfizer	358

Query 8

```
SELECT InfectionTypes.name, Ministries.name, COUNT(DISTINCT Students.personID) AS
TotalInfections
FROM InfectionTypes, Ministries, Infections, Students, Facilities
WHERE InfectionTypes.infectionTypeID = Infections.infectionTypeID
    AND Infections.personID = Students.personID
    AND  Students.facilityID = Facilities.facilityID
    AND Facilities.ministryID = Ministries.ministryID
GROUP BY InfectionTypes.name, Ministries.name
ORDER BY InfectionTypes.name ASC, TotalInfections DESC;
```

	InfectionTypes.name	Ministries.name	TotalInfections
1	COVID-19	Ministry of Education Quebec	69
2	COVID-19	Ministry of Education Newfoundalnd Labrador	37
3	COVID-19	Ministry of Education Ontario	34
4	COVID-19	Ministry of Education PEI	34
5	COVID-19	Ministry of Education Manitoba	32
6	COVID-19	Ministry of Education Saskatchewan	32
7	COVID-19	Ministry of Education Nova Scotia	28
8	COVID-19	Ministry of Education British Columbia	23
9	COVID-19	Ministry of Education Alberta	19
10	Other	Ministry of Education Quebec	64
11	Other	Ministry of Education Manitoba	49
12	Other	Ministry of Education Newfoundalnd Labrador	43
13	Other	Ministry of Education PEI	29
14	Other	Ministry of Education Nova Scotia	27
15	Other	Ministry of Education Ontario	27
16	Other	Ministry of Education British Columbia	27
17	Other	Ministry of Education Alberta	22
18	Other	Ministry of Education Saskatchewan	22
19	SARS-Cov-2 Variant	Ministry of Education Quebec	62
20	SARS-Cov-2 Variant	Ministry of Education Saskatchewan	39
21	SARS-Cov-2 Variant	Ministry of Education Manitoba	38
22	SARS-Cov-2 Variant	Ministry of Education PEI	33
23	SARS-Cov-2 Variant	Ministry of Education Nova Scotia	32
24	SARS-Cov-2 Variant	Ministry of Education British Columbia	31
25	SARS-Cov-2 Variant	Ministry of Education Ontario	30
26	SARS-Cov-2 Variant	Ministry of Education Newfoundalnd Labrador	28
27	SARS-Cov-2 Variant	Ministry of Education Alberta	21

Part 4: Populate Tables of Database

Here are some **examples** of statements to populate the database tables. **The complete set of statements can be found in the code files attached**.

```
# Create Ministries and Facilities
INSERT INTO Ministries (ministryId, name) VALUES (1,'Ministry of Education Quebec');
INSERT INTO Facilities (facilityID, ministryID, name, address, city, province,
postalCode, phoneNumber, web, capacity, isManagementGeneral)
VALUES (1,1,'Management Montréal Branch','1200 Blvd.
Saint-Laurent','Montréal','QC','H197B2','514-222-2222','www.montreal.educanada.ca',200
0,true);
INSERT INTO Facilities (facilityID, ministryID, name, address, city, province,
postalCode, phoneNumber, web, capacity, isManagementHeadOffice)
VALUES (2,1,'Management Head Office','999
Ste-Catherine','Montréal','QC','H197B2','514-222-2222','www.montreal.educanada.ca',200
0,true);
INSERT INTO Facilities (facilityID, ministryID, name, address, city, province,
postalCode, phoneNumber, web, capacity, isSchoolMiddle, isSchoolHigh)
VALUES (3,1,'Laval Secondary School','1200 Blvd.
Saint-Laurent','Laval','QC','H197B2','514-222-2222','laval.montreal.educanada.ca',2000
,true,true);


# Vaccine Types
INSERT INTO VaccinationTypes VALUE (1,'Pfizer');
INSERT INTO VaccinationTypes VALUE (2,'Moderna');
INSERT INTO VaccinationTypes VALUE (3,'AstraZeneca');
INSERT INTO VaccinationTypes VALUE (4,'Johnson & Johnson');

# Infection Types
INSERT INTO InfectionTypes VALUE (1,'COVID-19');
INSERT INTO InfectionTypes VALUE (2,'SARS-Cov-2 Variant');
INSERT INTO InfectionTypes VALUE (3,'Other');


# Creating students
INSERT INTO Persons (personID, firstName, lastName, dateOfBirth, medicare,
medicareExpiryDate, phoneNumber, address, city, province, citizenship, email)
VALUES (3001, 'Marie', 'Hilyer', '1998-01-11', 'FVQH 3948 6741', '2024-02-22',
'506-156-1108', '2158 Mint Avenue', 'Sault Ste. Marie', 'NS', 'Canadian',
'MarieHilyer@gmail.com');
INSERT INTO Students (personID, facilityID, startDate, endDate, grade)
```

```
VALUES (3001, 3, '2020-11-30', NULL, 'Secondary 3');
INSERT INTO Persons (personID, firstName, lastName, dateOfBirth, medicare,
medicareExpiryDate, phoneNumber, address, city, province, citizenship, email)
VALUES (3002, 'Annie', 'Braswell', '1994-01-26', 'GFCL 3824 4259', '2025-11-12',
'935-047-5391', '5919 Lilac Street', 'London', 'NS', 'Canadian',
'AnnieBraswell@gmail.com');
INSERT INTO Students (personID, facilityID, startDate, endDate, grade)
VALUES (3002, 3, '2022-05-15', NULL, 'Secondary 5');
INSERT INTO Persons (personID, firstName, lastName, dateOfBirth, medicare,
medicareExpiryDate, phoneNumber, address, city, province, citizenship, email)
VALUES (3003, 'Brian', 'Bowman', '1970-10-28', 'GHOX 6379 6007', '2025-02-20',
'931-071-5912', '1978 Ivy Terrace', 'Chatham-Kent', 'NT', 'Canadian',
'BrianBowman@gmail.com');


# Creating teachers
INSERT INTO Persons (personID, firstName, lastName, dateOfBirth, medicare,
medicareExpiryDate, phoneNumber, address, city, province, citizenship, email)
VALUES (201, 'Shane', 'York', '2000-01-07', 'TWSX 3137 5093', '2027-02-10',
'936-817-8968', '1291 Hillcrest Avenue', 'Saskatoon', 'QC', 'Canadian',
'ShaneYork@gmail.com');
INSERT INTO Employees (personID, facilityID, startDate, endDate,
primaryEmploymentRoleID, secondaryEmploymentRoleID, tertiaryEmploymentRoleID)
VALUES (201, 3, '2021-01-01', NULL, 12, NULL, NULL);
INSERT INTO Persons (personID, firstName, lastName, dateOfBirth, medicare,
medicareExpiryDate, phoneNumber, address, city, province, citizenship, email)
VALUES (202, 'Lyn', 'Knight', '1957-10-31', 'DBGD 5859 1558', '2026-11-04',
'438-152-0030', '3973 Saffron Crescent', 'Sault Ste. Marie', 'NU', 'Canadian',
'LynKnight@gmail.com');
INSERT INTO Employees (personID, facilityID, startDate, endDate,
primaryEmploymentRoleID, secondaryEmploymentRoleID, tertiaryEmploymentRoleID)
VALUES (202, 3, '2021-01-01', NULL, 12, NULL, NULL);
INSERT INTO Persons (personID, firstName, lastName, dateOfBirth, medicare,
medicareExpiryDate, phoneNumber, address, city, province, citizenship, email)
VALUES (203, 'Rita', 'Kerr', '1964-08-24', 'ULYH 5022 3335', '2023-06-18',
'587-596-9724', '9512 Cypress Drive', 'Nanaimo', 'SK', 'Canadian',
'RitaKerr@gmail.com');



# Creating infections
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (5, '2020-11-24', 3);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (6, '2022-07-02', 3);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (15, '2021-02-18', 3);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (15, '2020-11-10', 1);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (16, '2021-12-29', 1);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (17, '2023-11-28', 1);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (110, '2023-11-09',
3);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (201, '2020-03-24',
2);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (202, '2021-06-10',
3);
INSERT INTO Infections (personID, date, infectionTypeID) VALUES (204, '2022-10-23',
1);
```

```
# Creating Vaccination
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (3,
'2021-07-14', 4, 1);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (14,
'2021-11-28', 2, 1);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (14,
'2022-04-06', 3, 2);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (15,
'2021-04-21', 2, 1);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (15,
'2022-07-23', 2, 2);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (17,
'2021-06-23', 1, 1);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (17,
'2022-08-23', 2, 2);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (19,
'2021-05-06', 2, 1);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (19,
'2022-04-14', 4, 2);
INSERT INTO Vaccines (personID, date, vaccinationTypeID, dose) VALUES (110,
'2021-05-16', 3, 1);
```

Part 5: Result of SELECT COUNT(*) FROM R

```
SELECT COUNT(*) FROM Employees;                # 1435 record
```

	`COUNT(*)`
1	1435

```
SELECT COUNT(*) FROM EmploymentRoles;          # 13 records
```

	`COUNT(*)`
1	13

```
SELECT COUNT(*) FROM Facilities;               # 98 records
```

	`COUNT(*)`
1	98

```
SELECT COUNT(*) FROM Infections;               # 1800 records
```

	`COUNT(*)`
1	1800

```
SELECT COUNT(*) FROM InfectionTypes;           # 3 records
```

	`COUNT(*)`
1	3

```
SELECT COUNT(*) FROM Ministries;               # 9 records
```

	`COUNT(*)`
1	9

```
SELECT COUNT(*) FROM Persons;                  # 3104 records
```

	`COUNT(*)`
1	3104

```
SELECT COUNT(*) FROM Students;                 # 1669 records
```

	`COUNT(*)`
1	1669

```
SELECT COUNT(*) FROM VaccinationTypes;         # 4 records
```

	`COUNT(*)`
1	4

```
SELECT COUNT(*) FROM Vaccines;                 # 2753 records
```

	`COUNT(*)`
1	2753