

SQL Subqueries

Subqueries

- SQL queries take as input one or more tables and produce a table as result
- Decompose a complex query into simpler parts and then assemble them back together
- **Replace a table with a query (SELECT statement) in another query**

Scalar Subqueries

- Queries that return a single value (scalar) can be used in the WHERE clause for conditions
- select *

from PC

where price = **(select max(price) from PC)**

IN and NOT IN

- Check if a value is member in a set
- select maker
from Product
where type = 'pc' and
maker **IN (select maker
from Product
where type = 'laptop')**

EXISTS and NOT EXISTS

- Check if a query returns tuples or not (empty set)

- `select *`

`from PC`

`where not exists`

`(select *`

`from PC PC1`

`where PC1.price > PC.price`

`)`

LIMIT Clause

- Limit the number of tuples in the result
- select maker, ram from Product P, PC where P.model = PC.model and not exists (select ram from PC PC1 where PC1.ram > PC.ram)
- select maker, ram from Product P, PC where P.model = PC.model and not exists (select ram from PC PC1 where PC1.ram > PC.ram)

LIMIT 1

Correlated Subqueries

- Use attributes from an outer query inside a subquery
- select *

from **PC**

where not exists

(select *

from PC PC1

where PC1.price > **PC.price**

)

Subqueries in FROM

- Any query can be placed in FROM because it is a table
- select P.model, maker, **SQ.price**

FROM Product P,

(select model, price

from PC

where ram = (select max(ram) from PC)

) SQ

where P.model = **SQ.model**

Examples

- Computers
- TPC-H