

INF1343, Winter 2012

# Data Modeling and Database Design

Yuri Takhteyev

Faculty of Information  
University of Toronto



This presentation is licensed under Creative Commons Attribution License, v. 3.0. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/>. This presentation incorporates images from the Crystal Clear icon collection by Everaldo Coelho, available under LGPL from <http://everaldo.com/crystal/>.

**Week 9**

**Advanced Queries**

# “Loading” Data

## **Loading data from tab-delimited CSV Files**

the files can be prepared in  
OpenOffice Calc / Excel

Note: **tab**-delimited

### **Remote method:**

DB server accesses the files

### **Local method:**

Through the database client

# LOAD DATA INFILE

```
load data infile "<file_name>"  
into table <table_name>;
```

---

```
load data infile  
"/home/okenobi/names.csv" into  
table names;
```

The DB server must have access to the file.  
Your DB account must have file/super privileges.

# LOAD DATA LOCAL

```
load data local infile  
"<file_name>" into table  
<table_name>;
```

---

```
load data local infile  
"c:/Users/okenobi/Desktop/names.csv"  
into table names;
```

The file path in this case is to a file on the computer where your DB client is running.

Backslashes (\) in Windows files names must be flipped (/) or escaped (\\).

What are the names of the Diveshop's customers who paid cash for their orders?

```
select customer.name
from customer
  join vacation_order
    using(customer_id)
where
  payment_method = "Cash";
```

*How many* orders were paid in cash?

```
select count(*)  
from vacation_order  
where  
    payment_method = "Cash";
```

How much cash was paid altogether?

```
select sum(cost)
from vacation_order
where
    payment_method = "Cash";
```



And how much money was paid through Visa?

```
select sum(cost)
from vacation_order
where
    payment_method = "Visa";
```

And how much money was paid through Master Card?

```
select sum(cost)
from vacation_order
where
    payment_method = "Master Card";
```

Can we just get all the sums for each payment method at once?

```
select payment_method,  
       sum(cost)  
from vacation_order  
group by payment_method;
```

Which payment method brought in most money?

```
select payment_method,  
       sum(cost)  
from vacation_order  
group by payment_method  
order by sum(cost) desc;
```

Which payment method brought in most money for vacations that involved up to 2 people?

```
select payment_method,  
       sum(cost)  
from vacation_order  
where no_of_people<=2  
group by payment_method  
order by sum(cost) desc limit 1;
```

Note: "where" before "group by"!

Which payment method was used most often for for vacations that involved up to 2 people?

```
select payment_method,  
       count (*)  
from vacation_order  
where no_of_people<=2  
group by payment_method  
order by count (*) desc limit 1;
```

Which payment method was used most often for vacations that involved up to 2 people and cost over \$5000?

```
select payment_method,  
       count (*)  
from vacation_order  
where no_of_people<=2  
      and cost>5000  
group by payment_method  
order by count (*) desc limit 1;
```

Which payment methods brought in more than \$15,000 in *total*?

```
select payment_method,  
       sum(cost)  
from vacation_order  
group by payment_method  
having sum(cost) > 15000  
order by count(*) desc limit 1;
```

Note the order!



# Where vs Having

## **where**

selects rows from the original table (after all the joins)

## **having by**

selects rows from the aggregated table

# Order of Clauses

- 7** **select ...**
- 1** **from ...**
  - join ... (several times)**
- 2** **where ...**
- 3** **group by ...**
- 4** **having ...**
- 5** **order by ...**
- 6** **limit ...**

Which payment methods brought in more than \$15,000 on *average* per order?

```
select payment_method,  
       avg(cost)  
from vacation_order  
group by payment_method  
having avg(cost) > 15000;
```

What was the average amount coming from each payment methods for orders costing over \$15,000?

```
select avg(cost)
from vacation_order
where cost>15000
group by payment_method;
```

What about this query?

```
select avg(cost)
from vacation_order
where avg(cost) > 15000;
```

Invalid!

And this one?

```
select payment_method,  
       avg(cost)  
from vacation_order  
group by payment_method  
having cost > 15000;
```

Invalid!

What payment methods were used for vacations costing *above* average?

```
select avg(cost)
from vacation_order;
```

```
select payment_method
from vacation_order
where cost > 18462.8
group by payment_method;
```

What payment methods were used for vacations costing *above* average?

```
set @avg_cost = (  
    select avg(cost)  
    from vacation_order  
)  
;  
select payment_method  
from vacation_order  
where cost > @avg_cost  
group by payment_method;
```



What payment methods were used for vacations costing *above* average?

```
select payment_method
from vacation_order
where cost > (
    select avg(cost)
    from vacation_order
)
group by payment_method;
```

An “uncorrelated” subquery

What payment methods on average brought more money than the average for all vacations?

```
select payment_method,  
       avg(cost)  
from vacation_order  
group by payment_method  
having avg(cost) > (  
    select avg(cost)  
    from vacation_order  
);
```

Still “uncorrelated”

Which vacations cost more than the average for their payment method?

```
select order_id
from vacation_order as o
where cost > (
    select avg(cost)
    from vacation_order
    where payment_method
        =o.payment_method
);
```

This is a “correlated” query!

Which categories of marine life have more than one species? (Use species.)

```
select
  category, count (*)
from species
group by category
having count (*) > 1;
```

**We did not go through this and subsequent slides in class. Please review them on your own.**

What are the smallest and the largest lengths in each of the categories that have more than 1 species?

```
select category, min(length_cm) ,  
               max(length_cm)  
from species  
group by category  
having count(*) > 1;
```

Which category with  $>1$  species has the largest ratio between the largest and the smallest length of species?

```
select category,  
       max(length_cm) / min(length_cm)  
from species  
group by category  
having count(*) > 1;
```

Step 1

Which category with  $>1$  species has the largest ratio between the largest and the smallest length of species?

```
select category,  
       max(length_cm) / min(length_cm)  
       as ratio  
from species  
group by category  
having count(*) > 1;
```

Step 2

Which category with  $>1$  species has the largest ratio between the largest and the smallest length of species?

```
select * from (  
  select category,  
         max(length_cm) / min(length_cm)  
         as ratio  
  from species  
  group by category  
  having count(*) > 1) as categories  
 order by categories.ratio desc;
```

Step 3



Questions?