Reminders

- Quiz 4 is available
  - Due on Monday

- Homework assignment 1
  - Due in a week
  - Track progress on Moodle to-do list
  - Submit through Moodle.
Witnesses

- What is the price of the cheapest product?

```sql
SELECT min(price)
FROM Product
```

- What is the cheapest product?

```sql
SELECT pname
FROM Product
WHERE price = (SELECT min(price)
               FROM Product)
```
Subqueries

- A subquery is a SQL query nested inside a larger query
- Such inner-outer queries are called nested queries
- A subquery may occur in:
  - A SELECT clause
  - A FROM clause
  - A WHERE clause
- Rule of thumb: avoid writing nested queries when possible; keep in mind that sometimes it’s impossible
1. Subqueries in SELECT

Q: For each product return the country where it is manufactured

```
SELECT  pname, (SELECT  country
    FROM  Company
    WHERE  cname = manufacturer)
FROM  Product
```

What happens if the subquery returns more than one country?

Runtime error

Test it:
alter table company drop constraint company_pkey cascade;
insert into company values ('Canon', 'USA');
1. Subqueries in SELECT

Q: For each product return the country where it is manufactured

```
SELECT  pname, (SELECT  country
             FROM    Company
             WHERE   cname = manufacturer)
FROM     Product
```

"unnesting the query"

Whenever possible, don't use nested queries

```
SELECT  P.pname, C.country
FROM     Product P, Company C
WHERE    C.cname = P.manufacturer
```
2. Subqueries in FROM

Q: Find all products whose price is > 20 and < 50

```
SELECT X.pname
FROM (SELECT * FROM Product as P
       WHERE price > 20 ) as X
WHERE X.price < 50
```

unnesting

```
SELECT pname
FROM Product
WHERE price > 20 and price < 50
```
3. Subqueries in WHERE

Q: Find all companies that make some products with price < 100!

Using EXISTS:

```
SELECT DISTINCT cname
FROM Company
WHERE EXISTS (SELECT *
  FROM Product P
  WHERE cname = manufacturer
  and price < 100)
```
3. Subqueries in WHERE

Q: Find all companies that make some products with price < 100!

Using IN:

```
SELECT DISTINCT cname
FROM Company
WHERE cname IN (SELECT manufacturer
  FROM Product P
  WHERE price < 100)
```
3. Subqueries in WHERE

Q: Find all companies that make some products with price < 100!

Using ANY:

```
SELECT DISTINCT cname
FROM Company
WHERE 100 > ANY (SELECT price
                   FROM Product
                   WHERE manufacturer = cname)
```
3. Subqueries in WHERE

Q: Find all companies that make some products with price < 100!

Now, let's unnest:

```
SELECT DISTINCT cname
FROM Company, Product
WHERE cname = manufacturer
and price < 100
```

Existential quantifiers are easy 😊
3. Subqueries in WHERE

Q: Find all companies that make only products with price < 100!

same as:

Q: Find all companies for which all products have price < 100!

Universal quantifiers ∀

Universal quantifiers are more complicated 😞
3. Subqueries in WHERE

1. Find the other companies: i.e., they have some product $\geq 100$!

```
SELECT DISTINCT C.cname
FROM Company C
WHERE C.cname IN (SELECT P.manufacturer
                   FROM Product P
                   WHERE P.price >= 100)
```

2. Find all companies s.t. all their products have price $< 100$!

```
SELECT DISTINCT C.cname
FROM Company C
WHERE C.cname NOT IN (SELECT P.manufacturer
                        FROM Product P
                        WHERE P.price >= 100)
```
3. Subqueries in WHERE

Q: Find all companies that make some products with price < 100!

```
SELECT DISTINCT cname
FROM Company
WHERE EXISTS (SELECT * 
               FROM Product P 
               WHERE cname = manufacturer 
               AND price < 100)
```

Q: Find all companies that make only products with price < 100!

```
SELECT DISTINCT cname
FROM Company
WHERE NOT EXISTS (SELECT * 
                  FROM Product 
                  WHERE cname = manufacturer 
                  AND price >= 100)
```
3. Subqueries in WHERE

Q: Find all companies that make some products with price < 100!

```sql
SELECT DISTINCT cname
FROM Company
WHERE 100 > ANY (SELECT price
FROM Product
WHERE manufacturer = cname)
```

Q: Find all companies that make only products with price < 100!

```sql
SELECT DISTINCT cname
FROM Company
WHERE 100 > ALL (SELECT price
FROM Product
WHERE manufacturer = cname)
```
Practice

Q: Find the country that produces the cheapest product

```sql
SELECT country
FROM Company, Product pl
WHERE pl.manufacturer = cname
    AND price <= ALL (SELECT price
                        FROM Product)
```
Practice

**Hard query: Find the employee with the most projects**

*Hint 1: Find how many projects each employee is working on*

*Hint 2: Find the maximum number of projects that an employee is working on*

*Hint 3: Find the employee whose project count equals the number above*
Practice

Hard query: Find the employee with the most projects

Hint 1: Find how many projects each employee is working on

Hint 2: Find the maximum number of projects that an employee is working on

Hint 3: Find the employee whose project count equals the number above

```
select name from employees x1, projects y1
where x1.empID = y1.empID group by name
having count(*) = (select max(total)
                          from (select name, count(*) as total
                                from employees x, projects y
                                where x.empID = y.empID
                                group by name) z);
```
**INTERSECT and EXCEPT**

\[
\begin{align*}
\text{(SELECT R.A} & \text{ FROM R)} \\
\text{INTERSECT} & \\
\text{(SELECT S.A} & \text{ FROM S)} \\
\text{SELECT R.A} & \text{ FROM R} \\
\text{WHERE EXISTS} & \text{ (SELECT * FROM S WHERE R.A=S.A)} \\
\text{SELECT R.A} & \text{ FROM R} \\
\text{WHERE NOT EXISTS} & \text{ (SELECT * FROM S WHERE R.A=S.A)} \\
\end{align*}
\]

*Not in all DBMSs*
Unnesting Aggregates

Find the number of companies in each country

Note: no need for DISTINCT
(DISTINCT is the same as GROUP BY)
Unnesting Aggregates

Find the number of products made in each country

\[
\text{SELECT DISTINCT country,}
\text{(SELECT count(*) FROM Company X, Product Y WHERE X.country = Z.country AND X.cname = Y.manufacturer)}
\text{FROM Company Z}
\]

\[
\text{SELECT country, count(*) FROM Company, Product WHERE cname = manufacturer GROUP BY country}
\]

Are they equivalent?