

ADVANCED

# SQL FOR ANALYTICS & BUSINESS INTELLIGENCE

Showing rows 0 - 24 (599 total, Query took 0.0212 seconds.)

```
1  SELECT customer.first_name,
2      customer.last_name,
3      COUNT(DISTINCT rental_id) AS rentals
4  FROM rental
5  LEFT JOIN customer
6      ON customer.customer_id = rental.customer_id
7  GROUP BY
8      customer.first_name,
9      customer.last_name
10 ORDER BY
11    rentals DESC
```

★★★★★ With Expert SQL Instructor John Pauler



# COURSE STRUCTURE



This is a **project-based course**, for students looking for a *practical, hands-on, and highly engaging* approach to querying and analyzing databases with MySQL

*Additional resources include:*

- ★ **Downloadable Ebook** to serve as a helpful reference when you're offline or on the go
- ★ **Quizzes & Homework Exercises** to test and reinforce key concepts, with step-by-step solutions
- ★ **Bonus Projects** to test your abilities and apply the skills developed throughout the course

# COURSE OUTLINE

1	<b>Introduction &amp; Setup</b>	<i>Discuss eCommerce databases, download Community Server and Workbench, and create the project database</i>
2	<b>Traffic Analysis &amp; Optimization</b>	<i>Identify top traffic sources, measure their conversion rates, analyze trends, and use segmentation for bidding optimization</i>
3	<b>Website Measurement &amp; Testing</b>	<i>Find the most-visited pages and top entry pages, calculate bounce rates, build conversion funnels, and analyze tests</i>

## [MID-COURSE PROJECT]

4	<b>Channel Analysis &amp; Optimization</b>	<i>Compare marketing channels, understand relative performance, optimize a channel portfolio, and analyze trends</i>
5	<b>Product-Level Analysis</b>	<i>Analyze sales, build product-level conversion funnels, learn about cross-selling, and measure the impact of launching new products</i>
6	<b>User-Level Analysis</b>	<i>Learn about behaviors of repeat visitors and purchasers, and compare new and repeat visitor website conversion patterns</i>

## [FINAL PROJECT]

# INTRODUCING THE COURSE PROJECT

## THE SITUATION

You've just been hired as an **eCommerce Database Analyst** for **Maven Fuzzy Factory**, an online retailer which has just launched their first product.

## THE BRIEF

As a member of the startup team, you will work with the *CEO*, the *Head of Marketing*, and the *Website Manager* to help steer the business.

You will analyze and optimize marketing channels, measure and test website conversion performance, and use data to understand the impact of new product launches.

## THE OBJECTIVE

### Use SQL to:

- *Access and explore the Maven Fuzzy Factory database*
- *Become the data expert for the company, and the go-to person for mission critical analyses*
- *Analyze and optimize the business' marketing channels, website, and product portfolio*

# SETTING EXPECTATIONS

## 1 You'll be learning MySQL, and practicing that using MySQL Workbench

- *In your career, you may end up using other “flavors” of SQL (T-SQL, PL/SQL, PostgreSQL, etc.)*
- *Each flavor is very similar, with only minor syntax changes; the concepts you learn will apply universally*

## 2 This course is meant for people who already have basic SQL skills

- *This course skips over the SQL basics, and jumps straight into a simulated real-world experience where students can practice advanced analytical skills, and learn to extract insights from data*

## 3 We will focus on extracting and analyzing data to generate insights

- *We will be using a custom-built and extremely rich eCommerce database to help steer the decisions of a business by analyzing marketing channels, measuring website performance, and exploring the product portfolio*

## 4 We will NOT cover building & maintenance of databases in this course

- *In the process of analyzing data, we will get into some very light table creation and altering of tables*
- *We will NOT cover database creation and performance optimization, managing user permissions, or other “DBA” skills*

# PREREQUISITE SKILLS REVIEW



# PREREQUISITE SKILLS

---

-  You should be very familiar with **the “Big 6” statements and clauses**, including **SELECT, FROM, WHERE, GROUP BY, HAVING, and ORDER BY**
-  You should know how to use **GROUP BY with Aggregate Functions**, including **COUNT, COUNT(DISTINCT), MIN, MAX, AVG, and SUM**
-  You should know how to **query multiple tables with INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL OUTER JOIN, and UNION**

# THE “BIG 6” ELEMENTS OF A SQL SELECT STATEMENT

START OF  
STATEMENT

**SELECT**

*Identifies the column(s) you want your query to select for your results*

**FROM**

*Identifies the table(s) your query will pull data from*

**WHERE**

*(Optional) Specifies record-filtering criteria for filtering your results*

**GROUP BY**

*(Optional) Specifies how to group the data in your results*

**HAVING**

*(Optional) Specifies group-filtering criteria for filtering your results*

**ORDER BY**

*(Optional) Specifies the order in which your query results are displayed*

**SELECT columnName**

**FROM tableName**

**WHERE logicalCondition**

**GROUP BY columnName**

**HAVING logicalCondition**

**ORDER BY columnName**

END OF  
STATEMENT

# AGGREGATE FUNCTIONS USED WITH GROUP BY

**COUNT**

Count of Records  
*Skips NULL, except COUNT(\*)*

**COUNT(columnName)**

**COUNT DISTINCT**

Count of Distinct Values  
*Skips NULL values*

**COUNT(DISTINCT columnName)**

**MIN**

Finds the Smallest Value  
*Skips NULL values*

**MIN(columnName)**

**MAX**

Finds the Largest Value  
*Skips NULL values*

**MAX(columnName)**

**AVG**

Average of All Values  
*Skips NULL values*

**AVG(columnName)**

**SUM**

SUM of All Values  
*Treats NULL values as Zero*

**SUM(columnName)**

# COMMON JOIN TYPES

## INNER JOIN

*Returns records that exist in **BOTH** tables, and excludes unmatched records from either table*

**FROM** leftTableName  
**INNER JOIN** rightTableName

## LEFT JOIN

*Returns ALL records from the **LEFT** table, and any matching records from the **RIGHT** table*

**FROM** leftTableName  
**LEFT JOIN** rightTableName

## RIGHT JOIN

*Returns ALL records from the **RIGHT** table, and any matching records from the **LEFT** table*

**FROM** leftTableName  
**RIGHT JOIN** rightTableName

## FULL OUTER JOIN

*Returns ALL records from **BOTH** tables, including non-matching records*

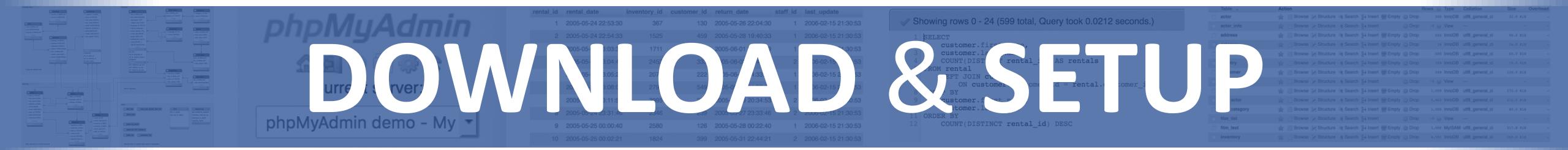
**FROM** leftTableName  
**FULL JOIN** rightTableName

## UNION

*Returns all data from one table, with all data from another table **appended to the end***

**SELECT FROM** firstTableName  
**UNION**  
**SELECT FROM** secondTableName

# DOWNLOAD & SETUP



# MySQL DOWNLOAD & SETUP – OVERVIEW

Step 1

## Download Community Server

*This allows SQL to run on your machine*

Step 2

## Download MySQL Workbench

*This is the program you'll use to write and run SQL queries (it's intuitive, and works across operating systems)*

Step 3

## Connect Workbench to Server

*We'll get you connected to the server so you can use Workbench to start running your own SQL queries*

Step 4

## Review Workbench Interface

*We'll take a quick tour of the Workbench interface to get you familiar with the layout and key components*

Step 5

## Create the Database

*We'll run the SQL code to build the database which we'll be exploring throughout the course (this part is easy!)*



### HEY THIS IS IMPORTANT!

If you took one of my other courses, and have already installed Community Server and MySQL Workbench, then you can skip ahead to creating the database. No need to re-install. Whatever version you have is great.

# STEP 1: COMMUNITY SERVER (MAC)



## HEY THIS IS IMPORTANT!

If you took one of my other courses, and have already installed Community Server and MySQL Workbench, then you can skip ahead to creating the database. No need to re-install. Whatever version you have is great.

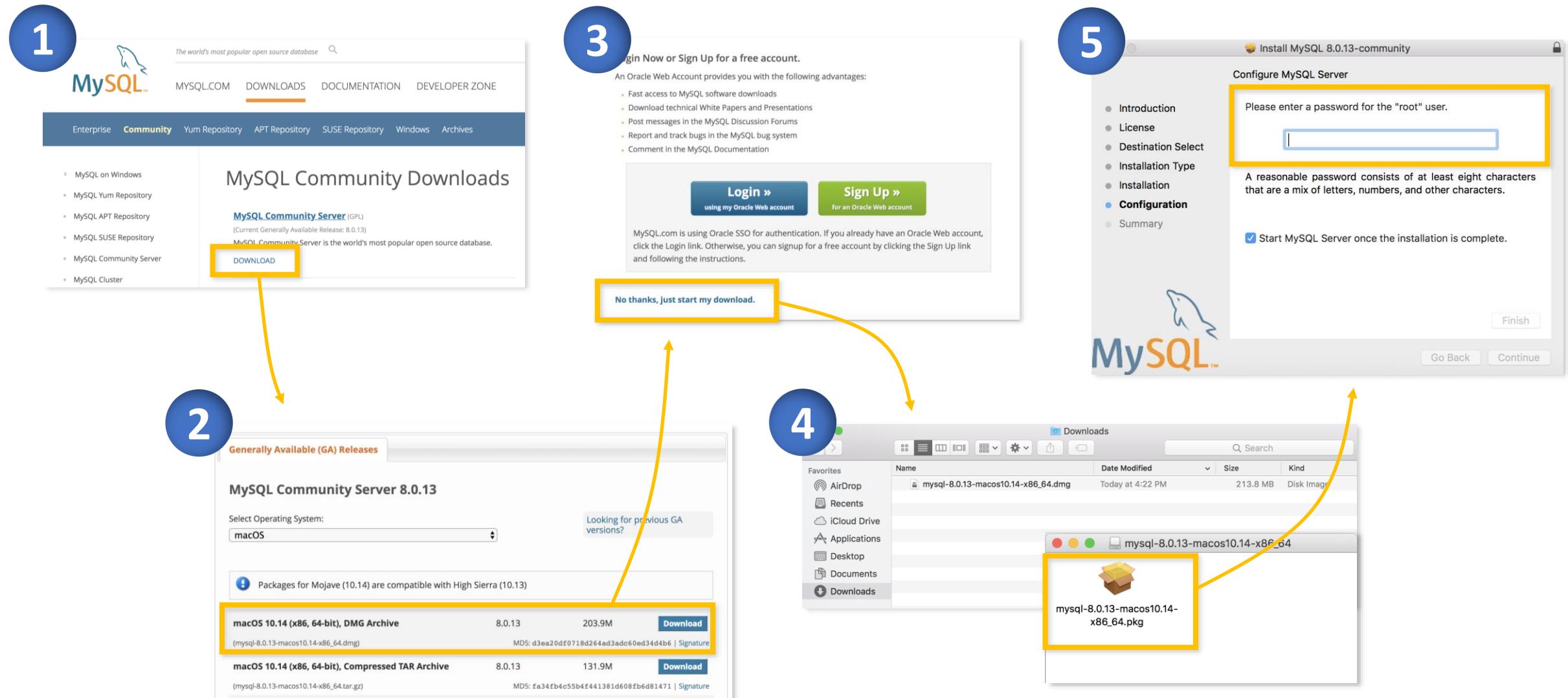
The screenshot shows the phpMyAdmin interface on a Mac OS X desktop. In the top left, there's a large 'phpMyAdmin' logo. Below it, a sidebar lists various databases and tables. The main area has four panes: 1) A 'Structure' diagram for a table. 2) A table data grid with columns: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update. It contains 10 rows of data. 3) A query results grid showing the results of a query: SELECT customer.first\_name, customer.last\_name, COUNT(DISTINCT rental\_id) AS rentals FROM rental JOIN customer ON rental.customer\_id = customer.customer\_id GROUP BY customer.first\_name, customer.last\_name ORDER BY COUNT(DISTINCT rental\_id) DESC. 4) A table list pane on the right showing all tables in the database, their types (e.g., MyISAM, InnoDB), and sizes.

# MySQL COMMUNITY SERVER – MAC DOWNLOAD GUIDE



- 1 Go to <https://dev.mysql.com/downloads> and download **MySQL Community Server**
- 2 Select the **MacOS** operating system, and download the **DMG Archive** version
  - *Note: you'll likely see a later version than the one shown (just download the latest)*
- 3 No need to Login or Sign Up, just click "**No thanks, just start my download**"
- 4 Find the install file in your downloads, then double click to run the installer package
- 5 Click through each install step, leaving defaults unless you need customized settings
  - *Note: Make sure to store your **root password** somewhere, you'll need this later!*

# MySQL COMMUNITY SERVER – MAC DOWNLOAD GUIDE



# STEP 1: COMMUNITY SERVER (PC)



## HEY THIS IS IMPORTANT!

If you took one of my other courses, and have already installed Community Server and MySQL Workbench, then you can skip ahead to creating the database. No need to re-install. Whatever version you have is great.

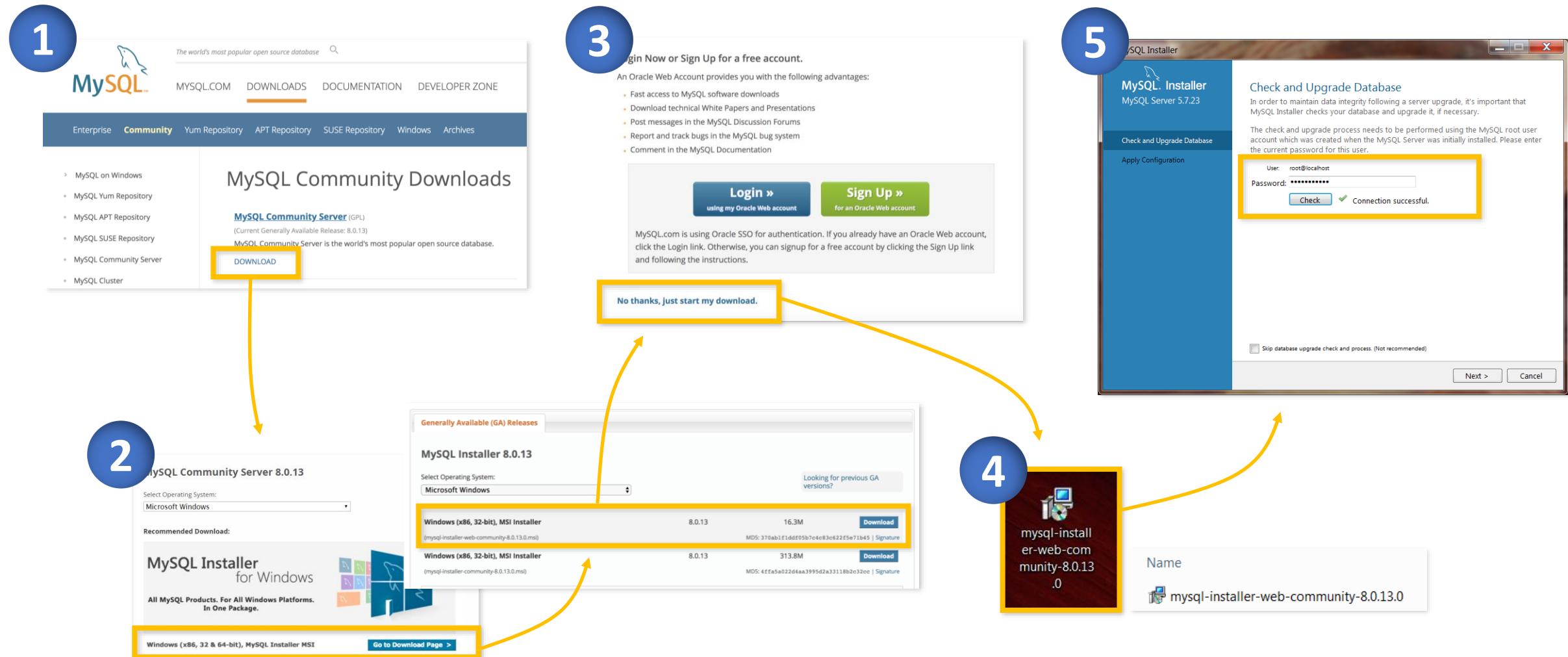
The screenshot shows the phpMyAdmin interface with a blue header. On the left, there's a tree view of the database schema. The main area has four panes: 1) A table structure for 'rental' with columns: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update. Data rows are listed from 1 to 10. 2) A results grid for a query showing columns: customer.first\_name, customer.last\_name, COUNT(DISTINCT rental\_id), and rental.last\_update. Rows 0-24 are shown, with 599 total. 3) A code editor with a SQL query for finding customers with the most rentals. 4) A table list on the right with various tables like actor, address, category, film, film\_actor, film\_text, and inventory, each with their respective details.

# MySQL COMMUNITY SERVER – PC DOWNLOAD GUIDE



- 1 Go to <https://dev.mysql.com/downloads> and download **MySQL Community Server**
- 2 Select the **Microsoft Windows** operating system, and the **Installer MSI** download
  - *Note: On the download page you may see two versions: select mysql-installer-web-community if you are connected to the internet, and keep in mind that you may see a later version than the one shown (just download the latest)*
- 3 No need to Login or Sign Up, just click “**No thanks, just start my download**”
- 4 Find the install file in your downloads, then double click to run the installer package
- 5 Click through each install step, leaving defaults unless you need customized settings
  - *Note: Make sure to store your **root password** somewhere, you'll need this later!*

# MySQL COMMUNITY SERVER – PC DOWNLOAD GUIDE



# STEP 2: MySQL WORKBENCH (MAC)



## HEY THIS IS IMPORTANT!

If you took one of my other courses, and have already installed Community Server and MySQL Workbench, then you can skip ahead to creating the database. No need to re-install. Whatever version you have is great.

The screenshot shows the MySQL Workbench interface. On the left, there's a tree view of a database schema with tables like actor, address, category, customer, film, film\_actor, film\_category, inventory, and rental. In the center, a results grid displays data from the rental table. To the right, a query editor shows a SELECT statement for finding the most frequent customers. Below the query editor is a table list.

Table	Action	Rows	Type	Collation	Size	Overflow
actor	Browse  > Structure  > Search  > Insert  > Drop	300	innodb	utf8_general_ci	33.0 KB	-
actor_info	Browse  > Structure  > Search  > Insert  > Drop	300	innodb	utf8_general_ci	16.0 KB	-
address	Browse  > Structure  > Search  > Insert  > Drop	400	innodb	utf8_general_ci	44.0 KB	-
category	Browse  > Structure  > Search  > Insert  > Drop	15	innodb	utf8_general_ci	16.0 KB	-
customer	Browse  > Structure  > Search  > Insert  > Drop	100	innodb	utf8_general_ci	130.0 KB	-
film	Browse  > Structure  > Search  > Insert  > Drop	500	innodb	utf8_general_ci	171.0 KB	-
film_actor	Browse  > Structure  > Search  > Insert  > Drop	300	innodb	utf8_general_ci	37.0 KB	-
film_category	Browse  > Structure  > Search  > Insert  > Drop	150	innodb	utf8_general_ci	40.0 KB	-
inventory	Browse  > Structure  > Search  > Insert  > Drop	1000	innodb	utf8_general_ci	447.0 KB	-
language	Browse  > Structure  > Search  > Insert  > Drop	5	innodb	utf8_general_ci	16.0 KB	-
rental	Browse  > Structure  > Search  > Insert  > Drop	2000	innodb	utf8_general_ci	100.0 KB	-
staff	Browse  > Structure  > Search  > Insert  > Drop	10	innodb	utf8_general_ci	16.0 KB	-
store	Browse  > Structure  > Search  > Insert  > Drop	1	innodb	utf8_general_ci	16.0 KB	-



# MySQL WORKBENCH – MAC DOWNLOAD GUIDE

- 1 Go to <https://dev.mysql.com/downloads/workbench>, scroll down to **Generally Available (GA) Releases**, and select the **MacOS** operating system
- 2 We'll be using version **8.0.16** for this course, so you can either click "***Looking for previous GA versions?***" to search for the same one, or simply download the latest available
- 3 No need to Login or Sign Up, just click "***No thanks, just start my download***"
- 4 Find the install file in your downloads, click the MySQL Workbench logo (*with the dolphin*) and drag it into your **Applications** folder
- 5 Look for MySQL workbench in your list of applications, double click to launch, then proceed to ***Step 3: Connecting to the server***

# MySQL WORKBENCH – MAC DOWNLOAD GUIDE



1

The screenshot shows the MySQL Workbench 8.0.16 download page. A yellow box highlights the 'Select Operating System' dropdown menu where 'macOS' is chosen. Another yellow box highlights the 'Looking for previous GA versions?' link.

3

The screenshot shows the MySQL.com login or sign-up page. It includes a 'Login Now or Sign Up for a free account.' section, a list of advantages for Oracle Web Account users, and 'Login >' and 'Sign Up >' buttons. A yellow box highlights the 'No thanks, just start my download.' button.

5

The screenshot shows the Mac OS X Applications folder. A yellow arrow points from the 'MySQLWorkbench' icon in the list to the Applications folder icon in the sidebar.

2

*Look for version **8.0.16**, or download the latest*

The screenshot shows the MySQL Workbench 6.3.10 download page. A yellow box highlights the 'Select Version' dropdown menu where '6.3.10' is chosen. Another yellow box highlights the 'Looking for the latest GA version?' link. A yellow box also highlights the 'Download' button.

4

The screenshot shows the MySQL Workbench 8.0 splash screen. It features the MySQL logo, the text 'MySQL Workbench 8.0', and instructions: 'Drag the MySQL Workbench icon to the Applications folder.' Two icons are shown: 'MySQLWorkbench' (with a blue dolphin icon) and 'Applications' (with a folder icon). Yellow boxes highlight both icons.

# STEP 2: MySQL WORKBENCH (PC)



## HEY THIS IS IMPORTANT!

If you took one of my other courses, and have already installed Community Server and MySQL Workbench, then you can skip ahead to creating the database. No need to re-install. Whatever version you have is great.

The screenshot shows the MySQL Workbench interface. On the left, there's a tree view of the database schema. In the center, there's a table viewer showing data from the 'rental' table. Below it, a query editor displays a SELECT statement. On the right, there's a results viewer showing the output of the query.

rental_id	rental_date	inventory_id	customer_id	return_date	staff_id	last_update
1	2005-05-24 22:53:30	367	130	2005-05-26 22:04:30	1	2006-02-15 21:30:53
2	2005-05-24 22:54:33	1525	459	2005-05-28 19:40:33	1	2006-02-15 21:30:53
3	2005-05-24 22:03:39	1711	408	2005-06-01 22:12:39	1	2006-02-15 21:30:53
4	2005-05-24 22:03:44	2455	333	2005-06-01 14:34:44	2	2006-02-15 21:30:53
5	2005-05-24 22:05:21	2079	222	2005-06-02 03:23:33	1	2006-02-15 21:30:53
6	2005-05-24 22:29:53	2792	549	2005-05-27 20:00:53	2	2006-02-15 21:30:53
7	2005-05-24 23:21:53	2079	222	2005-05-29 20:00:53	1	2006-02-15 21:30:53
8	2005-05-24 23:31:46	2346	239	2005-05-27 23:33:46	2	2006-02-15 21:30:53
9	2005-05-25 00:00:40	2580	128	2005-05-28 00:22:40	1	2006-02-15 21:30:53
10	2005-05-25 00:02:21	1824	399	2005-05-31 22:44:21	2	2006-02-15 21:30:53

```
✓ Showing rows 0 - 24 (599 total, Query took 0.0212 seconds.)  
1. SELECT  
2.     customer.first_name,  
3.     customer.last_name,  
4.     COUNT(DISTINCT rental_id) AS rental_count  
5. FROM  
6.     rental  
7.     JOIN  
8.     customer  
9.     ON customer.customer_id = rental.staff_id  
10.    GROUP BY  
11.        customer.first_name  
12.        customer.last_name  
13. ORDER BY  
14.        COUNT(DISTINCT rental_id) DESC
```

Table	Action	Rows	Type	Collation	Size	Overhead
actor	Browse	100	innodb	utf8_general_ci	33.0 K	-
actor_info	Browse	100	innodb	utf8_general_ci	16.0 K	-
address	Browse	100	innodb	utf8_general_ci	44.0 K	-
category	Browse	100	innodb	utf8_general_ci	16.0 K	-
city	Browse	100	innodb	utf8_general_ci	16.0 K	-
customer	Browse	100	innodb	utf8_general_ci	130.0 K	-
customer_address	Browse	100	innodb	utf8_general_ci	16.0 K	-
customer_info	Browse	100	innodb	utf8_general_ci	170.0 K	-
film	Browse	100	innodb	utf8_general_ci	170.0 K	-
film_actor	Browse	100	innodb	utf8_general_ci	170.0 K	-
film_category	Browse	100	innodb	utf8_general_ci	40.0 K	-
film_text	Browse	100	innodb	utf8_general_ci	310.0 K	-
inventory	Browse	100	innodb	utf8_general_ci	40.0 K	-

# MySQL WORKBENCH – PC DOWNLOAD GUIDE

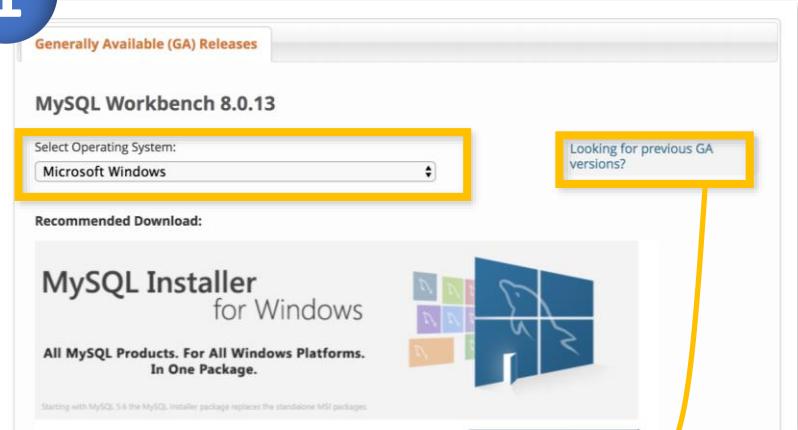


- 1 Go to <https://dev.mysql.com/downloads/workbench>, scroll down to **Generally Available (GA) Releases**, and select the **Microsoft Windows** operating system
- 2 We'll be using version **8.0.13** for this course, so you can either click "***Looking for previous GA versions?***" to search for the same one, or simply download the latest available
- 3 No need to Login or Sign Up, just click "***No thanks, just start my download***"
- 4 Find the install file in your downloads, double click to run the installation process, and stick with default settings unless you need a custom configuration
- 5 Look for MySQL workbench in your list of programs, double click to launch, then proceed to ***Step 3: Connecting to the server***
  - ***Note:*** You may see a warning if you aren't on **Windows 10+**, but most older systems (i.e. Windows 7) should be compatible

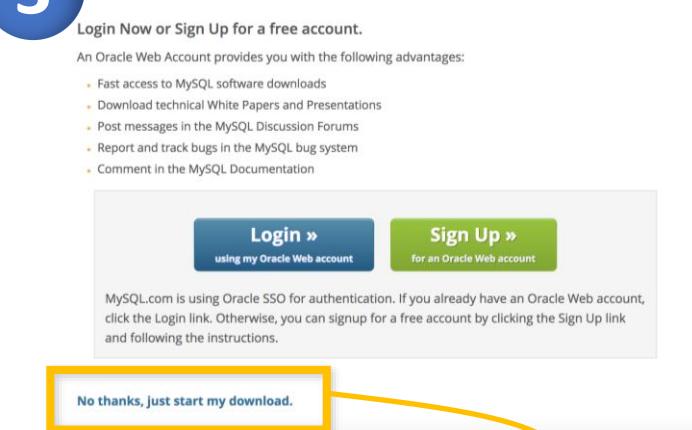


# MySQL WORKBENCH – PC DOWNLOAD GUIDE

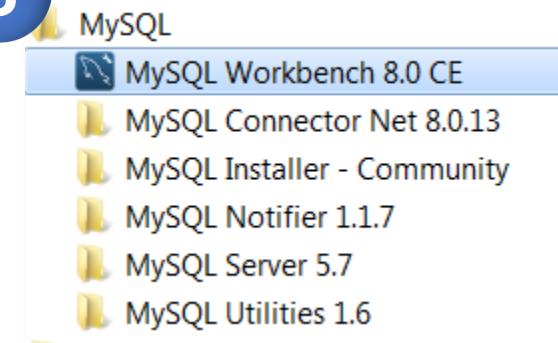
1



3

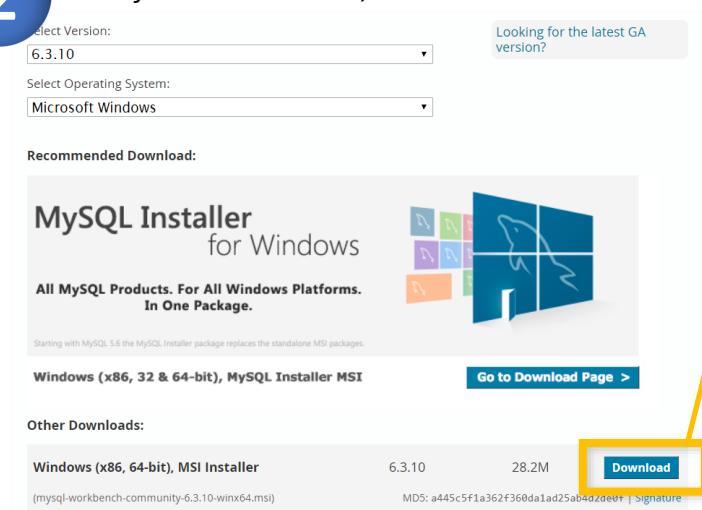


5

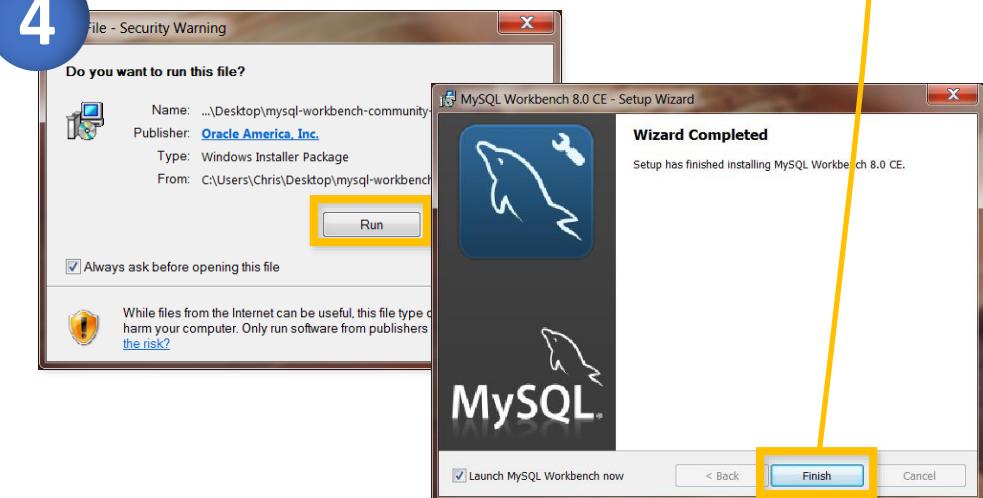


2

*Look for version **8.0.13**, or download the latest*



4



# STEP 3: CONNECTING TO THE SERVER

The image shows a screenshot of the phpMyAdmin interface. On the left, there is a navigation tree for a database named 'mydb'. The main area displays two tables: 'rental' and 'customer'. The 'rental' table has columns: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update. The 'customer' table has columns: customer\_id, first\_name, last\_name, address\_id, email, phone, and password. Below these tables is a query editor window with the following SQL code:

```
SELECT
    customer.first_name,
    customer.last_name,
    COUNT(DISTINCT rental_id) AS rentals
FROM
    rental
LEFT JOIN customer
    ON rental.customer_id = customer.customer_id
GROUP BY
    customer.first_name,
    customer.last_name
ORDER BY
    COUNT(DISTINCT rental_id) DESC
```

The results of the query are shown in a table below the editor, displaying 24 rows of data from the 'rental' table.

# CONNECTING TO THE SERVER

- 1 After launching Workbench, check the **MySQL Connections** section on the welcome page
  - *If you see a connection already, right-click to **Edit Connection**, otherwise click the **plus sign (+)** to add a new one*
- 2 Name the connection “**mavenmovies**”, confirm that the Username is “**root**”, and click **OK**
- 3 Once you see the **mavenmovies** connection on your welcome screen, simply click the tile and enter your **root password** to complete the connection



## Fun Fact!

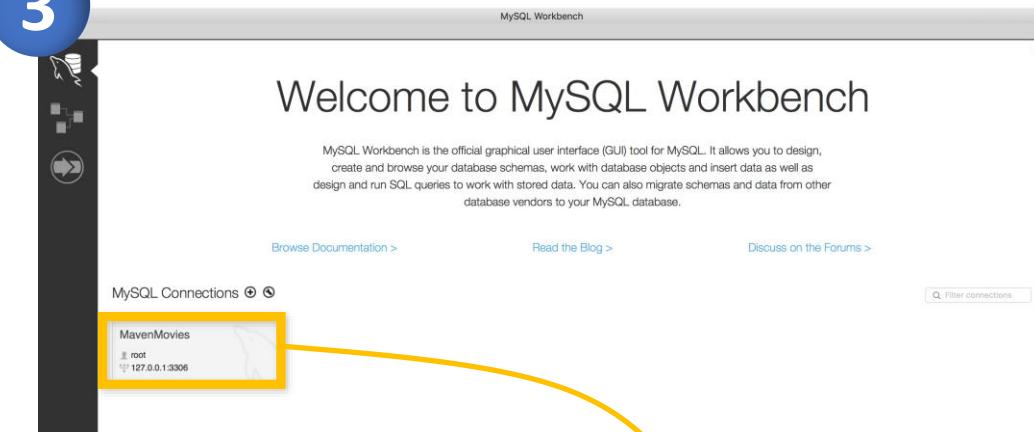
Maven Movies is the name of the database I used when I made my first course. I always name my connections ‘mavenmovies’ as tribute. **It does not matter what you name your connection. Name it anything you want!**

# CONNECTING TO THE SERVER

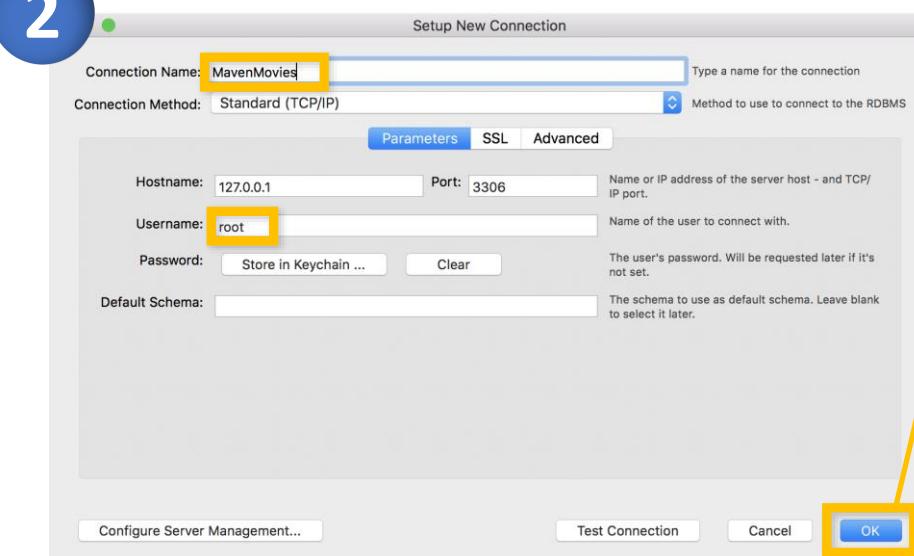
1



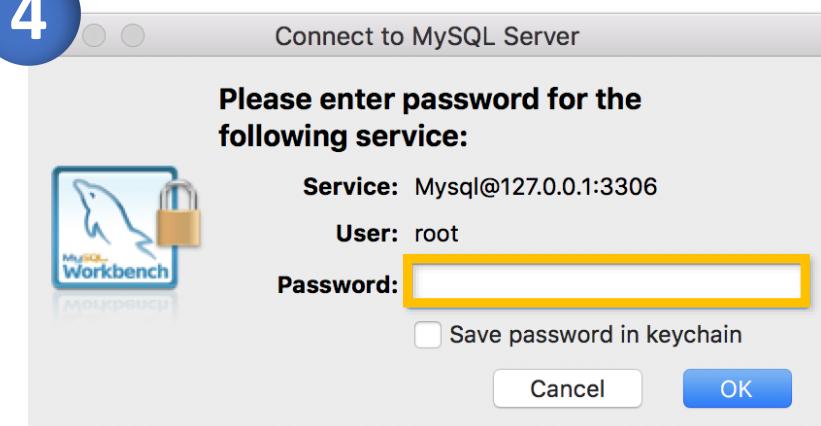
3



2



4



# STEP 4: MySQL WORKBENCH INTERFACE

The image shows a composite screenshot of a MySQL environment. On the left, there's a large, semi-transparent watermark of the MySQL Workbench interface, which includes a database schema diagram at the top, followed by tabs for 'Current server', 'phpMyAdmin demo - My...', and 'phpMyAdmin demo - My...'. In the center, a 'phpMyAdmin' window is open, showing a table named 'rental' with columns: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update. Below this, a table named 'customer' is shown with columns: first\_name, last\_name, address\_id, phone, email, and password. On the right, a terminal window displays an SQL query and its execution results. The query is:

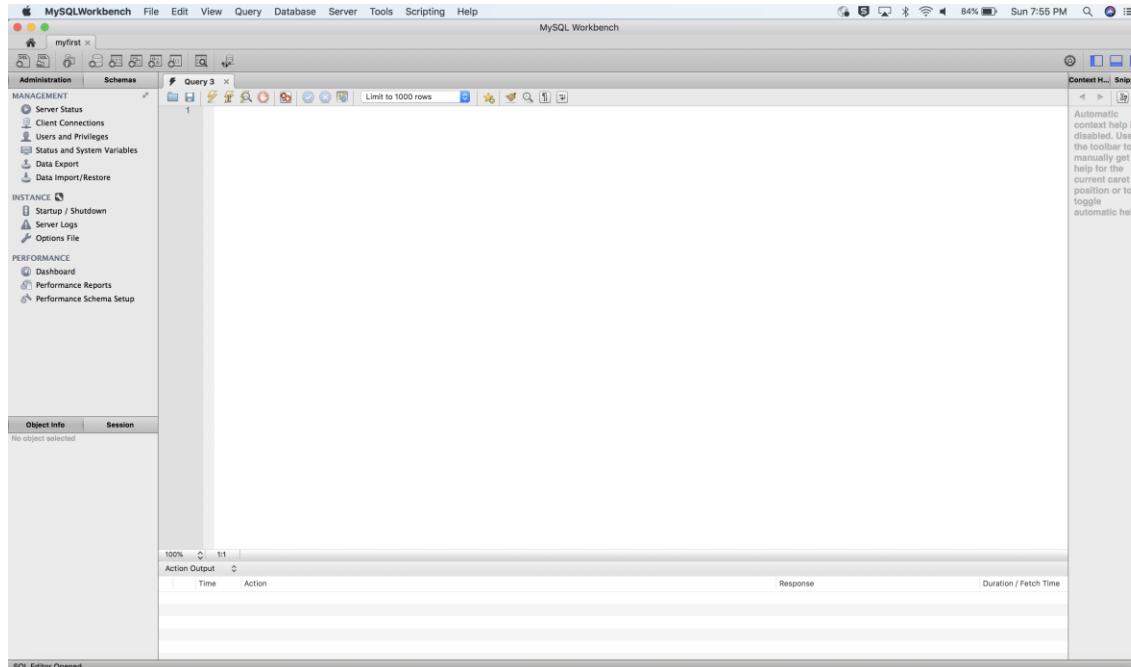
```
SELECT
    customer.first_name,
    customer.last_name
    COUNT(DISTINCT rental_id) AS rental_count
FROM
    inventory
    LEFT JOIN rental ON inventory.inventory_id = rental.inventory_id
    LEFT JOIN customer ON rental.customer_id = customer.customer_id
GROUP BY
    customer.first_name,
    customer.last_name
ORDER BY
    COUNT(DISTINCT rental_id) DESC
```

The results show 24 rows, indicating the number of rentals per customer, ordered from highest to lowest.

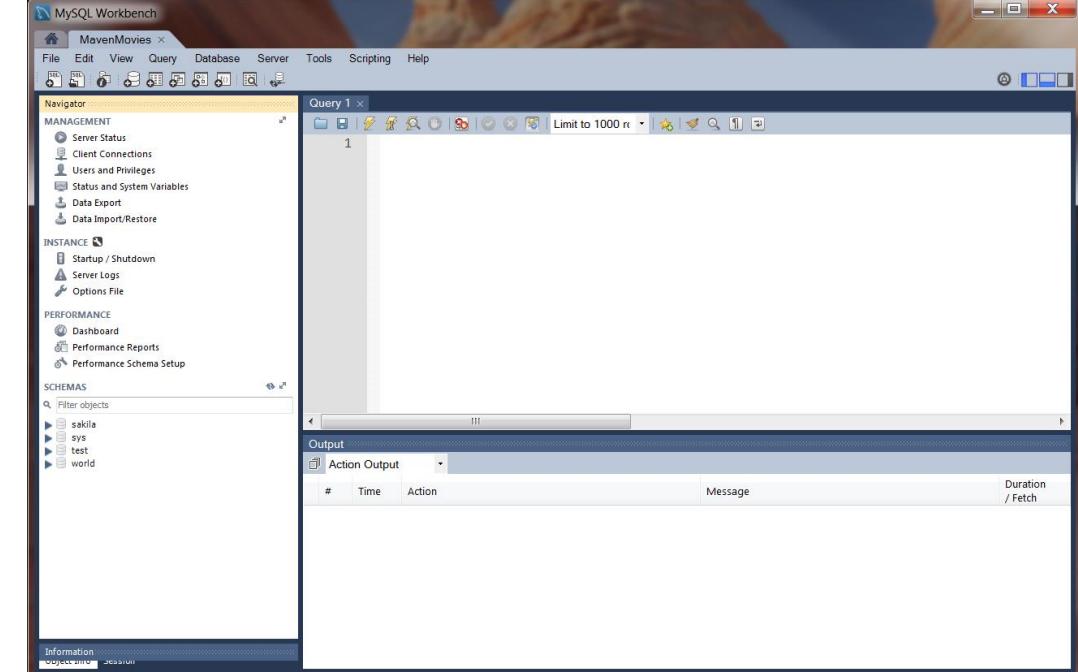
# MySQL WORKBENCH INTERFACE (MAC VS. PC)



*Mac interface*



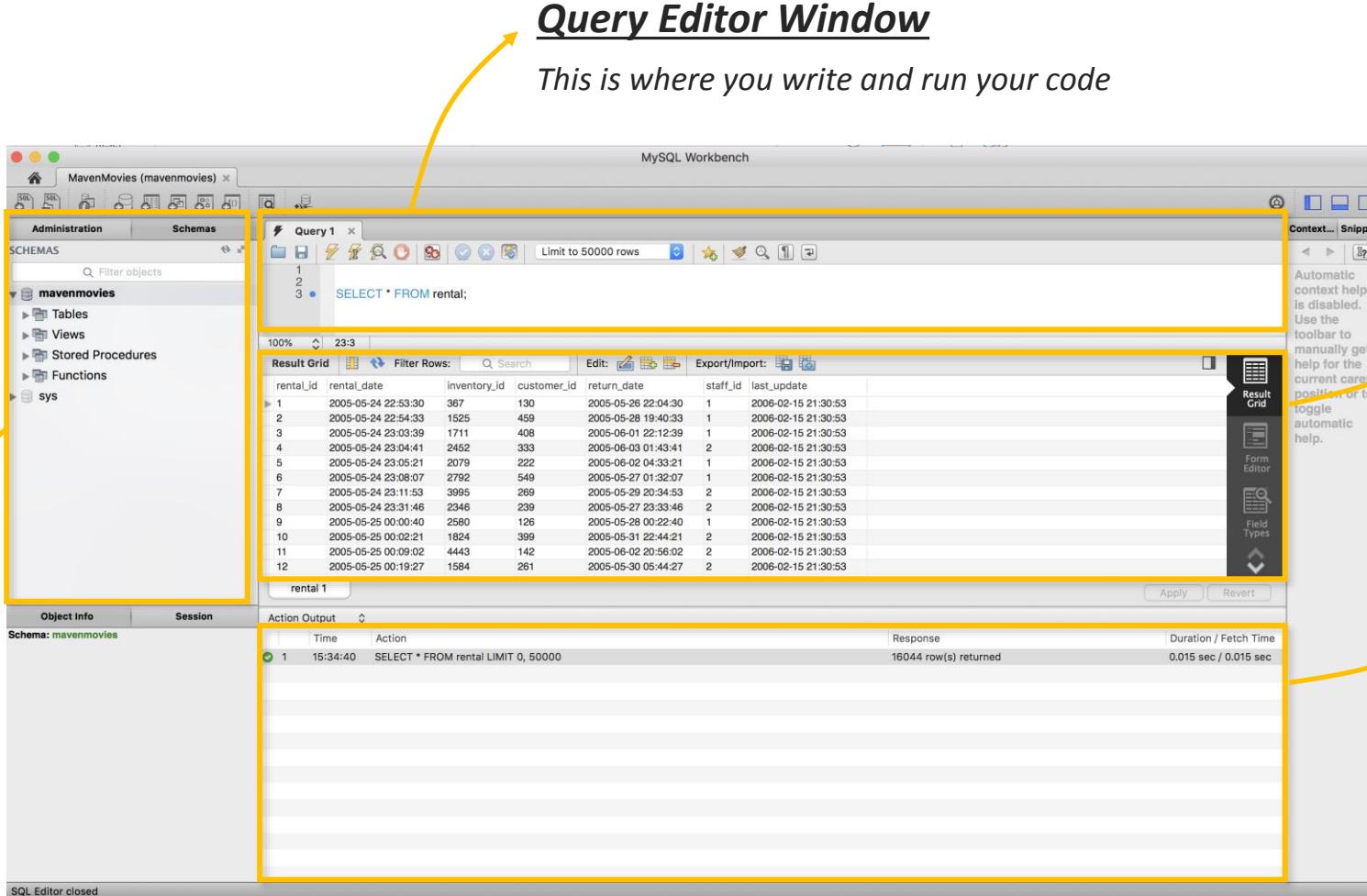
*PC interface*



## HEY THIS IS IMPORTANT!

Workbench looks slightly different on **Mac** vs. **PC**, but everything you need is found in the same place. While the course is recorded on a Mac, but you should have no problem keeping up on a PC

# QUICK TOUR: THE WORKBENCH INTERFACE



**Schemas Tab**  
Here you can view  
tables and views in  
your database

**Query Editor Window**  
*This is where you write and run your code*

**Result Grid**  
*After running your SQL  
queries, your results  
appear here*

**Action Output**  
*This is a summary of  
actions taken by the server  
(TIP: the '**Response**' column  
is great for troubleshooting  
errors!)*

# STEP 5: CREATING THE DATABASE

The screenshot displays the phpMyAdmin interface with the following components:

- Left Panel:** Shows a detailed database schema diagram with various tables like actor, address, category, customer, film, film\_actor, film\_category, inventory, and rental.
- Top Center:** A table titled "rental" showing data from rows 1 to 10. The columns include rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update.
- Middle Center:** A query results grid showing rows 0 to 24 of a query. The query is:

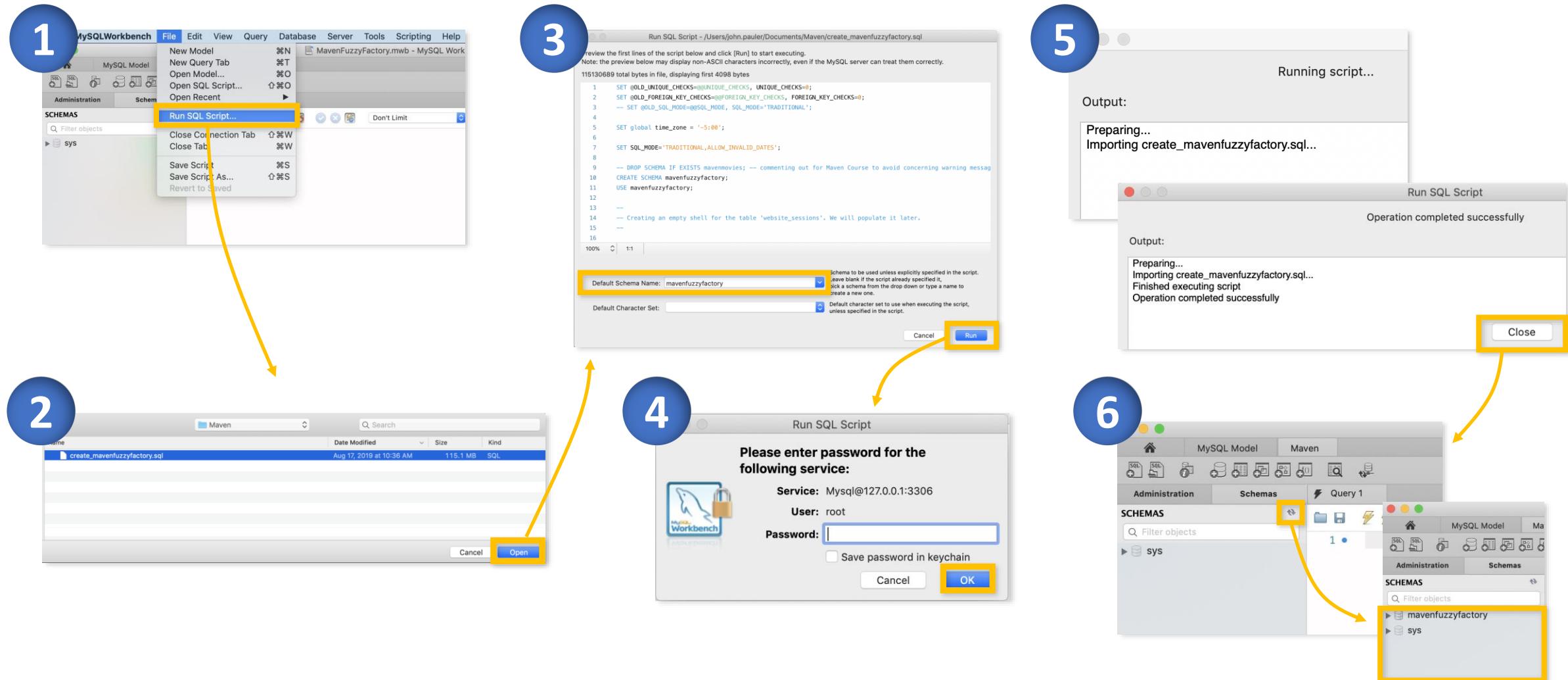
```
SELECT customer.first_name, customer.last_name, COUNT(DISTINCT rental_id) AS count FROM rental JOIN customer ON rental.customer_id = customer.customer_id GROUP BY customer.customer_id, customer.first_name, customer.last_name ORDER BY COUNT(DISTINCT rental_id) DESC
```
- Right Panel:** A table titled "privileges" listing various database objects and their privileges.

# CREATING THE DATABASE

---

- 1 In MySQL Workbench, click **File** from the top menu, then select **Run SQL Script**
  
- 2 Navigate to the ***create\_mavenfuzzyfactory.sql*** file provided in the course resources
  - *This code will automatically generate the entire database that we'll be exploring throughout the course, modeling a real-world eCommerce product company*
  
- 3 After running the code, confirm the following:
  1. *You see a list of results in the **Action Output** window, with **green check marks** and no errors in the **Response** column*
  2. *When you refresh the **Schemas** list, you should see a new database called **mavenfuzzyfactory**, containing **5 tables***

# CREATING THE DATABASE



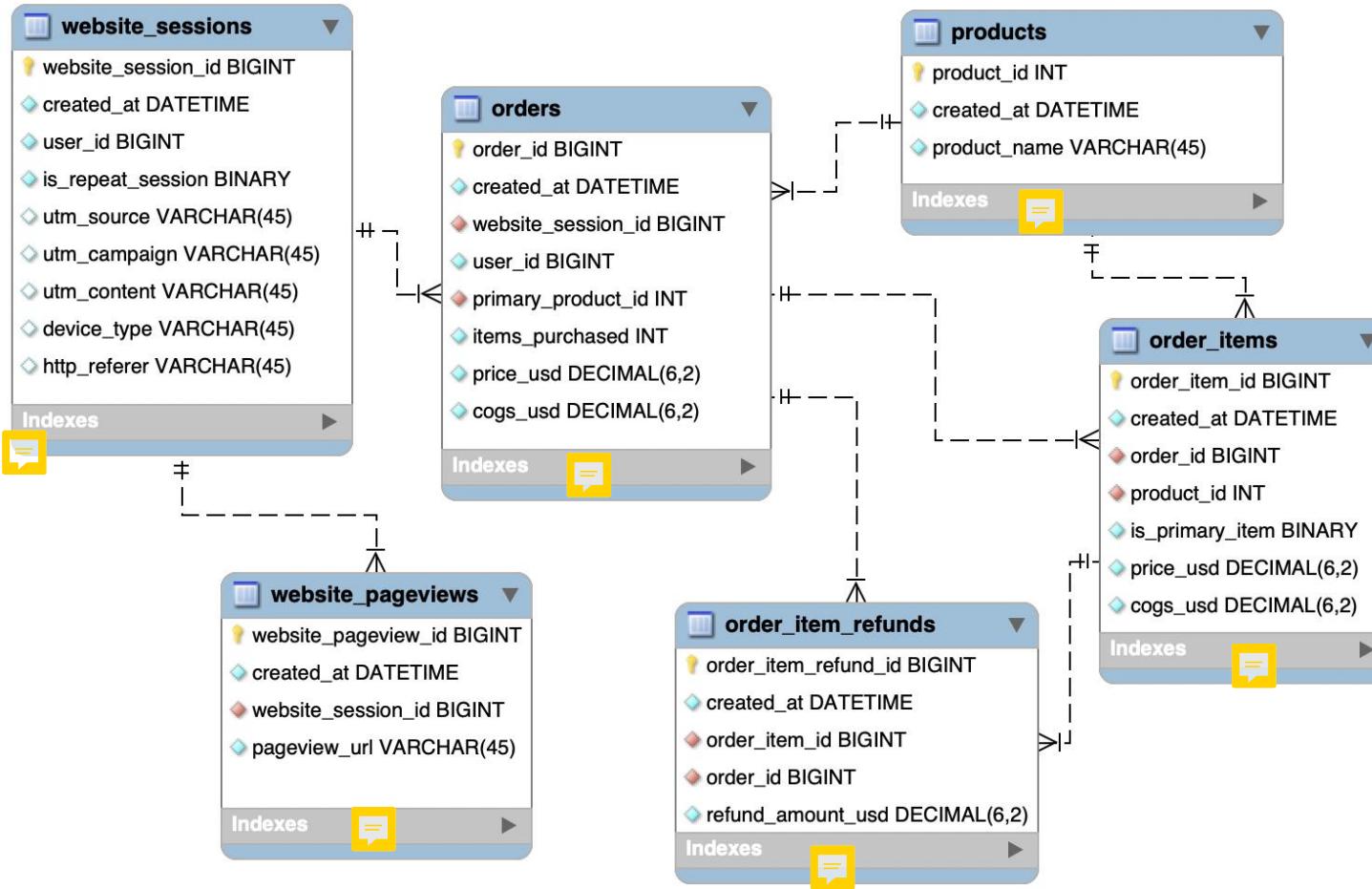
# GETTING TO KNOW THE DATABASE

The image shows a screenshot of the phpMyAdmin web application. At the top left, there's a navigation menu with links like 'Home', 'Logout', and 'Help'. Below it is a large, semi-transparent watermark-like graphic of the MySQL logo. The main content area is divided into several sections:

- Table Structure:** On the far left, a tree view shows the database schema with tables like 'actor', 'actor\_info', 'address', 'category', 'customer', 'language', 'language\_actor', 'language\_text', and 'Inventory'.
- Table Data:** A central grid displays data from the 'rental' table. The columns are: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update. The data shows various rental records with IDs ranging from 1 to 10, dates between May 2005 and February 2006, and staff IDs 1 or 2.
- Query Results:** To the right of the data grid, a results page shows the output of a query. It includes a header 'Showing rows 0 - 24 (599 total, Query took 0.0212 seconds.)' and the SQL code:

```
1 SELECT
2   customer.first_name,
3   customer.last_name,
4   COUNT(DISTINCT rental_id) AS rental_count
5   FROM
6   rental
7   JOIN
8   customer
9   ON customer.customer_id = rental.customer_id
10  GROUP BY
11    customer.first_name,
12    customer.last_name
13 ORDER BY
14   COUNT(DISTINCT rental_id) DESC
```
- Table Structure:** On the far right, another grid shows the structure of the 'rental' table with columns: id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update.

# OVERVIEW OF THE MAVEN FUZZY FACTORY DATABASE

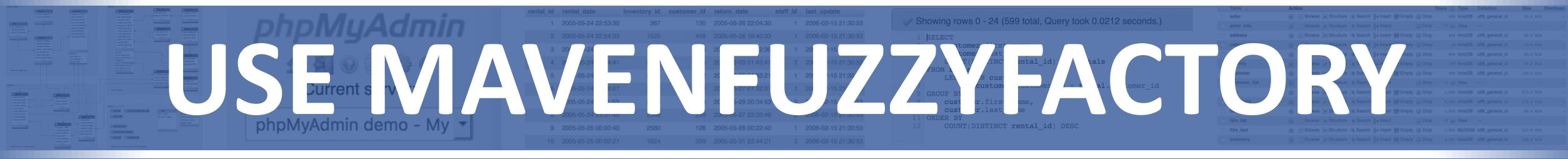


We will be working with six related tables, which contain eCommerce data about:

- **Website Activity**
- **Products**
- **Orders and Refunds**

We'll use MySQL to understand how customers access and interact with the site, analyze landing page performance and conversion, and explore product-level sales.

# USE MAVENFUZZYFACTORY



## HEY THIS IS IMPORTANT!

The USE statement identifies the schema you will be selecting data from in Workbench.

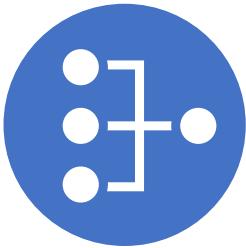
*Example: USE mavenfuzzyfactory;*

If you encounter an error that says 'no database selected', you'll need to select your database with a USE statement

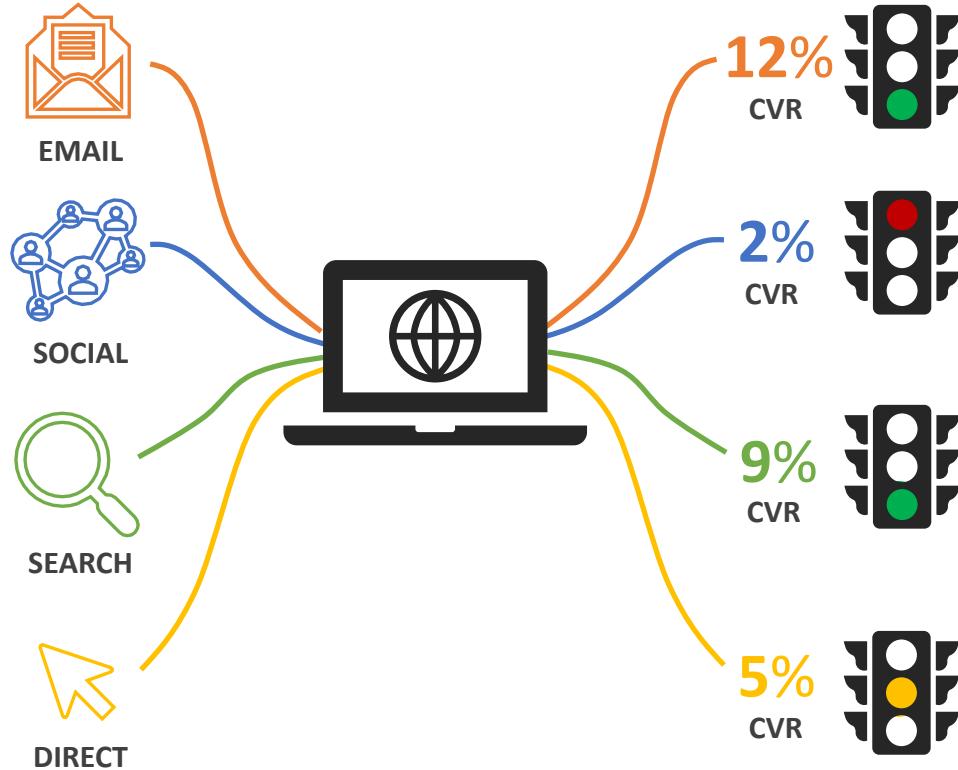
# ANALYZING TRAFFIC SOURCES



# BUSINESS CONCEPT: TRAFFIC SOURCE ANALYSIS



Traffic source analysis is about understanding **where your customers are coming from** and **which channels are driving the highest quality traffic**



## COMMON USE CASES:

- Analyzing search data and shifting budget towards the engines, campaigns or keywords driving the strongest conversion rates
- Comparing user behavior patterns across traffic sources to inform creative and messaging strategy
- Identifying opportunities to eliminate wasted spend or scale high-converting traffic

# KEY TABLES: WEBSITE SESSIONS, PAGEVIEWS & ORDERS

## WEBSITE\_SESSIONS

**SELECT \* FROM website\_sessions WHERE website\_session\_id = 1059**

website_session_id	created_at	user_id	is_repeat_session	utm_source	utm_campaign	utm_content	device_type	http_referer
1059	2012-03-26 13:51:37	1055	0	gsearch	nonbrand	g_ad_1	desktop	https://www.gsearch.com

## WEBSITE\_PAGEVIEWS

**SELECT \* FROM website\_pageviews WHERE website\_session\_id = 1059**

website_pageview_id	created_at	website_session_id	pageview_url
2039	2012-03-26 13:51:37	1059	/home
2040	2012-03-26 13:54:27	1059	/products
2041	2012-03-26 13:56:48	1059	/the-original-mr-fuzzy
2042	2012-03-26 14:00:14	1059	/cart
2043	2012-03-26 14:04:06	1059	/shipping
2044	2012-03-26 14:05:47	1059	/billing
2045	2012-03-26 14:13:56	1059	/thank-you-for-your-order

## ORDERS

**SELECT \* FROM orders WHERE website\_session\_id = 1059**

order_id	created_at	website_session_id	user_id	primary_product_id	items_purchased	price_usd	cogs_usd
32	2012-03-26 14:13:56	1059	1055	1	1	49.99	19.49

# PAID MARKETING CAMPAIGNS: UTM TRACKING PARAMETERS

When businesses run paid marketing campaigns, they often obsess over performance and measure *everything*; how much they spend, how well traffic converts to sales, etc.

Paid traffic is commonly tagged with tracking (UTM) parameters, which are appended to URLs and allow us to tie website activity back to specific traffic sources and campaigns

www.abcwebsite.com?utm\_source=trafficSource&utm\_campaign=campaignName

The diagram illustrates the flow from a URL to a database query and its results. On the left, a URL is shown: www.abcwebsite.com?utm\_source=trafficSource&utm\_campaign=campaignName. Two blue arrows point from this URL to two separate components on the right. The first arrow points to a dark blue box labeled "WEBSITE\_SESSIONS". Inside this box is a SQL query:

```
SELECT DISTINCT
    utm_source,
    utm_campaign
FROM website_sessions
```

The second arrow points to a light gray table titled "utm\_source | utm\_campaign". This table contains the following data:

utm_source	utm_campaign
HULL	HULL
bsearch	brand
bsearch	nonbrand
gsearch	brand
gsearch	nonbrand
socialbook	desktop_targeted
socialbook	pilot

# TRAFFIC SOURCE ANALYSIS

- We use the utm parameters stored in the database to identify paid website sessions

- From our session data, we can link to our order data to understand how much revenue our paid campaigns are driving

## PRO TIP:

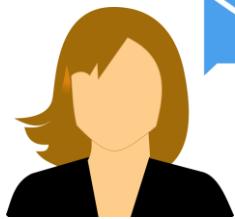
Use *GROUP BY* with *COUNT()* and *SUM()* to identify critical traffic drivers

## MySQL QUERY IN ACTION:

```
SELECT
    utm_source,
    COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
    COUNT(DISTINCT orders.order_id) AS orders
FROM website_sessions
LEFT JOIN orders
    ON orders.website_session_id = website_sessions.website_session_id
WHERE website_sessions.website_session_id BETWEEN 1000 and 2000 -- arbitrary
GROUP BY
    utm_source
```

## QUERY RESULTS:

utm_source	sessions	orders
HULL	18	0
bsearch	2	0
gsearch	981	35



## NEW MESSAGE

April 12, 2012

From: Cindy Sharp (CEO)

Subject: Site traffic breakdown

Good morning,

We've been live for almost a month now and we're starting to generate sales. Can you help me understand where the bulk of our website sessions are coming from, through yesterday?

I'd like to see a breakdown by **UTM source, campaign** and **referring domain** if possible. Thanks!

-Cindy

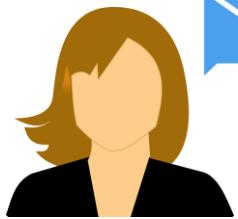
Reply

Forward

## Result Preview

Result Grid			Filter Rows:	Search	Export:	
utm_source	utm_campaign	http_referer	sessions			
			3613			
			28			
			27			
			26			
			7			
			7			

# TEST YOUR SKILLS: FINDING TOP TRAFFIC SOURCES



## NEW MESSAGE

April 12, 2012

From: **Cindy Sharp (CEO)**  
Subject: **Site traffic breakdown**

Good morning,

We've been live for almost a month now and we're starting to generate sales. Can you help me understand where the bulk of our website sessions are coming from, through yesterday?

I'd like to see a breakdown by **UTM source, campaign** and **referring domain** if possible. Thanks!

-Cindy

Reply

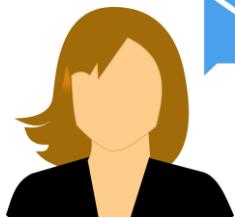
Forward

## Solution Query

```
USE mavenfuzzyfactory;

SELECT
    utm_source,
    utm_campaign,
    http_referer,
    COUNT(website_session_id) AS sessions
FROM website_sessions
WHERE created_at < '2012-04-12' -- this line is required
GROUP BY
    utm_source,
    utm_campaign,
    http_referer
ORDER BY
    sessions DESC;
```

TEST YOUR SKILLS: FINDING TOP TRAFFIC SOURCES



## NEW MESSAGE

April 12, 2012

From: Cindy Sharp (CEO)

Subject: RE: Site traffic breakdown

Great analysis!

Based on your findings, it seems like we should probably dig into **gsearch nonbrand** a bit deeper to see what we can do to optimize there.

I'll loop in Tom tomorrow morning to get his thoughts on next steps.

-Cindy

Reply

Forward

## Result Grid



Filter Rows:

Search

Export:

utm_source	utm_campaign	http_referer	sessions
gsearch	nonbrand	https://www.gsearch.com	3613
NULL	NULL	NULL	28
NULL	NULL	https://www.gsearch.com	27
gsearch	brand	https://www.gsearch.com	26
bsearch	brand	https://www.bsearch.com	7
NULL	NULL	https://www.bsearch.com	7

## NEXT STEPS:

- *Drill deeper into **gsearch nonbrand** campaign traffic to explore potential optimization opportunities*
- *Await further instruction from Tom*

TEST YOUR SKILLS: FINDING TOP TRAFFIC SOURCES



## NEW MESSAGE

April 14, 2012

From: Tom Parmesan (Marketing Director)

Subject: Gsearch conversion rate

Hi there,

Sounds like gsearch nonbrand is our major traffic source, but we need to understand if those sessions are driving sales.

Could you please **calculate the conversion rate (CVR) from session to order?** Based on what we're paying for clicks, we'll need a CVR of **at least 4%** to make the numbers work.

If we're much lower, we'll need to reduce bids. If we're higher, we can increase bids to drive more volume.

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

sessions	orders	session_to_order_conv_rate
[REDACTED]	[REDACTED]	[REDACTED]



# TEST YOUR SKILLS: TRAFFIC CONVERSION RATES



## NEW MESSAGE

April 14, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Gsearch conversion rate**

Hi there,

Sounds like gsearch nonbrand is our major traffic source, but we need to understand if those sessions are driving sales.

Could you please **calculate the conversion rate (CVR) from session to order?** Based on what we're paying for clicks, we'll need a CVR of **at least 4%** to make the numbers work.

If we're much lower, we'll need to reduce bids. If we're higher, we can increase bids to drive more volume.

Thanks, Tom

Reply

Forward

## Solution Query

```
USE mavenfuzzyfactory;

SELECT
    COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
    COUNT(DISTINCT orders.order_id) AS orders
FROM website_sessions
LEFT JOIN orders
    ON orders.website_session_id = website_sessions.website_session_id
WHERE website_sessions.created_at < '2012-04-14'
    AND utm_source = 'gsearch'
    AND utm_campaign = 'nonbrand'
```

# TEST YOUR SKILLS: TRAFFIC CONVERSION RATES



## NEW MESSAGE

April 14, 2012

From: **Tom Parmesan (Marketing Director)**  
Subject: RE: Gsearch conversion rate

Hmm, looks like we're below the **4%** threshold we need to make the economics work.

Based on this analysis, we'll need to dial down our search bids a bit. We're over-spending based on the current conversion rate.

Nice work, your analysis just saved us some \$\$\$!

Reply

Forward

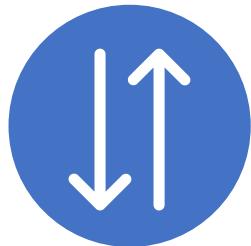
sessions	orders	session_to_order_conv_rt
3895	112	0.0288

### NEXT STEPS:

- *Monitor the impact of bid reductions*
- *Analyze performance trending by device type in order to refine bidding strategy*

# TEST YOUR SKILLS: TRAFFIC CONVERSION RATES

# BUSINESS CONCEPT: BID OPTIMIZATION



Analyzing for bid optimization is about **understanding the value of various segments of paid traffic, so that you can optimize your marketing budget**



## COMMON USE CASES:

- Using conversion rate and revenue per click analyses to figure out how much you should spend per click to acquire customers
- Understanding how your website and products perform for various subsegments of traffic (i.e. mobile vs desktop) to optimize within channels
- Analyzing the impact that bid changes have on your ranking in the auctions, and the volume of customers driven to your site

# MySQL DATE FUNCTIONS

## MONTH()

A function you can use to extract the month from a date or datetime (just one example)

### MONTH(dateOrDatetime)

This lets the SQL server know you want to extract the month from the date or datetime value

This is where you prescribe which date or datetime value your function will operate on

Examples:

- `MONTH(created_at) AS created_month`
- `YEAR(created_at) AS created_year`
- `WEEK(updated_at) AS updated_week`



### HEY THIS IS IMPORTANT!

MySQL has LOTS of date functions.

We will only cover the most common ones for use this course. You should get good at googling 'MySQL date functions' anytime you need to work with dates.



### PRO TIP:

Use date functions with **GROUP BY** and aggregate functions like **COUNT()** and **SUM()** to show trending

# DATE FUNCTIONS

- The powerful functions below can all be used with **GROUP BY** to provide trended summaries

Function	How You Might Use It
<b>YEAR()</b>	Return the year for a given date
<b>QUARTER()</b>	Return the quarter for a given date
<b>MONTH()</b>	Return the month for a given date
<b>WEEK()</b>	Return the week for a given date
<b>DATE()</b>	Return the date for a given datetime
<b>NOW()</b>	Calculate time relative to now

## MySQL QUERY IN ACTION:

**SELECT**

```
YEAR(created_at) AS created_yr,  
WEEK(created_at) AS created_wk,  
COUNT(DISTINCT website_session_id) AS sessions  
FROM website_sessions  
WHERE website_session_id BETWEEN 100000 and 115000 -- arbitrary  
GROUP BY  
YEAR(created_at),  
week(created_at)
```

## QUERY RESULTS:

Result Grid		
created_yr	created_wk	sessions
2013	22	883
2013	23	1920
2013	24	2066
2013	25	2027
2013	26	1919
2013	27	1938
2013	28	2007
2013	29	2052
2013	30	189

# PRO TIP: “PIVOTING” DATA WITH COUNT & CASE

## CASE “PIVOTS”

*Excel’s ability to pivot to columns can be replicated in SQL using COUNT and CASE*

Excel makes it very easy to “pivot” data on two dimensions.

Here we’re breaking down the count of **order\_id** by **primary\_product\_id** (**rows**) and **items\_purchased** (**columns**) to see how many orders were placed for each primary product and how many of those orders included multiple items.

MySQL can do the same thing using **COUNT** functions inside of a **CASE** statement:

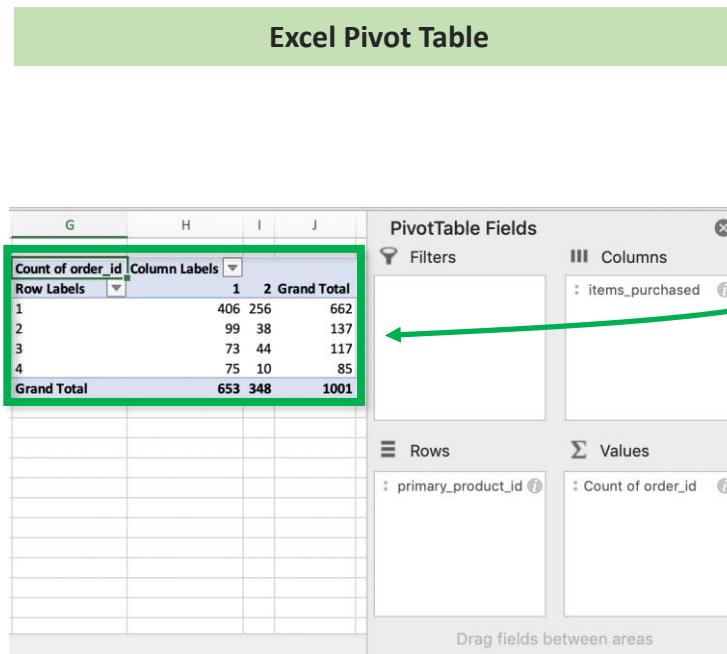
**Orders Table**

```
1 • use mavenfuzzyfactory;
2
3 • SELECT
4     order_id,
5     primary_product_id,
6     items_purchased,
7     created_at
8
9 FROM orders
10 WHERE order_id BETWEEN 31000 AND 32000
```

Result Grid

order\_id primary\_product\_id items\_purchased created\_at

31000	1	2	2015-02-27 16:54:42
31001	1	2	2015-02-27 16:57:03
31002	1	1	2015-02-27 17:07:27
31003	3	1	2015-02-27 17:08:27
31004	1	1	2015-02-27 17:11:45
31005	1	2	2015-02-27 17:16:30
31006	1	1	2015-02-27 17:26:17
31007	1	1	2015-02-27 17:45:28
31008	1	1	2015-02-27 17:45:34
31009	1	1	2015-02-27 17:52:59
31010	3	1	2015-02-27 18:02:58
31011	2	1	2015-02-27 18:02:59
31012	1	1	2015-02-27 18:07:32
31013	1	1	2015-02-27 19:40:15
31014	1	1	2015-02-27 21:33:02
31015	1	1	2015-02-27 22:09:57



*Both methods yield identical results*

**MySQL CASE STATEMENT**

```
1 • use mavenfuzzyfactory;
2
3 • SELECT
4     primary_product_id,
5     COUNT(DISTINCT CASE WHEN items_purchased = 1 THEN order_id ELSE NULL END) AS orders_w_1_item,
6     COUNT(DISTINCT CASE WHEN items_purchased = 2 THEN order_id ELSE NULL END) AS orders_w_2_items,
7     COUNT(DISTINCT order_id) AS total_orders
8
9 FROM orders
10 WHERE order_id BETWEEN 31000 AND 32000
11 GROUP BY 1
```

Result Grid

primary_product_id	orders_w_1_item	orders_w_2_items	total_orders
1	406	256	662
2	99	38	137
3	73	44	117
4	75	10	85

# CASE & COUNT

- The *orders\_w\_1\_item* column is created by counting *order\_id* values for records which have a value of 1 in the *items\_purchased* column
- This method of counting records where a condition is true can be incredibly useful!

## PRO TIP:

Use **GROUP BY** to define your row labels, and **CASE** to pivot to columns

## MySQL QUERY IN ACTION:

```
1 • use mavenfuzzyfactory;
2
3 • SELECT
4     primary_product_id,
5     COUNT(DISTINCT CASE WHEN items_purchased = 1 THEN order_id ELSE NULL END) AS orders_w_1_item,
6     COUNT(DISTINCT CASE WHEN items_purchased = 2 THEN order_id ELSE NULL END) AS orders_w_2_items,
7     COUNT(DISTINCT order_id) AS total_orders
8 FROM orders
9 WHERE order_id BETWEEN 31000 AND 32000
10 GROUP BY 1
```

You can identify which column to GROUP BY quickly by specifying it's column order within the SELECT statement. In this case, *primary\_product\_id* is column 1, so we can simply write 'GROUP BY 1'

## ORIGINAL TABLE:

Result Grid			
order_id	primary_product_id	items_purchased	created_at
31000	1	2	2015-02-27 16:54:42
31001	1	2	2015-02-27 16:57:03
31002	1	1	2015-02-27 17:07:27
31003	3	1	2015-02-27 17:08:27
31004	1	1	2015-02-27 17:11:45
31005	1	2	2015-02-27 17:16:30
31006	1	1	2015-02-27 17:26:17
31007	1	1	2015-02-27 17:45:28
31008	1	1	2015-02-27 17:45:34
31009	1	1	2015-02-27 17:52:59
31010	3	1	2015-02-27 18:02:58
31011	2	1	2015-02-27 18:02:59
31012	1	1	2015-02-27 18:07:32
31013	1	1	2015-02-27 19:40:15
31014	1	1	2015-02-27 21:33:02
31015	1	1	2015-02-27 22:09:57
31016	1	1	2015-02-27 22:20:12
31017	1	2	2015-02-27 23:08:20
31018	1	1	2015-02-27 23:14:36
31019	1	2	2015-02-27 23:41:34
31020	2	2	2015-02-27 23:42:50
31021	4	1	2015-02-27 23:48:48

## QUERY RESULTS:

Result Grid			
primary_product_id	orders_w_1_item	orders_w_2_items	total_orders
1	406	256	662
2	99	38	137
3	73	44	117
4	75	10	85



## NEW MESSAGE

May 12, 2012

From: Tom Parmesan (Marketing Director)

Subject: Gsearch volume trends

Hi there,

Based on your conversion rate analysis, we **bid down gsearch nonbrand** on 2012-04-15.

Can you pull **gsearch nonbrand trended session volume, by week**, to see if the bid changes have caused volume to drop at all?

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter

week_start_date	sessions
-----------------	----------

▶ 2012-03-19	
2012-03-25	
2012-04-01	
2012-04-08	
2012-04-15	
2012-04-22	
2012-04-29	
2012-05-06	

# TEST YOUR SKILLS: TRAFFIC SOURCE TRENDING



## NEW MESSAGE

May 10, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Gsearch volume trends**

Hi there,

Based on your conversion rate analysis, we **bid down gsearch nonbrand** on 2012-04-15.

Can you pull **gsearch nonbrand trended session volume, by week**, to see if the bid changes have caused volume to drop at all?

Thanks, Tom

Reply

Forward

## Solution Query

```
USE mavenfuzzyfactory;

SELECT
    -- YEARWEEK(website_sessions.created_at) AS year_week,
    MIN(DATE(created_at)) AS week_start_date,
    COUNT(DISTINCT website_sessions.website_session_id) AS sessions

FROM website_sessions

WHERE website_sessions.created_at < '2012-05-10'
    AND website_sessions.utm_source = 'gsearch'
    AND website_sessions.utm_campaign = 'nonbrand'

GROUP BY
    YEARWEEK(website_sessions.created_at)
```

# TEST YOUR SKILLS: TRAFFIC SOURCE TRENDING



## NEW MESSAGE

May 10, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **RE: Gsearch volume trends**

Hi there, great analysis!

Okay, based on this, it does look like **gsearch nonbrand** is fairly sensitive to bid changes.

We want maximum volume, but don't want to spend more on ads than we can afford.

Let me think on this. I will likely follow up with some ideas.

Thanks, Tom

Reply

Forward

## Result Grid



Filter

week_start_date	sessions
2012-03-19	896
2012-03-25	956
2012-04-01	1152
2012-04-08	983
2012-04-15	621
2012-04-22	594
2012-04-29	681
2012-05-06	399

## NEXT STEPS:

- *Continue to monitor volume levels*
- *Think about how we could make the campaigns more efficient so that we can increase volume again*

# TEST YOUR SKILLS: TRAFFIC SOURCE TRENDING



## NEW MESSAGE

May 11, 2012

From: **Tom Parmesan (Marketing Director)**  
Subject: Gsearch device-level performance

Hi there,

I was trying to use our site on my mobile device the other day, and the experience was not great.

Could you pull **conversion rates from session to order**, by **device type**?

If desktop performance is better than on mobile we may be able to bid up for desktop specifically to get more volume?

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

device_type	sessions	orders	session_to_order_conv_rate
mobile	[REDACTED]	[REDACTED]	[REDACTED]
desktop	[REDACTED]	[REDACTED]	[REDACTED]

# TEST YOUR SKILLS: TRAFFIC SOURCE BID OPTIMIZATION



## NEW MESSAGE

May 11, 2012

From: Tom Parmesan (*Marketing Director*)  
Subject: Gsearch device-level performance

Hi there,

I was trying to use our site on my mobile device the other day, and the experience was not great.

Could you pull **conversion rates from session to order, by device type?**

If desktop performance is better than on mobile we may be able to bid up for desktop specifically to get more volume?

Thanks, Tom

Reply

Forward

## Solution Query

```
SELECT
    website_sessions.device_type,
    COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
    COUNT(DISTINCT orders.order_id) AS orders,
    COUNT(DISTINCT orders.order_id)/COUNT(DISTINCT website_sessions.website_session_id) AS conv_rt
FROM website_sessions
LEFT JOIN orders
    ON orders.website_session_id = website_sessions.website_session_id
WHERE website_sessions.created_at < '2012-05-11'
    AND utm_source = 'gsearch'
    AND utm_campaign = 'nonbrand'

GROUP BY 1
```

TEST YOUR SKILLS: TRAFFIC SOURCE BID OPTIMIZATION



## NEW MESSAGE

May 11, 2012

From: **Tom Parmesan (Marketing Director)**  
Subject: RE: Gsearch device-level performance

Great!

I'm going to increase our bids on desktop.

When we bid higher, we'll rank higher in the auctions, so I think your insights here should lead to a sales boost.

Well done!!

-Tom

Reply

Forward

## Result Grid



Filter Rows:

device_type	sessions	orders	conv_rt
desktop	3911	146	0.0373
mobile	2492	24	0.0096

## NEXT STEPS:

- *Analyze volume by device type to see if the bid changes make a material impact*
- *Continue to look for ways to optimize campaigns*

# TEST YOUR SKILLS: TRAFFIC SOURCE BID OPTIMIZATION



## NEW MESSAGE

June 09, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Gsearch device-level trends**

Hi there,

After your device-level analysis of conversion rates, we realized desktop was doing well, so **we bid our gsearch nonbrand desktop campaigns up on 2012-05-19**.

Could you pull **weekly trends for both desktop and mobile** so we can see the impact on volume?

You can use 2012-04-15 until the bid change as a baseline.

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

week_start_date	dtop_sessions	mob_sessions
2012-04-15		
2012-04-22		
2012-04-29		
2012-05-06		
2012-05-13		
2012-05-20		
2012-05-27		
2012-06-03		

TEST YOUR SKILLS: TRAFFIC SOURCE SEGMENT TRENDING



## NEW MESSAGE

June 09, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Gsearch device-level trends**

Hi there,

After your device-level analysis of conversion rates, we realized desktop was doing well, so **we bid our gsearch nonbrand desktop campaigns up on 2012-05-19.**

Could you pull **weekly trends for both desktop and mobile** so we can see the impact on volume?

You can use 2012-04-15 until the bid change as a baseline.

Thanks, Tom

Reply

Forward

## Solution Query

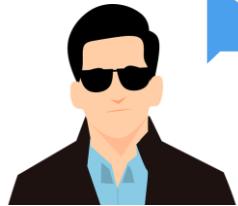
```
SELECT
    -- YEARWEEK(website_sessions.created_at) AS year_week,
    MIN(DATE(created_at)) AS week_start_date,
    COUNT(DISTINCT CASE WHEN device_type = 'desktop' THEN website_sessions.website_session_id ELSE NULL END) AS dtop_sessions,
    COUNT(DISTINCT CASE WHEN device_type = 'mobile' THEN website_sessions.website_session_id ELSE NULL END) AS mob_sessions

FROM website_sessions

WHERE website_sessions.created_at < '2012-06-09'
    AND website_sessions.created_at > '2012-04-15'
    AND website_sessions.utm_source = 'gsearch'
    AND website_sessions.utm_campaign = 'nonbrand'

GROUP BY
    YEARWEEK(website_sessions.created_at)
```

TEST YOUR SKILLS: TRAFFIC SOURCE SEGMENT TRENDING



## NEW MESSAGE

June 09, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **RE: Gsearch device-level trends**

Nice work digging into this!

It looks like mobile has been pretty flat or a little down, but **desktop is looking strong thanks to the bid changes we made based on your previous conversion analysis.**

Things are moving in the right direction!

Thanks, Tom

Reply

Forward

week_start_date	dtop_sessions	mob_sessions
2012-04-15	383	238
2012-04-22	360	234
2012-04-29	425	256
2012-05-06	430	282
2012-05-13	403	214
2012-05-20	661	190
2012-05-27	585	183
2012-06-03	582	157

## NEXT STEPS:

- Continue to monitor device-level volume and be aware of the impact bid levels has
- Continue to monitor conversion performance at the device-level to optimize spend

TEST YOUR SKILLS: TRAFFIC SOURCE SEGMENT TRENDING

# ANALYZING WEBSITE PERFORMANCE

The image is a horizontal collage of three screenshots related to database management and analysis:

- Left Screenshot:** A MySQL database schema diagram showing various tables like actor, address, category, customer, film, film\_actor, film\_category, inventory, and rental.
- Middle Screenshot:** A screenshot of the phpMyAdmin interface for a demo database. It shows a table named 'rental' with columns: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update. The data includes rows for rentals from May 24, 2005, to May 25, 2005, involving different customers and staff members.
- Right Screenshot:** A screenshot of a MySQL command-line interface showing a query results page. The query is: 

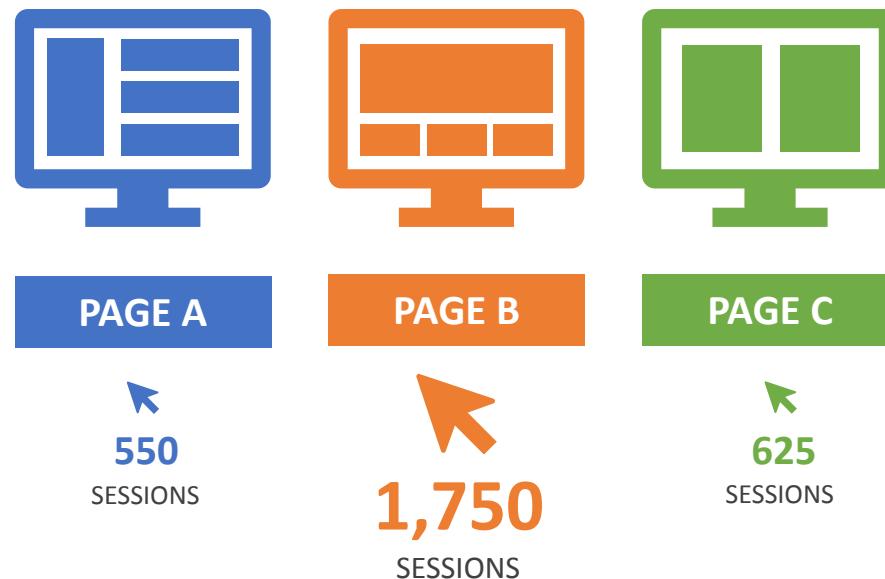
```
1 SELECT
2   COUNT(DISTINCT rental_id) AS totals
3   FROM rental
4   JOIN customer
5   ON rental.customer_id = customer.customer_id
6   GROUP BY
7   customer.customer_id
8   ORDER BY
9   COUNT(DISTINCT rental_id) DESC
```

. The results show the total number of unique rentals per customer, with the top result being 599.

# BUSINESS CONCEPT: ANALYZING TOP WEBSITE CONTENT



Website content analysis is about **understanding which pages are seen the most by your users, to identify where to focus on improving your business**



## COMMON USE CASES:

- Finding the most-viewed pages that customers view on your site
- Identifying the most common entry pages to your website – the first thing a user sees
- For most-viewed pages and most common entry pages, understanding how those pages perform for your business objectives

# CREATING TEMPORARY TABLES

## CREATE TEMPORARY TABLE

*Allows you to create a dataset stored as a table which you can query*

### CREATE TEMPORARY TABLE newTempTableName

*This lets the SQL server know you want to create a temporary table*

*This is where you give a name to your temporary table so you can query it later*

*Example:*

```
CREATE TEMPORARY TABLE first_hundred_sessions  
SELECT * FROM website_sessions  
WHERE website_session_id <= 100
```



#### HEY THIS IS IMPORTANT!

Temporary tables persist only for the current MySQL Workbench session. Using temporary tables for analyses keeps you from creating endless permanent tables, but keep in mind they won't be there anymore if you shut down Workbench and have to sign in again. **So, save your code!**



#### PRO TIP:

*Using temporary tables to perform **multi-step analyses** is where you transition from “I can do all the basics” to **becoming unstoppable as an Analyst** (and developing into more of a “**data programmer**”)*

# FINDING TOP PAGES

- We can analyze our pageviews data and GROUP BY url to see which pages are viewed most

- To find top entry pages, we will limit to just the first page a user sees during a given session, using a temporary table

## PRO TIP:

Performing a top pages analysis is a quick way to decide **where to focus**

## MySQL QUERY IN ACTION:

**SELECT**

```
pageview_url,  
COUNT(DISTINCT website_pageview_id) AS views  
FROM website_pageviews  
WHERE website_pageview_id < 1000  
GROUP BY 1  
ORDER BY 2 DESC;
```

## QUERY RESULTS:

pageview_url	views
/home	523
/products	195
/the-original-mr-fuzzy	134
/cart	56
/shipping	39
/billing	34
/thank-you-for-your-order	18



## NEW MESSAGE

June 09, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Top Website Pages**

Hi there!

I'm Morgan, the new Website Manager.

Could you help me get my head around the site by pulling the **most-viewed website pages, ranked by session volume?**

Thanks!

-Morgan

Reply

Forward

## Result Preview

### Result Grid



### Filter Rows:



pageview\_url

sessions

TEST YOUR SKILLS: IDENTIFYING TOP WEBSITE PAGES



## NEW MESSAGE

June 09, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Top Website Pages**

Hi there!

I'm Morgan, the new Website Manager.

Could you help me get my head around the site by pulling the **most-viewed website pages, ranked by session volume?**

Thanks!

-Morgan

Reply

Forward

## Solution Query

```
USE mavenfuzzyfactory;

SELECT
    pageview_url,
    COUNT(DISTINCT website_session_id) AS sessions
FROM website_pageviews
WHERE created_at < '2012-06-09'
GROUP BY
    pageview_url
ORDER BY
    sessions DESC
```

# TEST YOUR SKILLS: IDENTIFYING TOP WEBSITE PAGES



## NEW MESSAGE

June 09, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **RE: Top Website Pages**

Thank you!

It definitely seems like **the homepage, the products page, and the Mr. Fuzzy page get the bulk of our traffic.**

I would like to understand traffic patterns more.

I'll follow up soon with a request to **look at entry pages.**

Thanks!

-Morgan

Reply

Forward

pageview_url	sessions
/home	10398
/products	4238
/the-original-mr-fuzzy	3036
/cart	1305
/shipping	869
/billing	716
/thank-you-for-your-order	306

## NEXT STEPS:

- *Dig into whether this list is also representative of our top entry pages*
- *Analyze the performance of each of our top pages to look for improvement opportunities*

# TEST YOUR SKILLS: IDENTIFYING TOP WEBSITE PAGES



## NEW MESSAGE

June 12, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Top Entry Pages**

Hi there!

Would you be able to pull a **list of the top entry pages**? I want to confirm where our users are hitting the site.

If you could **pull all entry pages and rank them on entry volume**, that would be great.

Thanks!

-Morgan

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

landing\_page

sessions\_hitting\_this\_landing\_page

TEST YOUR SKILLS: IDENTIFYING TOP ENTRY PAGES



## NEW MESSAGE

June 12, 2012

From: **Morgan Rockwell (Website Manager)**  
Subject: **Top Entry Pages**

Hi there!

Would you be able to pull a **list of the top entry pages**? I want to confirm where our users are hitting the site.

If you could **pull all entry pages and rank them on entry volume**, that would be great.

Thanks!

-Morgan

Reply

Forward

## Solution Query

```
USE mavenfuzzyfactory;
CREATE TEMPORARY TABLE first_pageviews
SELECT
    website_session_id,
    MIN(website_pageview_id) AS min_pageview_id
FROM website_pageviews
WHERE created_at < '2012-06-12'
GROUP BY
    website_session_id;
```

```
SELECT
    website_pageviews.pageview_url AS landing_page,
    COUNT(first_pageviews.website_session_id) AS sessions_hitting_this_landing_page
FROM first_pageviews
LEFT JOIN website_pageviews
    ON website_pageviews.website_pageview_id = first_pageviews.min_pageview_id
GROUP BY
    website_pageviews.pageview_url;
```

# TEST YOUR SKILLS: IDENTIFYING TOP ENTRY PAGES



## NEW MESSAGE

June 12, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Top Entry Pages**

Wow, looks like our traffic all comes in through the homepage right now!

Seems pretty obvious where we should focus on making any improvements 😊

I will likely have some follow up requests to look into performance for the homepage – stay tuned!

Thanks,  
-Morgan

Reply

Forward

## Result Grid



Filter Rows:

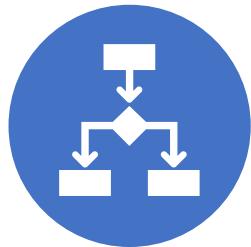
landing_page_url	sessions_hitting_page
/home	10711

## NEXT STEPS:

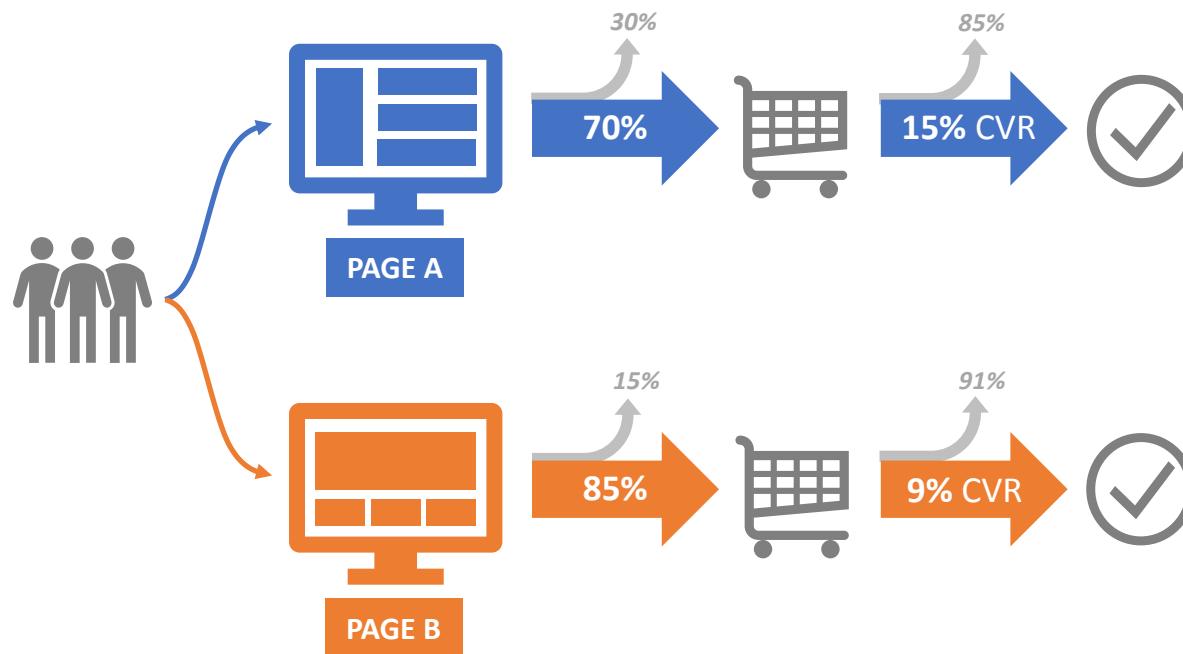
- *Analyze landing page performance, for the homepage specifically*
- *Think about whether or not the homepage is the best initial experience for all customers*

# TEST YOUR SKILLS: IDENTIFYING TOP ENTRY PAGES

# BUSINESS CONCEPT: LANDING PAGE PERFORMANCE & TESTING



Landing page analysis and testing is about understanding **the performance of your key landing pages and then testing to improve your results**



## COMMON USE CASES:

- Identifying your top opportunities for landing pages – high volume pages with higher than expected bounce rates or low conversion rates
- Setting up A/B experiments on your live traffic to see if you can improve your bounce rates and conversion rates
- Analyzing test results and making recommendations on which version of landing pages you should use going forward

# LP PERFORMANCE & TESTING

- To analyze landing page performance and compare multiple pages, we will again use temporary tables and write a multi-step “data program”
- We will find the first pageview for relevant sessions, associate that pageview with the url seen, then analyze whether that session had additional pageviews



## PRO TIP:

When your queries get longer, **make use of comments to aid readability**

```
10    customer.last_name  
11 ORDER BY
```

## MySQL QUERY IN ACTION:

```
-- Solution is a multi-step query. See video for details  
  
-- STEP 1: finding the first website_pageview_id for relevant sessions  
-- STEP 2: identifying the landing page of each session  
-- STEP 3: counting pageviews for each session, to identify "bounces"  
-- STEP 4: summarizing by counting total sessions and bounced sessions
```

## QUERY RESULTS:

Result Grid		
landing_page	sessions	bounced_sessions
/home	4093	1575
/lander-2	6500	2855
/lander-3	4232	2606



## NEW MESSAGE

June 14, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Bounce Rate Analysis**

Hi there!

The other day you showed us that **all of our traffic is landing on the homepage** right now. We should check how that landing page is performing.

Can you pull bounce rates for traffic landing on the homepage? I would like to see three numbers...**Sessions**, **Bounced Sessions**, and **% of Sessions which Bounced** (aka "Bounce Rate").

Thanks!

-Morgan

Reply

Forward

## Result Preview

### Result Grid



### Filter Rows:



sessions	bounced_sessions	bounce_rate
11044	6536	0.5918

# TEST YOUR SKILLS: CALCULATING BOUNCE RATES



## NEW MESSAGE

June 14, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Bounce Rate Analysis**

Hi there!

The other day you showed us that **all of our traffic is landing on the homepage** right now. We should check how that landing page is performing.

Can you pull bounce rates for traffic landing on the homepage? I would like to see three numbers...**Sessions**, **Bounced Sessions**, and **% of Sessions which Bounced** (aka "Bounce Rate").

Thanks!

-Morgan

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See video for details

- STEP 1: finding the first website\_pageview\_id for relevant sessions
- STEP 2: identifying the landing page of each session
- STEP 3: counting pageviews for each session, to identify "bounces"
- STEP 4: summarizing by counting total sessions and bounced sessions

TEST YOUR SKILLS: CALCULATING BOUNCE RATES



## NEW MESSAGE

June 14, 2012

From: **Morgan Rockwell (Website Manager)**  
Subject: **RE: Bounce Rate Analysis**

Ouch...almost a 60% bounce rate!

That's pretty high from my experience, especially for paid search, which should be high quality traffic.

I will put together a custom landing page for search, and set up an experiment to see if the new page does better. I will likely need your help analyzing the test once we get enough data to judge performance.

Thanks, Morgan

Reply

Forward

Result Grid			Filter Rows:
sessions		bounce_rate	
11044	6536	0.5918	

## NEXT STEPS:

- *Keep an eye on bounce rates, which represent a major area of improvement*
- *Help Morgan measure and analyze a new page that she thinks will improve performance, and analyze the results of an A/B split test against the homepage*

# TEST YOUR SKILLS: CALCULATING BOUNCE RATES



## NEW MESSAGE

July 28, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Help Analyzing LP Test**

Hi there!

Based on your bounce rate analysis, we ran a new custom landing page (**/lander-1**) in a **50/50 test against the homepage (/home)** for our gsearch nonbrand traffic.

Can you **pull bounce rates for the two groups** so we can evaluate the new page? Make sure to **just look at the time period where /lander-1 was getting traffic**, so that it is a fair comparison.

Thanks, Morgan

Reply

Forward

## Result Preview

-- finding the first instance of /lander-1 to set analysis timeframe

### Result Grid



Filter Rows:



first\_created\_at

first\_pageview\_id

-- final analysis output

### Result Grid



Filter Rows:



Ex

landing\_page

total\_sessions

bounced\_sessions

bounce\_rate

/home

/lander-1

TEST YOUR SKILLS: ANALYZING LANDING PAGE TESTS



## NEW MESSAGE

July 28, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Help Analyzing LP Test**

Hi there!

Based on your bounce rate analysis, we ran a new custom landing page (</lander-1>) in a **50/50 test against the homepage** (</home>) for our gsearch nonbrand traffic.

Can you **pull bounce rates for the two groups** so we can evaluate the new page? Make sure to **just look at the time period where /lander-1 was getting traffic**, so that it is a fair comparison.

Thanks, Morgan

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See video for details

-- STEP 0: find out when the new page /lander launched  
-- STEP 1: finding the first website\_pageview\_id for relevant sessions  
-- STEP 2: identifying the landing page of each session  
-- STEP 3: counting pageviews for each session, to identify "bounces"  
-- STEP 4: summarizing total sessions and bounced sessions, by LP

# TEST YOUR SKILLS: ANALYZING LANDING PAGE TESTS



## NEW MESSAGE

July 28, 2012

From: **Morgan Rockwell (Website Manager)**  
Subject: RE: Help Analyzing LP Test

Hey!

This is so great. It looks like **the custom lander has a lower bounce rate...success!**

I will work with Tom to get campaigns updated so that all nonbrand paid traffic is pointing to the new page.

In a few weeks, I would like you to take a look at trends to make sure things have moved in the right direction.

Thanks, Morgan

Reply

Forward

first_created_at	first_pageview_id
▶ 2012-06-19 00:35:54	23504

landing_page	total_sessions	bounced_sessions	bounce_rate
▶ /home	2261	1319	0.58337
/lander-1	2315	1232	0.53218

## NEXT STEPS:

- *Help Morgan confirm that traffic is all running to the new custom lander after campaign updates*
- *Keep an eye on bounce rates and help the team look for other areas to test and optimize*

# TEST YOUR SKILLS: ANALYZING LANDING PAGE TESTS



## NEW MESSAGE

August 31, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Landing Page Trend Analysis**

Hi there,

Could you pull the volume of **paid search nonbrand traffic landing on /home and /lander-1, trended weekly since June 1st?** I want to confirm the traffic is all routed correctly.

Could you also **pull our overall paid search bounce rate trended weekly?** I want to make sure the lander change has improved the overall picture.

Thanks!

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

Exp

week_start_date	bounce_rate	home_sessions	lander_sessions
-----------------	-------------	---------------	-----------------

▶ 2012-06-01			
2012-06-03			
2012-06-10			
2012-06-17			
2012-06-24			
2012-07-01			
2012-07-08			
2012-07-15			
2012-07-22			
2012-07-29			
2012-08-05			
2012-08-12			
2012-08-19			
2012-08-26			

TEST YOUR SKILLS: LANDING PAGE TREND ANALYSIS



## NEW MESSAGE

August 31, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Landing Page Trend Analysis**

Hi there,

Could you pull the volume of **paid search nonbrand traffic landing on /home and /lander-1, trended weekly since June 1st?** I want to confirm the traffic is all routed correctly.

Could you also **pull our overall paid search bounce rate trended weekly?** I want to make sure the lander change has improved the overall picture.

Thanks!

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See video for details

-- STEP 1: finding the first website\_pageview\_id for relevant sessions  
-- STEP 2: identifying the landing page of each session  
-- STEP 3: counting pageviews for each session, to identify "bounces"  
-- STEP 4: summarizing by week (bounce rate, sessions to each lander)

# TEST YOUR SKILLS: LANDING PAGE TREND ANALYSIS



## NEW MESSAGE

August 31, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **RE: Landing Page Trend Analysis**

This is great. Thank you!

Looks like both pages were getting traffic for a while, and then we **fully switched over to the custom lander**, as intended. And it looks like **our overall bounce rate has come down over time...nice!**

I am going to do a full deep dive into our site, and will follow up with asks.

Thanks!

-Morgan

Reply

Forward

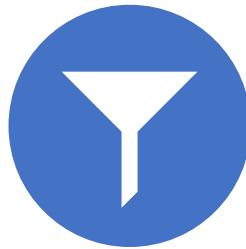
week_start_date	bounce_rate	home_sessions	lander_sessions
2012-06-01	0.60674	178	0
2012-06-03	0.58660	791	0
2012-06-10	0.61758	876	0
2012-06-17	0.55767	492	349
2012-06-24	0.58333	370	386
2012-07-01	0.58131	393	388
2012-07-08	0.56625	390	410
2012-07-15	0.54235	427	423
2012-07-22	0.51321	403	392
2012-07-29	0.49805	34	994
2012-08-05	0.53860	0	1088
2012-08-12	0.51205	0	996
2012-08-19	0.50148	0	1013
2012-08-26	0.53976	0	830

## NEXT STEPS:

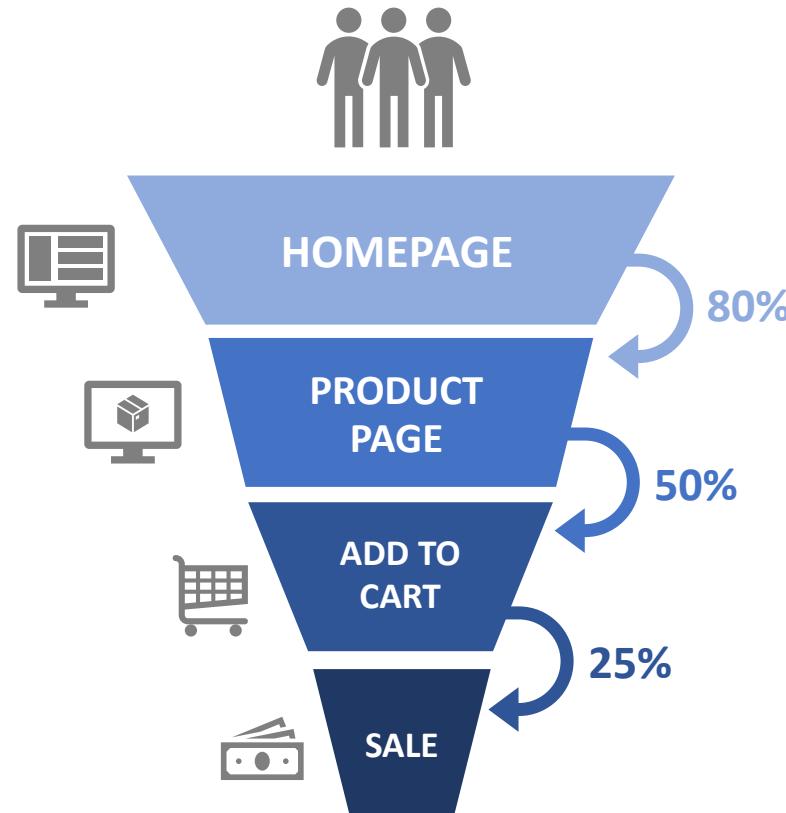
- *Enjoy the moment – your analysis of the experiment helped improve the business!*
- *Stay tuned for the next wave of analysis requests from Morgan, who is fired up about optimizing the website after her first win*

# TEST YOUR SKILLS: LANDING PAGE TREND ANALYSIS

# BUSINESS CONCEPT: ANALYZING & TESTING CONVERSION FUNNELS



Conversion funnel analysis is about **understanding and optimizing each step of your user's experience on their journey toward purchasing your products**



## COMMON USE CASES:

- Identifying the most common paths customers take before purchasing your products
- Identifying how many of your users continue on to each next step in your conversion flow, and how many users abandon at each step
- Optimizing critical pain points where users are abandoning, so that you can convert more users and sell more products

# KEY TABLE FOR CONVERSION FUNNELS: WEBSITE\_PAGEVIEWS

When we perform conversion funnel analysis, we will look at each step in our conversion flow to **see how many customers drop off** and **how many continue on** at each step

WEBSITE\_PAGEVIEWS

`SELECT * FROM website_pageviews WHERE website_session_id = 1059`

website_pageview_id	created_at	website_session_id	pageview_url
2039	2012-03-26 13:51:37	1059	/home
2040	2012-03-26 13:54:27	1059	/products
2041	2012-03-26 13:56:48	1059	/the-original-mr-fuzzy
2042	2012-03-26 14:00:14	1059	/cart
2043	2012-03-26 14:04:06	1059	/shipping
2044	2012-03-26 14:05:47	1059	/billing
2045	2012-03-26 14:13:56	1059	/thank-you-for-your-order

# USING SUBQUERIES

**SELECT \* FROM ( )**

*Allows you to query another query (similar to temporary tables)*

**SELECT columnName **FROM** ( yourSubqueryHere )**

Here is where you define which columns you want to select from the subquery within your brackets that come after your **FROM**

Instead of naming a specific table to **SELECT** from, here you will define a dataset to **SELECT FROM**

*Example:*

**SELECT \* FROM**  
**(SELECT \* FROM website\_sessions**  
**WHERE website\_session\_id <= 100) AS first\_hundred**



## HEY THIS IS IMPORTANT!

The subquery you write inside the brackets must meet two criteria to work correctly:

- 1) It must be a complete query on its own
- 2) You must give it an Alias

Missing an alias is a very common error here.

## PRO TIP:



*Subqueries and Temporary Tables perform very similar functions in SQL. Subqueries can be a little quicker to write for simple tasks, but can become very hard to follow for longer multi-step analyses*

# CONVERSION FUNNELS

- We will create temporary tables using pageview data in order to build our multi-step funnels
- We will first identify the sessions we care about, then bring in the relevant pageviews, then flag each session as having made it to certain funnel steps, and finally perform a summary analysis



## PRO TIP:

*When you have a complicated funnel, take it one step at a time!*

## MySQL QUERY IN ACTION:

```
-- Solution is a multi-step query. See video for details.  
-- STEP 1: select all pageviews for relevant sessions  
-- STEP 2: identify each pageview as the specific funnel step  
-- STEP 3: create the session-level conversion funnel view  
-- STEP 4: aggregate the data to assess funnel performance
```

## QUERY RESULTS:

Result Grid			
sessions	to_products	to_mrfuzzy	to_cart
4493	2115	1567	683

Result Grid		
lander_click_rt	products_click_rt	mrfuzzy_click_rt
0.4707	0.7409	0.4359



## NEW MESSAGE

September 05, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Help Analyzing Conversion Funnels**

Hi there!

I'd like to understand where we lose our gsearch visitors between the new /lander-1 page and placing an order. Can you **build us a full conversion funnel, analyzing how many customers make it to each step?**

Start with **/lander-1** and build the funnel all the way to our **thank you** page. Please use data since **August 5<sup>th</sup>**.

Thanks!

-Morgan

Reply

Forward

## Result Preview

Result Grid



Filter Rows:

Search

Export:



sessions | to\_products | to\_mrfuzzy | to\_cart | to\_shipping | to\_billing | to\_thankyou

▶ 4493

Result Grid



Filter Rows:

Search

Export:



lander\_click\_rt | products\_click\_rt | mrfuzzy\_click\_rt | cart\_click\_rt | shipping\_click\_rt | billing\_click\_rt

TEST YOUR SKILLS: BUILDING CONVERSION FUNNELS



## NEW MESSAGE

September 05, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **Help Analyzing Conversion Funnels**

Hi there!

I'd like to understand where we lose our gsearch visitors between the new /lander-1 page and placing an order. Can you **build us a full conversion funnel, analyzing how many customers make it to each step?**

Start with **/lander-1** and build the funnel all the way to our **thank you page**. Please use data since **August 5<sup>th</sup>**.

Thanks!

-Morgan

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See [video](#) for details.

- STEP 1: select all pageviews for relevant sessions
- STEP 2: identify each pageview as the specific funnel step
- STEP 3: create the session-level conversion funnel view
- STEP 4: aggregate the data to assess funnel performance

# TEST YOUR SKILLS: BUILDING CONVERSION FUNNELS



## NEW MESSAGE

September 05, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **RE: Help Analyzing Conversion Funnels**

This analysis is really helpful!

Looks like we should focus on the **lander**, **Mr. Fuzzy page**, and the **billing page**, which have the lowest click rates.

I have some **ideas for the billing page** that I think will make customers more comfortable entering their credit card info. I'll test a new page soon and will ask for help analyzing performance.

Thanks!

-Morgan

Reply

Forward

sessions	to_products	to_mrfuzzy	to_cart	to_shipping	to_billing	to_thankyou
4494	2116	1567	682	454	360	157

lander_click_rt	products_click_rt	mrfuzzy_click_rt	cart_click_rt	shipping_click_rt	billing_click_rt
0.4709	0.7405	0.4352	0.6657	0.7930	0.4361

## NEXT STEPS:

- *Help Morgan analyze the billing page test she plans to run*
- *Continue to look for opportunities to improve customer conversion rates*

# TEST YOUR SKILLS: BUILDING CONVERSION FUNNELS



## NEW MESSAGE

November 10, 2012

From: **Morgan Rockwell (Website Manager)**  
Subject: **Conversion Funnel Test Results**

Hello!

We **tested an updated billing page** based on your funnel analysis. Can you take a look and see whether **/billing-2** is doing any better than the original **/billing** page?

We're wondering **what % of sessions on those pages end up placing an order**. FYI – we ran this test for **all traffic, not just for our search visitors**.

Thanks!

-Morgan

Reply

Forward

## Result Preview

-- finding the first time **/billing-2** was seen

**Result Grid** Filter Rows:

first_created_at	first_pv_id
[redacted]	[redacted]

-- final test analysis output

**Result Grid** Filter Rows:  Search

billing_version_seen	sessions	orders	billing_to_order_rt
/billing	[redacted]	[redacted]	[redacted]
/billing-2	[redacted]	[redacted]	[redacted]

# TEST YOUR SKILLS: ANALYZING CONVERSION FUNNEL TESTS



## NEW MESSAGE

November 10, 2012

From: **Morgan Rockwell (Website Manager)**  
Subject: **Conversion Funnel Test Results**

Hello!

We tested an updated billing page based on your funnel analysis. Can you take a look and see whether **/billing-2** is doing any better than the original **/billing** page?

We're wondering **what % of sessions on those pages end up placing an order**. FYI – we ran this test for **all traffic, not just for our search visitors**.

Thanks!

-Morgan

Reply

Forward

## Solution Query

**SELECT**

```
MIN(website_pageviews.created_at) AS first_created_at,  
MIN(website_pageviews.website_pageview_id) AS first_pv_id  
FROM website_pageviews  
WHERE pageview_url = '/billing-2';
```

**SELECT**

```
billing_version_seen,  
COUNT(DISTINCT website_session_id) AS sessions,  
COUNT(DISTINCT order_id) AS orders,  
COUNT(DISTINCT order_id)/COUNT(DISTINCT website_session_id) AS billing_to_order_rt  
FROM(  
SELECT  
website_pageviews.website_session_id,  
website_pageviews.pageview_url AS billing_version_seen,  
orders.order_id  
FROM website_pageviews  
LEFT JOIN orders  
ON orders.website_session_id = website_pageviews.website_session_id  
WHERE website_pageviews.website_pageview_id >= 53550 -- first pageview_id where test was live  
AND website_pageviews.created_at < '2012-11-10' -- time of assignment  
AND website_pageviews.pageview_url IN ('/billing','/billing-2')  
) AS billing_sessions_w_orders  
GROUP BY  
billing_version_seen
```

TEST YOUR SKILLS: ANALYZING CONVERSION FUNNEL TESTS



## NEW MESSAGE

November 10, 2012

From: **Morgan Rockwell (Website Manager)**

Subject: **RE: Conversion Funnel Test Results**

This is so good to see!

Looks like the **new version of the billing page** is doing a much better job converting customers...yes!!

I will get Engineering to roll this out to all of our customers right away. Your insights just made us some major revenue.

Thanks so much!

-Morgan

Reply

Forward

first\_pv\_id

53550

billing_version_seen	sessions	orders	billing_to_order_rt
/billing	657	300	0.4566
/billing-2	654	410	0.6269

## NEXT STEPS:

- After Morgan gets Engineering to roll out the new version to 100% of traffic, use the data to confirm they have done so correctly
- Monitor overall sales performance to see the impact this change produces

# TEST YOUR SKILLS: ANALYZING CONVERSION FUNNEL TESTS

# INTRODUCING THE MID COURSE PROJECT

## THE **SITUATION**

Maven Fuzzy Factory has been live for ~8 months, and your CEO is due to present company performance metrics to the board next week. You'll be the one tasked with preparing relevant metrics to show the company's promising growth.

## THE **OBJECTIVE**

**Use SQL to:**

Extract and analyze website traffic and performance data from the Maven Fuzzy Factory database to quantify the company's growth, and to tell the story of how you have been able to generate that growth.

As an Analyst, the first part of your job is extracting and analyzing the data, and the next part of your job is effectively *communicating* the story to your stakeholders.

# INTRODUCING THE MID COURSE PROJECT



A screenshot of an email inbox interface. On the left, there's a profile picture of a woman with blonde hair. To her right, a blue envelope icon has a red notification bubble with the number '1' in it. Next to the icon, the text 'NEW MESSAGE' is displayed in bold capital letters. Below that, the date 'November 27, 2012' is shown. The email details are listed as follows:

**From:** Cindy Sharp (CEO)  
**Subject:** Board Meeting Next Week

The main body of the email contains the following text:

Good morning,

I need some help preparing a presentation for the board meeting next week.

The board would like to have a better understanding of our growth story over our first 8 months. This will also be a good excuse to show off our analytical capabilities a bit.

-Cindy

At the bottom of the email window, there are two buttons: 'Reply' with a left arrow icon and 'Forward' with a right arrow icon.

## YOUR OBJECTIVES:

- Tell the story of your company's growth, using trended performance data
- Use the database to explain some of the details around your growth story, and quantify the revenue impact of some of your wins
- Analyze current performance, and use that data available to assess upcoming opportunities

# MID COURSE PROJECT QUESTIONS

1

Gsearch seems to be the biggest driver of our business. Could you pull **monthly trends for gsearch sessions and orders** so that we can showcase the growth there?

~ 0:12

2

Next, it would be great to see a similar monthly trend for Gsearch, but this time **splitting out nonbrand and brand campaigns separately**. I am wondering if brand is picking up at all. If so, this is a good story to tell.

~ 3:17

3

While we're on Gsearch, could you dive into nonbrand, and pull **monthly sessions and orders split by device type**? I want to flex our analytical muscles a little and show the board we really know our traffic sources.

~ 5:32

4

I'm worried that one of our more pessimistic board members may be concerned about the large % of traffic from Gsearch. Can you pull **monthly trends for Gsearch, alongside monthly trends for each of our other channels**?

~ 7:44

# MID COURSE PROJECT QUESTIONS

5

I'd like to tell the story of our website performance improvements over the course of the first 8 months. Could you pull **session to order conversion rates, by month?**

~ 11:50

6

For the gsearch lander test, please **estimate the revenue that test earned us** (*Hint: Look at the increase in CVR from the test (Jun 19 – Jul 28), and use nonbrand sessions and revenue since then to calculate incremental value*)

~ 13:15

7

For the landing page test you analyzed previously, it would be great to show a **full conversion funnel from each of the two pages to orders**. You can use the same time period you analyzed last time (Jun 19 – Jul 28).

~ 19:57

8

I'd love for you to **quantify the impact of our billing test**, as well. Please analyze the lift generated from the test (Sep 10 – Nov 10), in terms of **revenue per billing page session**, and then pull the number of billing page sessions for the past month to understand monthly impact.

~ 25:17

# ANALYSIS FOR CHANNEL MANAGEMENT

The image shows a screenshot of the phpMyAdmin web application. In the background, there's a large watermark-like logo for "phpMyAdmin" with a gear icon. The main content area displays several database tables:

- rental**: A table with columns: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, last\_update. It contains 599 rows.
- customer**: A table with columns: first\_name, last\_name, address\_id, phone, email, password, activation\_key, last\_login, create\_date, update\_date. It contains 24 rows.
- address**: A table with columns: address\_id, address, city, district, postal\_code, country\_id. It contains 10 rows.
- category**: A table with columns: category\_id, name, description. It contains 15 rows.
- film**: A table with columns: film\_id, title, description, release\_year, language\_id, rental\_rate, length, replacement\_cost, rating, special\_features, last\_update. It contains 599 rows.
- language**: A table with columns: language\_id, name, description, last\_update. It contains 7 rows.
- staff**: A table with columns: staff\_id, first\_name, last\_name, address\_id, phone, email, hire\_date, manager\_staff\_id, last\_update. It contains 10 rows.
- inventory**: A table with columns: inventory\_id, film\_id, quantity, last\_update. It contains 599 rows.

A query results page is also visible, showing the results of a query that joins the **customer**, **rental**, and **inventory** tables to find customers who have rented the most unique movies. The results show 12 rows, ordered by the count of distinct rental IDs in descending order.

```
SELECT
    customer.first_name,
    customer.last_name,
    COUNT(DISTINCT rental_id) AS rental_count
FROM rental
LEFT JOIN customer ON rental.customer_id = customer.customer_id
GROUP BY customer.customer_id
ORDER BY COUNT(DISTINCT rental_id) DESC
```

# BUSINESS CONCEPT: CHANNEL PORTFOLIO OPTIMIZATION



Analyzing a portfolio of marketing channels is about **bidding efficiently and using data to maximize the effectiveness of your marketing budget**



## COMMON USE CASES:

- Understanding which marketing channels are driving the most sessions and orders through your website
- Understanding differences in user characteristics and conversion performance across marketing channels
- Optimizing bids and allocating marketing spend across a multi-channel portfolio to achieve maximum performance

# PAID MARKETING CAMPAIGNS: TRACKING PARAMETERS (REVIEW)

When businesses run paid marketing campaigns, they often obsess over performance and measure *everything*; how much they spend, how well traffic converts to sales, etc.

Paid traffic is commonly tagged with tracking (UTM) parameters, which are appended to URLs and allow us to tie website activity back to specific traffic sources and campaigns

www.abcwebsite.com?utm\_source=trafficSource&utm\_campaign=campaignName

The diagram illustrates the flow from a URL to a database query and its results. On the left, a URL is shown: www.abcwebsite.com?utm\_source=trafficSource&utm\_campaign=campaignName. Two blue arrows point from this URL to two separate components on the right. The first arrow points to a dark blue box labeled "WEBSITE\_SESSIONS". Inside this box, there is a SQL query:

```
SELECT DISTINCT
    utm_source,
    utm_campaign
FROM website_sessions
```

The second arrow points to a table titled "utm\_source | utm\_campaign". This table contains the following data:

utm_source	utm_campaign
HULL	HULL
bsearch	brand
bsearch	nonbrand
gsearch	brand
gsearch	nonbrand
socialbook	desktop_targeted
socialbook	pilot

# WEBSITE\_SESSIONS: ADDITIONAL SESSION DATA AVAILABLE

In addition to measuring and analyzing where traffic is coming from, we can use a business' session-level data to understand user characteristics and behaviors

**For example, we can see if the user is new to our site or if they are a repeat visitor, and which type of device they used during the session (mobile or desktop)**

WEBSITE_SESSIONS
<b>SELECT DISTINCT</b> is_repeat_session, device_type <b>FROM</b> website_sessions

is_repeat_session	device_type
0	mobile
0	desktop
1	desktop
1	mobile

# CHANNEL PORTFOLIO ANALYSIS

- To identify traffic coming from multiple marketing channels, we will use utm parameters stored in our sessions table
- We will LEFT JOIN to our orders table to understand which of the sessions converted to placing an order and generating revenue



## PRO TIP:

*Become an expert in the data stored in your session table. It is very valuable!*

## MySQL QUERY IN ACTION:

```
SELECT
    utm_content,
    COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
    COUNT(DISTINCT orders.order_id) AS orders,
    COUNT(DISTINCT orders.order_id)/COUNT(DISTINCT website_sessions.website_session_id) AS conv_rate
FROM website_sessions
LEFT JOIN orders
    ON orders.website_session_id = website_sessions.website_session_id
WHERE website_sessions.created_at BETWEEN '2014-01-01' AND '2014-02-01' -- arbitrary
GROUP BY 1
ORDER BY 2 DESC -- most orders
```

## QUERY RESULTS:

Result Grid			
utm_content	sessions	orders	conv_rate
g_ad_1	7500	543	0.0724
HULL	2724	194	0.0712
social_ad_1	1618	17	0.0105
b_ad_1	1614	109	0.0675
g_ad_2	1107	91	0.0822
b_ad_2	262	29	0.1107



## NEW MESSAGE

November 29, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Expanded Channel Portfolio**

Hi there,

With gsearch doing well and the site performing better, we launched a second paid search channel, bsearch, around August 22.

Can you pull **weekly trended session volume** since then and **compare to gsearch nonbrand** so I can get a sense for how important this will be for the business?

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

week_start_date	gsearch_sessions	bsearch_sessions
-----------------	------------------	------------------

► 2012-08-22		
2012-08-26		
2012-09-02		
2012-09-09		
2012-09-16		
2012-09-23		
2012-09-30		
2012-10-07		
2012-10-14		
2012-10-21		
2012-10-28		
2012-11-04		
2012-11-11		
2012-11-18		
2012-11-25		

TEST YOUR SKILLS: ANALYZING CHANNEL PORTFOLIOS



## NEW MESSAGE

November 29, 2012

From: Tom Parmesan (*Marketing Director*)

Subject: Expanded Channel Portfolio

Hi there,

With gsearch doing well and the site performing better, we launched a second paid search channel, bsearch, around August 22.

Can you pull weekly trended session volume since then and compare to gsearch nonbrand so I can get a sense for how important this will be for the business?

Thanks, Tom

Reply

Forward

## Solution Query

```
SELECT
    -- YEARWEEK(created_at) AS year_week,
    MIN(DATE(created_at)) AS week_start_date,
    COUNT(DISTINCT CASE WHEN utm_source = 'gsearch' THEN website_session_id ELSE NULL END) AS gsearch_sessions,
    COUNT(DISTINCT CASE WHEN utm_source = 'bsearch' THEN website_session_id ELSE NULL END) AS bsearch_sessions
FROM website_sessions
WHERE created_at > '2012-08-22' -- specified in the request
    AND created_at < '2012-11-29' -- dictated by the time of the request
    AND utm_campaign = 'nonbrand' -- limiting to nonbrand paid search
GROUP BY
    YEARWEEK(created_at)
```

TEST YOUR SKILLS: ANALYZING CHANNEL PORTFOLIOS



## NEW MESSAGE

November 29, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **RE: Expanded Channel Portfolio**

Hi there,

*This is very helpful to see. It looks like **bsearch** tends to get roughly a third the traffic of **gsearch**. This is big enough that we should really get to know the channel better.*

*I will follow up with some requests to understand channel characteristics and conversion performance.*

Thanks, Tom

Reply

Forward

week_start_date	gsearch_sessions	bsearch_sessions
2012-08-22	590	197
2012-08-26	1056	343
2012-09-02	925	290
2012-09-09	951	329
2012-09-16	1151	365
2012-09-23	1050	321
2012-09-30	999	316
2012-10-07	1002	330
2012-10-14	1257	420
2012-10-21	1302	431
2012-10-28	1211	384
2012-11-04	1350	429
2012-11-11	1246	438
2012-11-18	3508	1093
2012-11-25	2286	774

## NEXT STEPS:

- Keep an eye out for channel-specific requests from Tom
- Don't be shy about proactively slicing and dicing the data on your own to see what you can find

# TEST YOUR SKILLS: ANALYZING CHANNEL PORTFOLIOS



## NEW MESSAGE

November 30, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Comparing Our Channels**

Hi there,

I'd like to learn more about the **bsearch nonbrand** campaign. Could you please pull the **percentage of traffic coming on Mobile**, and **compare that to gsearch?**

Feel free to dig around and share anything else you find interesting. **Aggregate data since August 22nd** is great, no need to show trending at this point.

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

utm_source	sessions	mobile_sessions	pct_mobile
bsearch			
gsearch			

# TEST YOUR SKILLS: COMPARING CHANNEL CHARACTERISTICS



## NEW MESSAGE

November 30, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Comparing Our Channels**

Hi there,

I'd like to learn more about the **bsearch nonbrand** campaign. Could you please pull the **percentage of traffic coming on Mobile**, and **compare that to gsearch?**

Feel free to dig around and share anything else you find interesting. **Aggregate data since August 22nd** is great, no need to show trending at this point.

Thanks, Tom

Reply

Forward

## Solution Query

```
SELECT
    utm_source,
    COUNT(DISTINCT website_session_id) AS sessions,
    COUNT(DISTINCT CASE WHEN device_type = 'mobile' THEN website_session_id ELSE NULL END) mobile_sessions,
    COUNT(DISTINCT CASE WHEN device_type = 'mobile' THEN website_session_id ELSE NULL END) -- continued...
        /COUNT(DISTINCT website_session_id) AS pct_mobile -- continued from previous line for slide image
FROM website_sessions
WHERE created_at > '2012-08-22' -- specified in the request
    AND created_at < '2012-11-30' -- dictated by the time of the request
    AND utm_campaign = 'nonbrand' -- limiting to nonbrand paid search
GROUP BY
    utm_source
```

# TEST YOUR SKILLS: COMPARING CHANNEL CHARACTERISTICS



## NEW MESSAGE

November 30, 2012

From: **Tom Parmesan (Marketing Director)**  
Subject: **RE: Comparing Our Channels**

Wow, the desktop to mobile splits are very interesting. These channels are quite different from a device standpoint.

Let's keep this in mind as we continue to learn and optimize. Now that we know these channels are pretty different, I'm going to need your help digging in a bit more so that we can get our bids right.

Thanks, and keep up the great work!

-Tom

Reply

Forward

## Result Grid



Filter Rows:

Search

utm_source	sessions	mobile_sessions	pct_mobile
bsearch	6522	562	0.0862
gsearch	20073	4921	0.2452

## NEXT STEPS:

- *Keep an eye out for the request that Tom mentioned*
- *Try to anticipate what Tom will be asking for next; the more you can understand the types of data points P&L owners need, the stronger an analyst you become!*

# TEST YOUR SKILLS: COMPARING CHANNEL CHARACTERISTICS



## NEW MESSAGE

December 01, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Multi-Channel Bidding**

Hi there,

I'm wondering if bsearch nonbrand should have the same bids as gsearch. Could you pull **nonbrand conversion rates from session to order for gsearch and bsearch, and slice the data by device type?**

Please analyze data from **August 22 to September 18**; we ran a special pre-holiday campaign for gsearch starting on **September 19th**, so the data after that isn't fair game.

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

device_type	utm_source	sessions	orders	conv_rate
desktop	bsearch	1162		
desktop	gsearch	3011		
mobile	bsearch	130		
mobile	gsearch	1015		

# TEST YOUR SKILLS: CROSS CHANNEL BID OPTIMIZATION



## NEW MESSAGE

December 01, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Multi-Channel Bidding**

Hi there,

I'm wondering if bsearch nonbrand should have the same bids as gsearch. Could you pull **nonbrand conversion rates from session to order for gsearch and bsearch, and slice the data by device type?**

Please analyze data from **August 22 to September 18**; we ran a special pre-holiday campaign for gsearch starting on **September 19th**, so the data after that isn't fair game.

Thanks, Tom

Reply

Forward

## Solution Query

```
SELECT
    website_sessions.device_type,
    website_sessions.utm_source,
    COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
    COUNT(DISTINCT orders.order_id) AS orders,
    COUNT(DISTINCT orders.order_id)/COUNT(DISTINCT website_sessions.website_session_id) AS conv_rate
FROM website_sessions
LEFT JOIN orders
    ON orders.website_session_id = website_sessions.website_session_id
WHERE website_sessions.created_at > '2012-08-22' -- specified in the request
    AND website_sessions.created_at < '2012-09-19' -- dictated by the time of the request
    AND website_sessions.utm_campaign = 'nonbrand' -- limiting to nonbrand paid search
GROUP BY
    website_sessions.device_type,
    website_sessions.utm_source
```

TEST YOUR SKILLS: CROSS CHANNEL BID OPTIMIZATION



## NEW MESSAGE

December 01, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **RE: Multi-Channel Bidding**

Thanks, this is good to see.

As I suspected, the channels don't perform identically, so we should **differentiate our bids** in order to optimize our overall paid marketing budget.

**I'll bid down bsearch based on its under-performance.**

Great work!

-Tom

Reply

Forward

## Result Grid



Filter Rows:

Search

device_type	utm_source	sessions	orders	conv_rate
desktop	bsearch	1162	44	0.0379
desktop	gsearch	3011	136	0.0452
mobile	bsearch	130	1	0.0077
mobile	gsearch	1015	13	0.0128

## NEXT STEPS:

- Help Tom understand the impact of the bid changes he will make based on this analysis*
- Keep your eye out for new bid optimization opportunities. This is an area where a great analyst can be of tremendous value to a business!*

# TEST YOUR SKILLS: CROSS CHANNEL BID OPTIMIZATION



## NEW MESSAGE

December 22, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Impact of Bid Changes**

Hi there,

Based on your last analysis, we bid down bsearch nonbrand on **December 2nd**.

Can you pull **weekly session volume for gsearch and bsearch nonbrand, broken down by device, since November 4th?**

If you can **include a comparison metric to show bsearch as a percent of gsearch** for each device, that would be great too.

Thanks, Tom

Reply

Forward

## Result Preview

week_start_date	g_dtop_sessions	b_dtop_sessions	b_pct_of_g_dtop	g_mob_sessions	b_mob_sessions	b_pct_of_g_mob
2012-11-04						
2012-11-11						
2012-11-18						
2012-11-25						
2012-12-02						
2012-12-09						
2012-12-16						

# TEST YOUR SKILLS: CHANNEL PORTFOLIO TRENDS



## NEW MESSAGE

December 22, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **Impact of Bid Changes**

Hi there,

Based on your last analysis, we bid down bsearch nonbrand on **December 2nd**.

Can you pull **weekly session volume for gsearch and bsearch nonbrand, broken down by device, since November 4th?**

If you can include a comparison metric to show bsearch as a **percent of gsearch** for each device, that would be great too.

Thanks, Tom

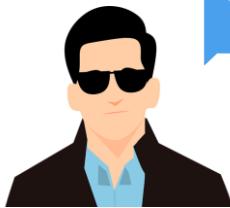
Reply

Forward

## Solution Query

```
SELECT
    -- YEARWEEK(created_at) AS year_week,
    MIN(DATE(created_at)) AS week_start_date,
    COUNT(DISTINCT CASE WHEN utm_source = 'gsearch' AND device_type = 'desktop' THEN website_session_id ELSE NULL END) AS g_dtop_sessions,
    COUNT(DISTINCT CASE WHEN utm_source = 'bsearch' AND device_type = 'desktop' THEN website_session_id ELSE NULL END) AS b_dtop_sessions,
    COUNT(DISTINCT CASE WHEN utm_source = 'bsearch' AND device_type = 'desktop' THEN website_session_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN utm_source = 'gsearch' AND device_type = 'desktop' THEN website_session_id ELSE NULL END) AS b_pct_of_g_dtop,
    COUNT(DISTINCT CASE WHEN utm_source = 'gsearch' AND device_type = 'mobile' THEN website_session_id ELSE NULL END) AS g_mob_sessions,
    COUNT(DISTINCT CASE WHEN utm_source = 'bsearch' AND device_type = 'mobile' THEN website_session_id ELSE NULL END) AS b_mob_sessions,
    COUNT(DISTINCT CASE WHEN utm_source = 'bsearch' AND device_type = 'mobile' THEN website_session_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN utm_source = 'gsearch' AND device_type = 'mobile' THEN website_session_id ELSE NULL END) AS b_pct_of_g_mob
FROM website_sessions
WHERE created_at > '2012-11-04' -- specified in the request
    AND created_at < '2012-12-22' -- dictated by the time of the request
    AND utm_campaign = 'nonbrand' -- limiting to nonbrand paid search
GROUP BY
    YEARWEEK(created_at)
```

# TEST YOUR SKILLS: CHANNEL PORTFOLIO TRENDS



## NEW MESSAGE

December 22, 2012

From: **Tom Parmesan (Marketing Director)**

Subject: **RE: Impact of Bid Changes**

Hi there,

Thanks for pulling this together!

Looks like **bsearch traffic dropped off a bit after the bid down**. Seems like gsearch was down too after Black Friday and Cyber Monday, but bsearch dropped even more.

I think this is okay given the low conversion rate.

Thanks, Tom

Reply

Forward

Result Grid



Filter Rows:

Search

Export:



week_start_date	g_dtop_sessions	b_dtop_sessions	b_pct_of_g_dtop	g_mob_sessions	b_mob_sessions	b_pct_of_g_mob
2012-11-04	1027	400	0.3895	323	29	0.0898
2012-11-11	956	401	0.4195	290	37	0.1276
2012-11-18	2655	1008	0.3797	853	85	0.0996
2012-11-25	2058	843	0.4096	692	62	0.0896
2012-12-02	1326	517	0.3899	396	31	0.0783
2012-12-09	1277	293	0.2294	424	46	0.1085
2012-12-16	1270	348	0.2740	376	41	0.1090

## NEXT STEPS:

- *Spend some time trying to fully grasp the results of this data*
- *Think about which of these metrics best controls for the seasonality Tom mentioned and isolates the impact of the bsearch bid changes*

# TEST YOUR SKILLS: CHANNEL PORTFOLIO TRENDS

# BUSINESS CONCEPT: ANALYZING DIRECT TRAFFIC



Analyzing your branded or direct traffic is about **keeping a pulse on how well your brand is doing with consumers, and how well your brand drives business**



## COMMON USE CASES:

- Identifying how much revenue you are generating from direct traffic – this is high margin revenue without a direct cost of customer acquisition
- Understanding whether or not your paid traffic is generating a “halo” effect, and promoting additional direct traffic
- Assessing the impact of various initiatives on how many customers seek out your business

# FREE TRAFFIC ANALYSIS

- To identify traffic coming to your site that you are not paying for with marketing campaigns, we will again turn to our utm params
- For non-paid traffic (i.e. organic search, direct type in), we can analyze data where the utm parameters are NULL

## PRO TIP:

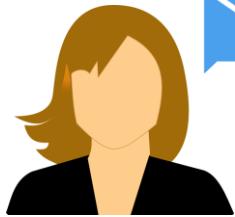
Learn to use *organic* and *direct* traffic as a health indicator for your brand

## MySQL QUERY IN ACTION:

```
SELECT
CASE
    WHEN http_referer IS NULL AND is_repeat_session = 0 THEN 'new_direct_type_in'
    WHEN http_referer IS NULL AND is_repeat_session = 1 THEN 'repeat_direct_type_in'
    WHEN http_referer IN('https://www.gsearch.com','https://www.bsearch.com') AND is_repeat_session = 0 THEN 'new_organic'
    WHEN http_referer IN('https://www.gsearch.com','https://www.bsearch.com') AND is_repeat_session = 1 THEN 'repeat_organic'
ELSE NULL END AS segment,
COUNT(DISTINCT website_session_id) AS sessions
FROM website_sessions
WHERE website_session_id BETWEEN 100000 AND 115000 -- arbitrary
    AND utm_source IS NULL -- not paid traffic
GROUP BY 1
```

## QUERY RESULTS:

segment	sessions
new_direct_type_in	367
new_organic	414
repeat_direct_type_in	688
repeat_organic	772



## NEW MESSAGE

December 23, 2012

From: Cindy Sharp (CEO)

Subject: Site traffic breakdown

Good morning,

A potential investor is asking if we're building any momentum with our brand or if we'll need to keep relying on paid traffic.

Could you pull organic search, direct type in, and paid brand search sessions by month, and show those sessions as a % of paid search nonbrand?

-Cindy

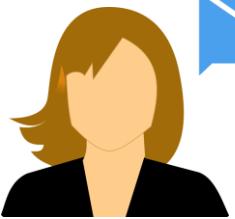
Reply

Forward

## Result Preview

yr	mo	nonbrand	brand	brand_pct_of_nonbrand	direct	direct_pct_of_nonbrand	organic	organic_pct_of_nonbrand
2012	3	1852						
2012	4	3509						
2012	5	3295						
2012	6	3439						
2012	7	3660						
2012	8	5318						
2012	9	5591						
2012	10	6883						
2012	11	12260						
2012	12	6643						

TEST YOUR SKILLS: ANALYZING FREE CHANNELS



## NEW MESSAGE

December 23, 2012

From: Cindy Sharp (CEO)

Subject: Site traffic breakdown

Good morning,

A potential investor is asking if we're building any momentum with our brand or if we'll need to keep relying on paid traffic.

Could you pull organic search, direct type in, and paid brand search sessions by month, and show those sessions as a % of paid search nonbrand?

-Cindy

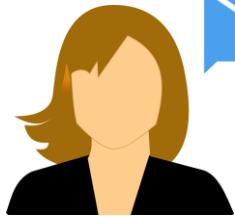
Reply

Forward

## Solution Query

```
SELECT
    YEAR(created_at) AS yr,
    MONTH(created_at) AS mo,
    COUNT(DISTINCT CASE WHEN channel_group = 'paid_nonbrand' THEN website_session_id ELSE NULL END) AS nonbrand,
    COUNT(DISTINCT CASE WHEN channel_group = 'paid_brand' THEN website_session_id ELSE NULL END) AS brand,
    COUNT(DISTINCT CASE WHEN channel_group = 'paid_brand' THEN website_session_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN channel_group = 'paid_nonbrand' THEN website_session_id ELSE NULL END) AS brand_pct_of_nonbrand,
    COUNT(DISTINCT CASE WHEN channel_group = 'direct_type_in' THEN website_session_id ELSE NULL END) AS direct,
    COUNT(DISTINCT CASE WHEN channel_group = 'direct_type_in' THEN website_session_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN channel_group = 'paid_nonbrand' THEN website_session_id ELSE NULL END) AS direct_pct_of_nonbrand,
    COUNT(DISTINCT CASE WHEN channel_group = 'organic_search' THEN website_session_id ELSE NULL END) AS organic,
    COUNT(DISTINCT CASE WHEN channel_group = 'organic_search' THEN website_session_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN channel_group = 'paid_nonbrand' THEN website_session_id ELSE NULL END) AS organic_pct_of_nonbrand
)
FROM(
SELECT
    website_session_id,
    created_at,
    CASE
        WHEN utm_source IS NULL AND http_referer IN ('https://www.gsearch.com','https://www.bsearch.com') THEN 'organic_search'
        WHEN utm_campaign = 'nonbrand' THEN 'paid_nonbrand'
        WHEN utm_campaign = 'brand' THEN 'paid_brand'
        WHEN utm_source IS NULL AND http_referer IS NULL THEN 'direct_type_in'
    END AS channel_group
FROM website_sessions
WHERE created_at < '2012-12-23'
) AS sessions_w_channel_group
GROUP BY
    YEAR(created_at),
    MONTH(created_at)
```

# TEST YOUR SKILLS: ANALYZING FREE CHANNELS



## NEW MESSAGE

December 23, 2012

From: Cindy Sharp (CEO)

Subject: RE: Site traffic breakdown

This is great to see!

Looks like not only are our brand, direct, and organic volumes growing, but **they are growing as a percentage of our paid traffic volume.**

Now this is a story I can sell to an investor!

-Cindy

Reply

Forward

yr	mo	nonbrand	brand	brand_pct_of_nonbrand	direct	direct_pct_of_nonbrand	organic	organic_pct_of_nonbrand
2012	3	1852	10	0.0054	9	0.0049	8	0.0043
2012	4	3509	76	0.0217	71	0.0202	78	0.0222
2012	5	3295	140	0.0425	151	0.0458	150	0.0455
2012	6	3439	164	0.0477	170	0.0494	190	0.0552
2012	7	3660	195	0.0533	187	0.0511	207	0.0566
2012	8	5318	264	0.0496	250	0.0470	265	0.0498
2012	9	5591	339	0.0606	285	0.0510	331	0.0592
2012	10	6883	432	0.0628	440	0.0639	428	0.0622
2012	11	12260	556	0.0454	571	0.0466	624	0.0509
2012	12	6643	464	0.0698	482	0.0726	492	0.0741

## NEXT STEPS:

- Pat yourself on the back. You just made your boss happy and helped the company win investors!*
- Spend some time thinking about what other data points you could pull that would make the company look good*

# TEST YOUR SKILLS: ANALYZING FREE CHANNELS

# BUSINESS PATTERNS & SEASONALITY

The image shows a screenshot of a MySQL database management system, likely phpMyAdmin, with a blue header bar. The main area displays several tables: rental, customer, actor, address, category, film, film\_actor, film\_text, and inventory. The rental table is currently selected, showing a list of 10 rows with columns: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update. The customer table is also visible, showing a list of 599 total rows. The SQL query results pane shows a query that counts distinct rental IDs for each customer and orders them by count in descending order. The status bar at the bottom right indicates "Copyright Maven Analytics, LLC".

rental_id	rental_date	inventory_id	customer_id	return_date	staff_id	last_update
1	2005-05-24 22:53:30	367	130	2005-05-26 22:04:30	1	2006-02-15 21:30:53
2	2005-05-24 22:54:33	1525	459	2005-05-28 19:40:33	1	2006-02-15 21:30:53
3	2005-05-24 23:00:39	1714	408	2005-06-01 22:12:46	1	2006-02-15 21:30:53
5	2005-05-24 23:05:07	24	20	2005-06-01 04:31:16	1	2006-02-15 21:30:53
6	2005-05-24 23:08:01	21	64	2005-06-01 01:13:26	1	2006-02-15 21:30:53
7	2005-05-24 23:11:12	36	30	2005-06-01 00:31:46	1	2006-02-15 21:30:53
8	2005-05-24 23:31:26	2340	258	2005-05-26 23:31:26	1	2006-02-15 21:30:53
9	2005-05-25 00:00:40	2880	128	2005-05-28 00:22:40	1	2006-02-15 21:30:53
10	2005-05-25 00:02:21	1824	399	2005-05-31 22:44:21	2	2006-02-15 21:30:53

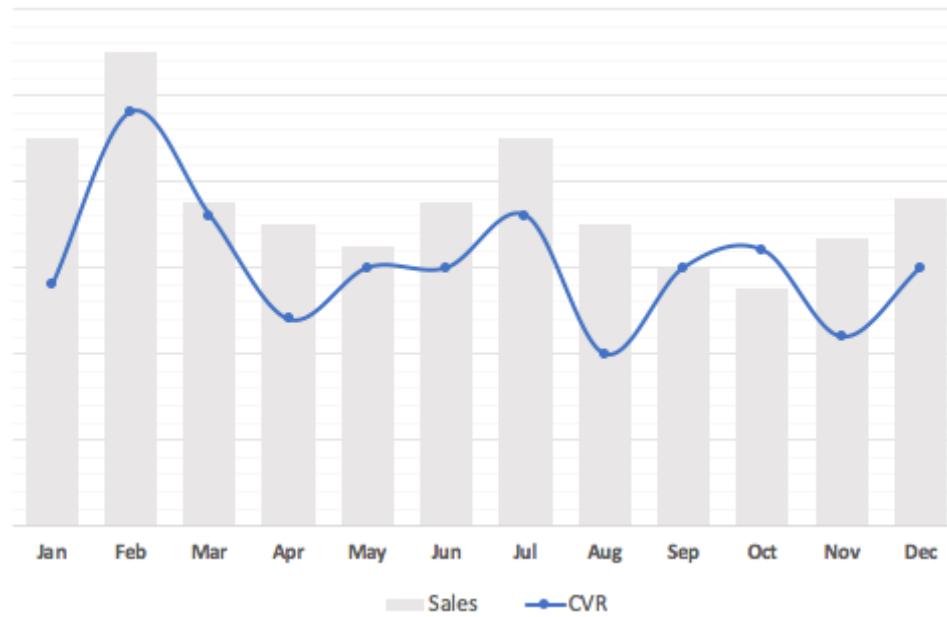
Showing rows 0 - 24 (599 total, Query took 0.0212 seconds.)

```
1 SELECT
  customer.first_name,
  customer.last_name,
  COUNT(DISTINCT rental_id) AS rental_count
  FROM rental
  LEFT JOIN customer
    ON rental.customer_id = customer.customer_id
  GROUP BY customer.first_name,
  customer.last_name
  ORDER BY
  COUNT(DISTINCT rental_id) DESC
```

# BUSINESS CONCEPT: ANALYZING SEASONALITY & BUSINESS PATTERNS



Analyzing business patterns is about **generating insights to help you maximize efficiency and anticipate future trends**



## COMMON USE CASES:

- Day-parting analysis to understand how much support staff you should have at different times of day or days of the week
- Analyzing seasonality to better prepare for upcoming spikes or slowdowns in demand

# ANALYZING SEASONALITY

- To dig into business patterns and seasonality, we will be using MySQL date functions again

Function	How You Might Use It
<b>QUARTER()</b>	Return the quarter for a given date
<b>MONTH()</b>	Return the month for a given date
<b>WEEK()</b>	Return the week for a given date
<b>DATE()</b>	Return the date for a given datetime
<b>WEEKDAY()</b>	Returns 0-6, corresponding to M-Sun
<b>HOUR()</b>	Calculate time relative to now

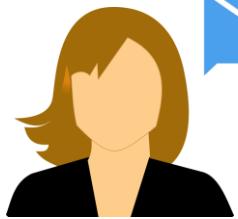
## MySQL QUERY IN ACTION:

SELECT

```
WEEK(created_at) AS wk,  
DATE(created_at) AS dt,  
WEEKDAY(created_at) AS wkday,  
HOUR(created_at) AS hr,  
COUNT(DISTINCT website_session_id) AS sessions  
FROM website_sessions  
WHERE website_session_id BETWEEN 100000 AND 115000 -- arbitrary  
GROUP BY 1,2,3,4
```

## QUERY RESULTS:

wk	dt	wkday	hr	sessions
22	2013-06-05	2	20	9
22	2013-06-05	2	21	12
22	2013-06-05	2	22	17
22	2013-06-05	2	23	11
22	2013-06-06	3	0	8
22	2013-06-06	3	1	13
22	2013-06-06	3	2	6
22	2013-06-06	3	3	4
22	2013-06-06	3	4	7
22	2013-06-06	3	5	3
22	2013-06-06	3	6	5
22	2013-06-06	3	7	9
22	2013-06-06	3	8	19



## NEW MESSAGE

January 02, 2013

From: Cindy Sharp (CEO)

Subject: Understanding Seasonality

Good morning,

2012 was a great year for us. As we continue to grow, we should **take a look at 2012's monthly and weekly volume patterns**, to see if we can find any seasonal trends we should plan for in 2013.

**If you can pull session volume and order volume**, that would be excellent.

Thanks,  
-Cindy

Reply

Forward

## Result Preview

Result Grid Filter Rows:

week\_start\_date sessions orders

### Result Grid

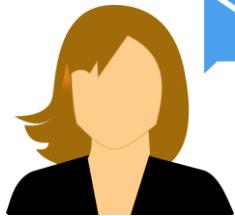


Filter Row

yr	mo	sessions	orders
2012	3		
2012	4		
2012	5		
2012	6		
2012	7		
2012	8		
2012	9		
2012	10		
2012	11		
2012	12		

week_start_date	sessions	orders
2012-03-19		
2012-03-25		
2012-04-01		
2012-04-08		
2012-04-15		
2012-04-22		
2012-04-29		
2012-05-06		
2012-05-13		
2012-05-20		
2012-05-27		
2012-06-03		
2012-06-10		
2012-06-17		
2012-06-24		
2012-07-01		
2012-07-08		
2012-07-15		
2012-07-22		
2012-07-29		
2012-08-05		
2012-08-12		
2012-08-19		
2012-08-26		
2012-09-02		
2012-09-09		
2012-09-16		
2012-09-23		
2012-09-30		
2012-10-07		
2012-10-14		
2012-10-21		
2012-10-28		
2012-11-04		
2012-11-11		
2012-11-18		
2012-11-25		
2012-12-02		
2012-12-09		
2012-12-16		
2012-12-23		
2012-12-30		

# TEST YOUR SKILLS: ANALYZING SEASONALITY



## NEW MESSAGE

January 02, 2013

From: Cindy Sharp (CEO)

Subject: Understanding Seasonality

Good morning,

2012 was a great year for us. As we continue to grow, we should **take a look at 2012's monthly and weekly volume patterns**, to see if we can find any seasonal trends we should plan for in 2013.

**If you can pull session volume and order volume**, that would be excellent.

Thanks,

-Cindy

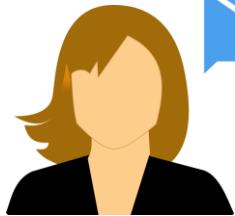
Reply

Forward

## Solution Query

```
-- Monthly trend query shown here. See Solution video for a weekly trend query
SELECT
    YEAR(website_sessions.created_at) AS yr,
    MONTH(website_sessions.created_at) AS mo,
    COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
    COUNT(DISTINCT orders.order_id) AS orders
FROM website_sessions
LEFT JOIN orders
    ON website_sessions.website_session_id = orders.website_session_id
WHERE website_sessions.created_at < '2013-01-01'
GROUP BY 1,2;
```

# TEST YOUR SKILLS: ANALYZING SEASONALITY



## NEW MESSAGE

January 02, 2013

From: Cindy Sharp (CEO)

Subject: RE: Understanding Seasonality

This is great to see.

Looks like we grew fairly steadily all year, and saw significant volume around the holiday months (especially the weeks of Black Friday and Cyber Monday).

We'll want to keep this in mind in 2013 as we think about customer support and inventory management.

Great analysis!

-Cindy

Reply

Forward

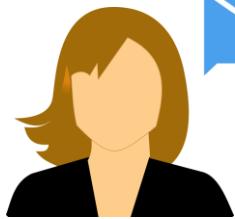
week_start_date	sessions	orders
2012-03-19	896	25
2012-03-25	983	35
2012-04-01	1193	29
2012-04-08	1029	28
2012-04-15	679	22
2012-04-22	655	18
2012-04-29	770	19
2012-05-06	798	17
2012-05-13	706	23
2012-05-20	965	28
2012-05-27	875	31
2012-06-03	920	34
2012-06-10	994	29
2012-06-17	966	37
2012-06-24	883	32
2012-07-01	892	30
2012-07-08	925	36
2012-07-15	987	47
2012-07-22	954	41
2012-07-29	1172	55
2012-08-05	1235	48
2012-08-12	1181	39
2012-08-19	1522	55
2012-08-26	1593	52
2012-09-02	1418	56
2012-09-09	1488	72
2012-09-16	1776	76
2012-09-23	1624	70
2012-09-30	1553	67
2012-10-07	1632	73
2012-10-14	1955	93
2012-10-21	2042	95
2012-10-28	1923	82
2012-11-04	2086	91
2012-11-11	1973	101
2012-11-18	5130	223
2012-11-25	4172	179
2012-12-02	2727	145
2012-12-09	2489	123
2012-12-16	2718	135
2012-12-23	1682	74
2012-12-30	309	21

yr	mo	sessions	orders
2012	3	1879	60
2012	4	3734	99
2012	5	3736	108
2012	6	3963	140
2012	7	4249	169
2012	8	6097	228
2012	9	6546	287
2012	10	8183	371
2012	11	14011	618
2012	12	10072	506

## NEXT STEPS:

- *Think about what other ways you could use your date function skills to help optimize the business*

# TEST YOUR SKILLS: ANALYZING SEASONALITY



## NEW MESSAGE

January 05, 2013

From: Cindy Sharp (CEO)

Subject: Data for Customer Service

Good morning,

We're considering adding live chat support to the website to improve our customer experience. Could you analyze the **average website session volume, by hour of day and by day week**, so that we can staff appropriately?

Let's avoid the holiday time period and use a date range of **Sep 15 - Nov 15, 2013**.

Thanks, Cindy

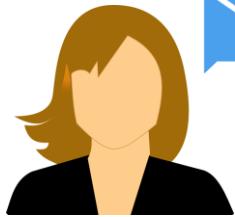
Reply

Forward

## Result Preview

hr	mon	tue	wed	thu	fri	sat	sun
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12	0.0	0.0	0.0	0.0	0.0	0.0	0.0
13	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14	0.0	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0	0.0	0.0	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0
20	0.0	0.0	0.0	0.0	0.0	0.0	0.0
21	0.0	0.0	0.0	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0

TEST YOUR SKILLS: ANALYZING BUSINESS PATTERNS



## NEW MESSAGE

January 05, 2013

From: Cindy Sharp (CEO)

Subject: Data for Customer Service

Good morning,

We're considering adding live chat support to the website to improve our customer experience. Could you analyze the **average website session volume, by hour of day and by day week**, so that we can staff appropriately?

Let's avoid the holiday time period and use a date range of **Sep 15 - Nov 15, 2013**.

Thanks, Cindy

Reply

Forward

## Solution Query

SELECT

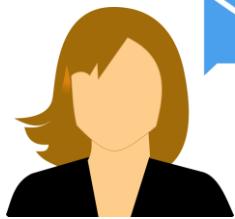
hr,  
ROUND(AVG(CASE WHEN wkday = 0 THEN website\_sessions ELSE NULL END),1) AS mon,  
ROUND(AVG(CASE WHEN wkday = 1 THEN website\_sessions ELSE NULL END),1) AS tue,  
ROUND(AVG(CASE WHEN wkday = 2 THEN website\_sessions ELSE NULL END),1) AS wed,  
ROUND(AVG(CASE WHEN wkday = 3 THEN website\_sessions ELSE NULL END),1) AS thu,  
ROUND(AVG(CASE WHEN wkday = 4 THEN website\_sessions ELSE NULL END),1) AS fri,  
ROUND(AVG(CASE WHEN wkday = 5 THEN website\_sessions ELSE NULL END),1) AS sat,  
ROUND(AVG(CASE WHEN wkday = 6 THEN website\_sessions ELSE NULL END),1) AS sun

) FROM(

SELECT

DATE(created\_at) AS created\_date,  
WEEKDAY(created\_at) AS wkday,  
HOUR(created\_at) AS hr,  
COUNT(DISTINCT website\_session\_id) AS website\_sessions  
FROM website\_sessions  
WHERE created\_at BETWEEN '2012-09-15' AND '2012-11-15' -- before holiday surge  
GROUP BY 1,2,3  
) daily\_hourly\_sessions  
GROUP BY hr

TEST YOUR SKILLS: ANALYZING BUSINESS PATTERNS



## NEW MESSAGE

January 05, 2013

From: Cindy Sharp (CEO)

Subject: RE: Data for Customer Service

Thanks, this is really helpful.

I've been speaking with support companies, and it sounds like ~10 sessions per hour per employee staffed is about right.

Looks like we can plan on one support staff around the clock and then we should double up to two staff members from 8am to 5pm Monday through Friday.

-Cindy

Reply

Forward

hr	mon	tue	wed	thu	fri	sat	sun
0	8.7	7.7	6.3	7.4	6.8	5.0	5.0
1	6.6	6.7	5.3	4.9	7.1	5.0	3.0
2	6.1	4.4	4.4	6.1	4.6	3.7	3.0
3	5.7	4.0	4.7	4.6	3.6	3.9	3.4
4	5.9	6.3	6.0	4.0	6.1	2.8	2.4
5	5.0	5.4	5.1	5.4	4.6	4.3	3.9
6	5.4	5.6	4.8	6.0	6.8	4.0	2.6
7	7.3	7.8	7.4	10.6	7.0	5.7	4.8
8	12.3	12.2	13.0	16.5	10.5	4.3	4.1
9	17.6	15.7	19.6	19.3	17.5	7.6	6.0
10	18.4	17.7	21.0	18.4	19.0	8.3	6.3
11	18.0	19.1	24.9	21.6	20.9	7.2	7.7
12	21.1	23.3	22.8	24.1	19.0	8.6	6.1
13	17.8	23.0	20.8	20.6	21.6	8.1	8.4
14	17.9	21.6	22.3	18.5	19.5	8.7	6.7
15	21.6	17.1	25.3	23.5	21.3	6.9	7.1
16	21.1	23.7	23.7	19.6	20.9	7.6	6.6
17	19.4	15.9	20.2	19.8	12.9	6.4	7.6
18	12.7	15.0	14.8	15.3	10.9	5.3	6.8
19	12.4	14.1	13.3	11.6	14.3	7.1	6.4
20	12.1	12.4	14.2	10.6	10.3	5.7	8.4
21	9.1	12.6	11.4	9.4	7.3	5.7	10.2
22	9.1	10.0	9.8	12.1	6.0	5.7	10.2
23	8.8	8.6	9.6	10.6	7.6	5.3	8.3

## NEXT STEPS:

- *Think proactively about how this type of trended analysis could be applied to other areas of the business*

TEST YOUR SKILLS: ANALYZING BUSINESS PATTERNS

# PRODUCT ANALYSIS

The screenshot displays the phpMyAdmin interface with a blue header and sidebar. The main area shows a database schema diagram with various tables like actor, address, category, country, customer, film, film\_actor, film\_category, inventory, rental, and staff. On the right, there's a table structure viewer showing columns for each table.

**Query Results:**

rental_id	rental_date	inventory_id	customer_id	return_date	staff_id	last_update
1	2005-05-24 22:53:30	367	130	2005-05-26 22:04:30	1	2006-02-15 21:30:53
2	2005-05-24 22:54:33	1525	459	2005-05-28 19:40:21	1	2006-02-15 21:30:53
3	2005-05-24 22:54:33	1525	459	2005-05-28 19:40:21	1	2006-02-15 21:30:53
4	2005-05-24 22:54:33	1525	333	2005-06-03 10:54:41	2	2006-02-15 21:30:53
5	2005-05-24 22:54:33	1525	222	2005-09-02 10:54:21	1	2006-02-15 21:30:53
6	2005-05-24 22:54:33	1525	549	2005-05-27 10:54:07	1	2006-02-15 21:30:53
7	2005-05-24 22:54:33	1525	549	2005-05-29 10:54:53	2	2006-02-15 21:30:53
8	2005-05-24 22:54:33	1525	2346	2005-05-27 10:54:46	2	2006-02-15 21:30:53
9	2005-05-25 00:00:40	2580	128	2005-05-28 00:22:40	1	2006-02-15 21:30:53
10	2005-05-25 00:02:21	1824	399	2005-05-31 22:44:21	2	2006-02-15 21:30:53

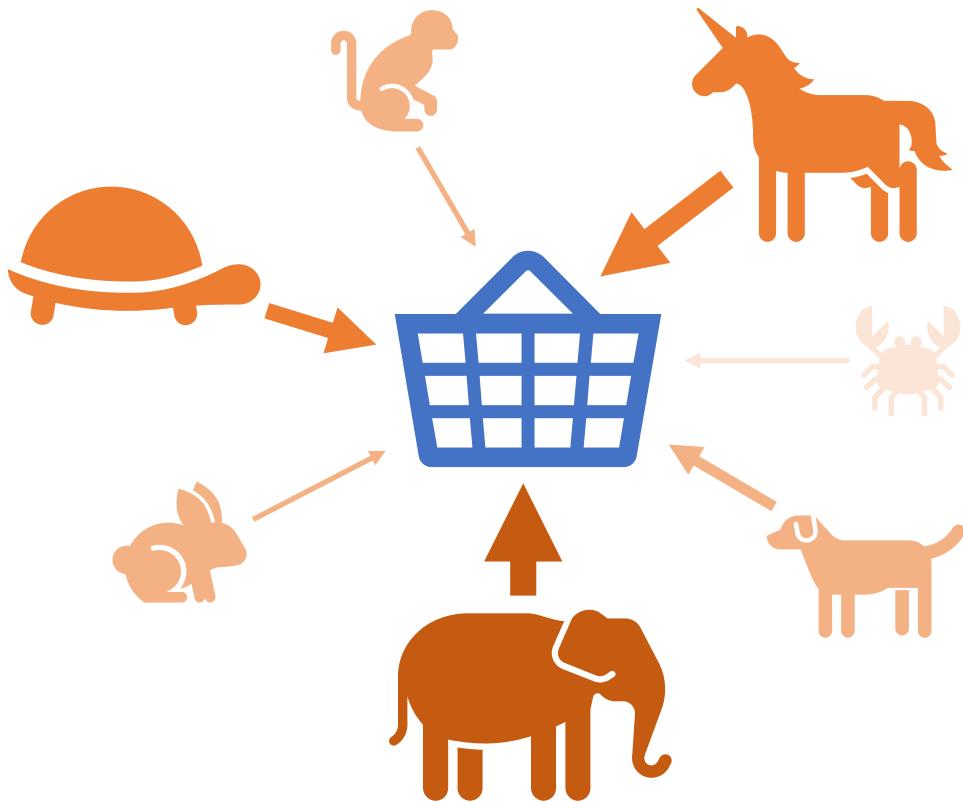
**Table Structure View:**

Table	Action	Rows	Type	Collation	Size	Overhead
actor	Browse	160	Empty	utf8_general_ci	33.0 KB	-
actor_info	Browse	160	Empty	utf8_general_ci	44.0 KB	-
address	Browse	160	Empty	utf8_general_ci	16.0 KB	-
category	Browse	160	Empty	utf8_general_ci	44.0 KB	-
city	Browse	160	Empty	utf8_general_ci	16.0 KB	-
country	Browse	160	Empty	utf8_general_ci	16.0 KB	-
customer	Browse	160	Empty	utf8_general_ci	16.0 KB	-
customer_list	Browse	160	Empty	utf8_general_ci	16.0 KB	-
film	Browse	160	Empty	utf8_general_ci	171.0 KB	-
film_actor	Browse	160	Empty	utf8_general_ci	171.0 KB	-
film_category	Browse	160	Empty	utf8_general_ci	171.0 KB	-
film_text	Browse	160	Empty	utf8_general_ci	40.0 KB	-
inventory	Browse	160	Empty	utf8_general_ci	40.0 KB	-

# BUSINESS CONCEPT: PRODUCT SALES ANALYSIS



Analyzing product sales helps you **understand how each product contributes to your business, and how product launches impact the overall portfolio**



## COMMON USE CASES:

- Analyzing sales and revenue by product
- Monitoring the impact of adding a new product to your product portfolio
- Watching product sales trends to understand the overall health of your business

# KEY BUSINESS TERMS: ORDERS, REVENUE, MARGIN, AOV

ORDERS

Number of orders placed by customers

**COUNT(order\_id)**

REVENUE

Money the business brings in from orders

**SUM(price\_usd)**

MARGIN

Revenue less the cost of good sold

**SUM(price\_usd – cogs\_usd)**

AOV

Average revenue generated per order

**AVG(price\_usd)**

SELECT

**COUNT(order\_id) AS orders,**  
**SUM(price\_usd) AS revenue,**  
**SUM(price\_usd – cogs\_usd) AS margin,**  
**AVG(price\_usd) AS average\_order\_value**

FROM orders

WHERE order\_id BETWEEN 100 AND 200



orders	revenue	margin	average_order_value
101	5048.99	3080.50	49.990000

# PRODUCT SALES ANALYSIS

- To analyze sales performance at a product level, we will look at our order data, and tie in the specific product(s) driving sales

- We will want to know how much of our order volume comes from each product, and the overall revenue and margin generated

## PRO TIP:

*Don't stop at revenue, look at margins to understand the full impact*

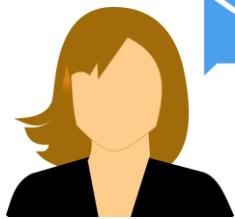
## MySQL QUERY IN ACTION:

### SELECT

```
primary_product_id,  
COUNT(order_id) AS orders,  
SUM(price_usd) AS revenue,  
SUM(price_usd - cogs_usd) AS margin,  
AVG(price_usd) AS average_order_value  
FROM orders  
WHERE order_id BETWEEN 10000 AND 11000  
GROUP BY 1  
ORDER BY 4 DESC
```

## QUERY RESULTS:

primary_product_id	orders	revenue	margin	average_order_value
1	731	42009.62	25900.00	57.468700
2	144	9118.46	5710.00	63.322639
3	126	6474.61	4386.50	51.385794



## NEW MESSAGE

January 04, 2013

From: **Cindy Sharp (CEO)**  
Subject: **Sales Trends**

Good morning,

We're about to launch a new product, and I'd like to do a deep dive on our current flagship product.

Can you please **pull monthly trends to date for number of sales, total revenue, and total margin generated** for the business?

-Cindy

Reply

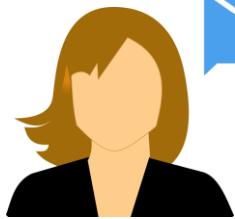
Forward

## Result Preview

Result Grid Filter Rows:  E

yr	mo	number_of_sales	total_revenue	total_margin
2012	3			
2012	4			
2012	5			
2012	6			
2012	7			
2012	8			
2012	9			
2012	10			
2012	11			
2012	12			
2013	1			

# TEST YOUR SKILLS: PRODUCT LEVEL SALES ANALYSIS



## NEW MESSAGE

January 04, 2013

From: Cindy Sharp (CEO)

Subject: Sales Trends

Good morning,

We're about to launch a new product, and I'd like to do a deep dive on our current flagship product.

Can you please pull monthly trends to date for number of sales, total revenue, and total margin generated for the business?

-Cindy

Reply

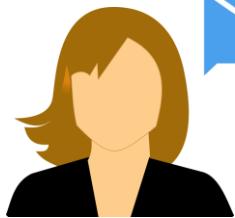
Forward

## Solution Query

**SELECT**

```
YEAR(created_at) AS yr,  
MONTH(created_at) AS mo,  
COUNT(DISTINCT order_id) AS number_of_sales,  
SUM(price_usd) AS total_revenue,  
SUM(price_usd - cogs_usd) AS total_margin  
  
FROM orders  
WHERE created_at < '2013-01-04' -- date of the request  
GROUP BY  
YEAR(created_at),  
MONTH(created_at)
```

TEST YOUR SKILLS: PRODUCT LEVEL SALES ANALYSIS



## NEW MESSAGE

January 04, 2013

From: Cindy Sharp (CEO)

Subject: RE: Sales Trends

Excellent, thank you!

This will serve as great baseline data so that we can see how our revenue and margin evolve as we roll out the new product.

It's also nice to see our growth pattern in general.

Thanks again,

-Cindy

Reply

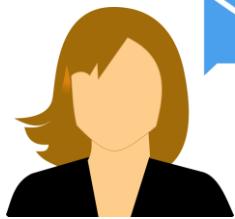
Forward

yr	mo	number_of_sales	total_revenue	total_margin
► 2012	3	60	2999.40	1830.00
2012	4	99	4949.01	3019.50
2012	5	108	5398.92	3294.00
2012	6	140	6998.60	4270.00
2012	7	169	8448.31	5154.50
2012	8	228	11397.72	6954.00
2012	9	287	14347.13	8753.50
2012	10	371	18546.29	11315.50
2012	11	618	30893.82	18849.00
2012	12	506	25294.94	15433.00
2013	1	42	2099.58	1281.00

## NEXT STEPS:

- *Keep an eye on performance as the new product launches*
- *Watch your inbox for launch-specific requests from Cindy, who seems excited about your second product*

# TEST YOUR SKILLS: PRODUCT LEVEL SALES ANALYSIS



## NEW MESSAGE

April 05, 2013

From: Cindy Sharp (CEO)

Subject: Impact of New Product Launch

Good morning,

We launched our second product back on January 6<sup>th</sup>. Can you pull together some trended analysis?

I'd like to see **monthly order volume, overall conversion rates, revenue per session**, and a **breakdown of sales by product**, all for the time period **since April 1, 2013**.

Thanks,

-Cindy

Reply

Forward

## Result Preview

Result Grid



Filter Rows:

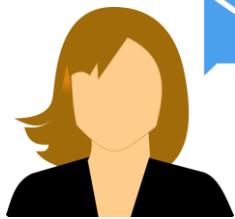
Search

Export:



yr	mo	orders	conv_rate	revenue_per_session	product_one_orders	product_two_orders
2012	4					
2012	5					
2012	6					
2012	7					
2012	8					
2012	9					
2012	10					
2012	11					
2012	12					
2013	1					
2013	2					
2013	3					
2013	4					

# TEST YOUR SKILLS: PRODUCT LAUNCH SALES ANALYSIS



## NEW MESSAGE

April 05, 2013

From: Cindy Sharp (CEO)

Subject: Impact of New Product Launch

Good morning,

We launched our second product back on January 6<sup>th</sup>. Can you pull together some trended analysis?

I'd like to see **monthly order volume, overall conversion rates, revenue per session, and a breakdown of sales by product**, all for the time period since April 1, 2013.

Thanks,  
-Cindy

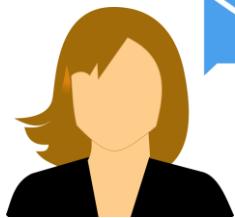
Reply

Forward

## Solution Query

```
SELECT
    YEAR(website_sessions.created_at) AS yr,
    MONTH(website_sessions.created_at) AS mo,
    -- COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
    COUNT(DISTINCT orders.order_id) AS orders,
    COUNT(DISTINCT orders.order_id)/COUNT(DISTINCT website_sessions.website_session_id) AS conv_rate,
    SUM(orders.price_usd)/COUNT(DISTINCT website_sessions.website_session_id) AS revenue_per_session,
    COUNT(DISTINCT CASE WHEN primary_product_id = 1 THEN order_id ELSE NULL END) AS product_one_orders,
    COUNT(DISTINCT CASE WHEN primary_product_id = 2 THEN order_id ELSE NULL END) AS product_two_orders
FROM website_sessions
LEFT JOIN orders
    ON website_sessions.website_session_id = orders.website_session_id
WHERE website_sessions.created_at < '2013-04-05' -- the date of the request
    AND website_sessions.created_at > '2012-04-01' -- specified in the request
GROUP BY 1,2
```

# TEST YOUR SKILLS: PRODUCT LAUNCH SALES ANALYSIS



## NEW MESSAGE

April 05, 2013

From: Cindy Sharp (CEO)

Subject: RE: Impact of New Product Launch

Thanks!

This confirms that our conversion rate and revenue per session are improving over time, which is great.

I'm having a hard time understanding if the growth since January is **due to our new product launch or just a continuation of our overall business improvements.**

I'll connect with Tom about digging into this some more.

-Cindy

Reply

Forward

yr	mo	orders	conv_rate	revenue_per_session	product_one_orders	product_two_orders
2012	4	99	0.0265	1.325391	99	0
2012	5	108	0.0289	1.445107	108	0
2012	6	140	0.0353	1.765985	140	0
2012	7	169	0.0398	1.988305	169	0
2012	8	228	0.0374	1.869398	228	0
2012	9	287	0.0438	2.191740	287	0
2012	10	371	0.0453	2.266441	371	0
2012	11	618	0.0441	2.204969	618	0
2012	12	506	0.0502	2.511412	506	0
2013	1	391	0.0611	3.127025	344	47
2013	2	497	0.0693	3.692108	335	162
2013	3	385	0.0615	3.176269	320	65
2013	4	96	0.0794	4.085227	82	14

## NEXT STEPS:

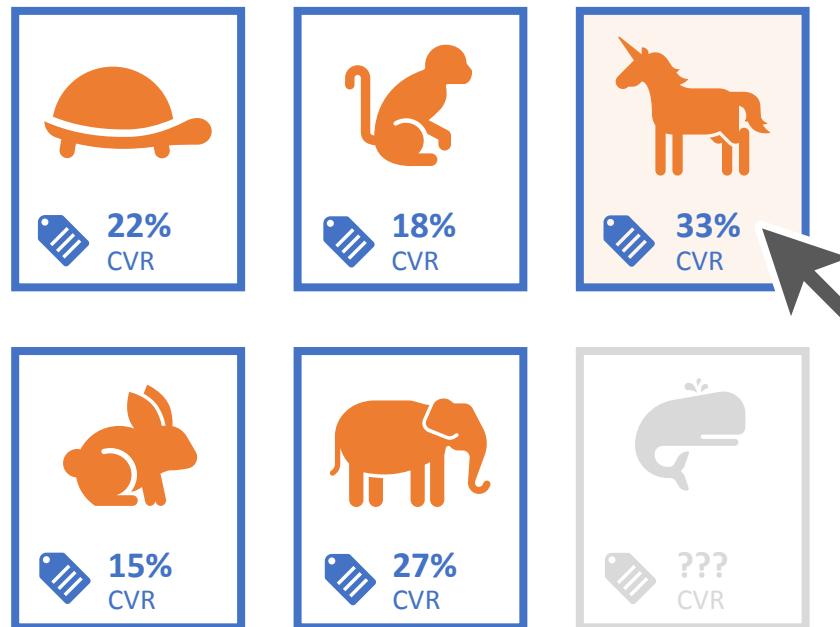
- Keep an eye out for additional launch-related requests from Cindy and Tom*
- Think proactively on your own. What do you think about the data you are seeing? Has Product 2 helped?*

# TEST YOUR SKILLS: PRODUCT LAUNCH SALES ANALYSIS

# BUSINESS CONCEPT: PRODUCT LEVEL WEBSITE ANALYSIS



Product-focused website analysis is about **learning how customers interact with each of your products, and how well each product converts customers**



## COMMON USE CASES:

- Understanding which of your products generate the most interest on multi-product showcase pages
- Analyzing the impact on website conversion rates when you add a new product
- Building product-specific conversion funnels to understand whether certain products convert better than others

# PRODUCT CONVERSION

- We'll use website\_pageviews data to identify users who viewed the /products page, and see which products they clicked next

- From specific product pages, we will look at view-to-order conversion rates, and create multi-step conversion funnels

## PRO TIP:

*Use temporary tables to break your query into manageable steps*

## MySQL QUERY IN ACTION:

```
-- Solution is a multi-step query. See video for details  
-- Step 1: find the relevant /products pageviews with website_session_id  
-- Step 2: find the next pageview id that occurs AFTER the product pageview  
-- Step 3: find the pageview_url associated with any applicable next pageview id  
-- Step 4: summarize the data and analyze the pre vs post periods
```

## QUERY RESULTS:

sessions	w_next_pg	pct_w_next_pg	to_mrFuzzy	pct_to_mrFuzzy	to_lovebear	pct_to_lovebear
26405	19547	0.7403	18001	0.6817	1546	0.0585



## NEW MESSAGE

April 06, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **Help w/ User Pathing**

Hi there!

Now that we have a new product, I'm thinking about our user path and conversion funnel. Let's look at **sessions which hit the /products page and see where they went next.**

Could you please pull **clickthrough rates from /products since the new product launch on January 6<sup>th</sup> 2013**, by product, and **compare to the 3 months leading up to launch as a baseline?**

Thanks, Morgan

Reply

Forward

## Result Preview

Result Grid	Filter Rows:	Export:
time_period	sessions w_next_pg pct_w_next_pg to_mrfuzzy pct_to_mrfuzzy to_lovebear pct_to_lovebear	
▶ A. Pre_Product_2		
B. Post_Product_2		

# TEST YOUR SKILLS: PRODUCT PATHING ANALYSIS



## NEW MESSAGE

April 06, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **Help w/ User Pathing**

Hi there!

Now that we have a new product, I'm thinking about our user path and conversion funnel. Let's look at **sessions which hit the /products page and see where they went next.**

Could you please pull clickthrough rates from /products since the new product launch on 2013-01-06, by product, and compare to the 3 months leading up to launch as a baseline?

Thanks, Morgan

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See video for details

-- Step 1: find the relevant /products pageviews with website\_session\_id  
-- Step 2: find the next pageview id that occurs AFTER the product pageview  
-- Step 3: find the pageview\_url associated with any applicable next pageview id  
-- Step 4: summarize the data and analyze the pre vs post periods

# TEST YOUR SKILLS: PRODUCT PATHING ANALYSIS



## NEW MESSAGE

April 06, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **RE: Help w/ User Pathing**

Great analysis!

Looks like the percent of /products pageviews that clicked to Mr. Fuzzy has gone down since the launch of the Love Bear, but the overall clickthrough rate has gone up, so it seems to be generating additional product interest overall.

**As a follow up, we should probably look at the conversion funnels for each product individually.**

Thanks!

-Morgan

Reply

Forward

time_period	sessions	w_next_pg	pct_w_next_pg	to_mrfuzzy	pct_to_mrfuzzy	to_lovebear	pct_to_lovebear
A. Pre_Product_2	15696	11347	0.7229	11347	0.7229	0	0.0000
B. Post_Product_2	10709	8200	0.7657	6654	0.6213	1546	0.1444

## NEXT STEPS:

- *Keep an eye out for the next request from Morgan, on product-specific conversion funnels*
- *Think proactively on your own. After seeing the data above, what do you want to know next?*

# TEST YOUR SKILLS: PRODUCT PATHING ANALYSIS



## NEW MESSAGE

April 10, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **Product Conversion Funnels**

Hi there!

I'd like to look at our two products since January 6th and analyze the **conversion funnels from each product page to conversion**.

It would be great if you could produce a **comparison between the two conversion funnels, for all website traffic**.

Thanks!

-Morgan

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

Export

product_seen	sessions	to_cart	to_shipping	to_billing	to_thankyou
--------------	----------	---------	-------------	------------	-------------

▶ lovebear					
------------	--	--	--	--	--

mrfuzzy					
---------	--	--	--	--	--

### Result Grid



Filter Rows:

Search

Export:



product_seen	product_page_click_rt	cart_click_rt	shipping_click_rt	billing_click_rt
--------------	-----------------------	---------------	-------------------	------------------

▶ lovebear				
------------	--	--	--	--

mrfuzzy				
---------	--	--	--	--

# TEST YOUR SKILLS: PRODUCT CONVERISON FUNNELS



## NEW MESSAGE

April 10, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **Product Conversion Funnels**

Hi there!

I'd like to look at our two products since January 6th and analyze the **conversion funnels from each product page to conversion**.

It would be great if you could produce a **comparison between the two conversion funnels, for all website traffic**.

Thanks!

-Morgan

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See video for details.

- STEP 1: select all pageviews for relevant sessions
- STEP 2: figure out which pageview urls to look for
- STEP 3: pull all pageviews and identify the funnel steps
- STEP 4: create the session-level conversion funnel view
- STEP 5: aggregate the data to assess funnel performance

# TEST YOUR SKILLS: PRODUCT CONVERISON FUNNELS



## NEW MESSAGE

April 10, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **RE: Product Conversion Funnels**

This is great to see!

We had found that adding a second product increased overall CTR from the /products page, and this analysis shows that the Love Bear has a better click rate to the /cart page and comparable rates throughout the rest of the funnel.

Seems like the second product was a great addition for our business. I wonder if we should add a third...

Thanks!  
-Morgan

Reply

Forward

product_seen	sessions	to_cart	to_shipping	to_billing	to_thankyou
lovebear	1599	877	603	488	301
mrfuzzy	6985	3038	2084	1710	1088

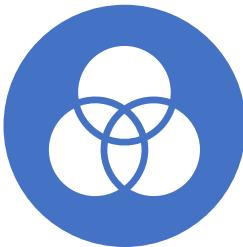
product_seen	product_page_click_rt	cart_click_rt	shipping_click_rt	billing_click_rt
lovebear	0.5485	0.6876	0.8093	0.6168
mrfuzzy	0.4349	0.6860	0.8205	0.6363

## NEXT STEPS:

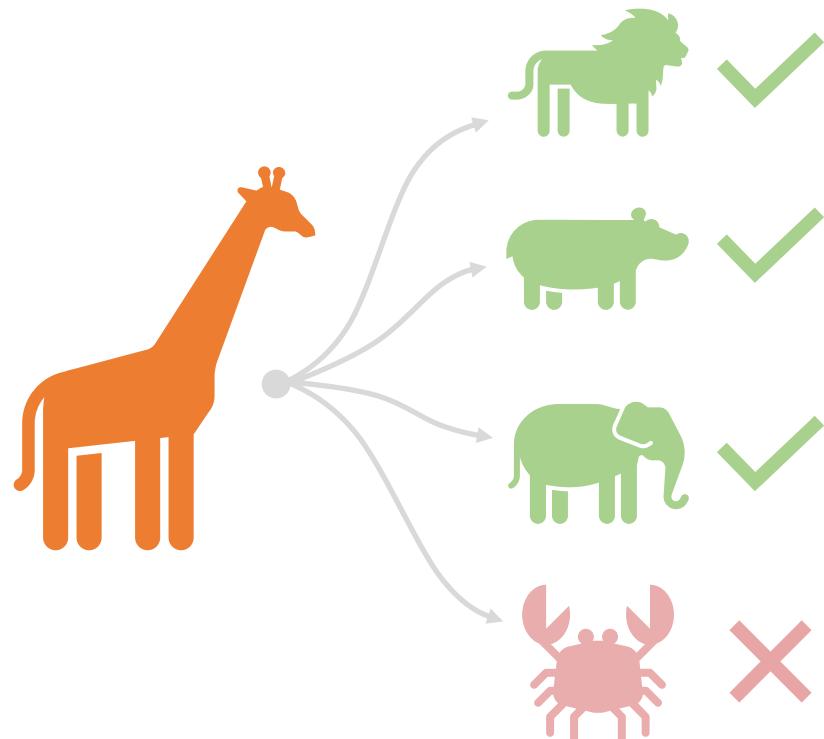
- *Make sure you fully grasp the funnel data*
- *Think proactively about what else you might want to learn, and about why you think you might be seeing the differences shown here*

# TEST YOUR SKILLS: PRODUCT CONVERISON FUNNELS

# BUSINESS CONCEPT: CROSS-SELLING PRODUCTS



Cross-sell analysis is about **understanding which products users are most likely to purchase together, and offering smart product recommendations**



## COMMON USE CASES:

- Understanding which products are often purchased together
- Testing and optimizing the way you cross-sell products on your website
- Understanding the conversion rate impact and the overall revenue impact of trying to cross-sell additional products

# CROSS-SELL ANALYSIS

- We can analyze orders and order\_items data to understand which products cross-sell, and analyze the impact on revenue

- We'll also use website\_pageviews data to understand if cross-selling hurts overall conversion rates

- Using this data, we can develop a deeper understanding of our customer purchase behaviors

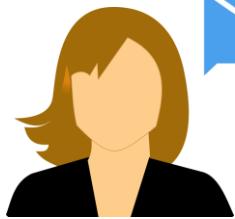
## MySQL QUERY IN ACTION:

SELECT

```
orders.primary_product_id,  
order_items.product_id AS cross_sold_product_id,  
COUNT(DISTINCT orders.order_id) AS orders  
FROM orders  
LEFT JOIN order_items  
ON orders.order_id = order_items.order_id  
AND order_items.is_primary_item = 0 -- cross sell only (exclude primary)  
WHERE orders.order_id BETWEEN 10000 AND 11000 -- arbitrary  
GROUP BY 1,2  
ORDER BY 3 DESC;
```

## QUERY RESULTS:

primary_product_id	cross_sold_product_id	orders
1	HULL	624
2	HULL	134
3	HULL	113
1	3	68
1	2	39
3	1	10
2	1	5
2	3	5
3	2	3



## NEW MESSAGE

November 22, 2013

From: Cindy Sharp (CEO)

Subject: Cross-Selling Performance

Good morning,

On September 25<sup>th</sup> we started giving customers the **option to add a 2<sup>nd</sup> product while on the /cart page**. Morgan says this has been positive, but I'd like your take on it.

Could you please **compare the month before vs the month after the change?** I'd like to see **CTR from the /cart page, Avg Products per Order, AOV, and overall revenue per /cart page view.**

Thanks, Cindy

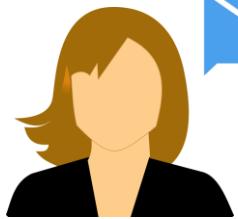
Reply

Forward

## Result Preview

Result Grid	Filter Rows:	Export:				
time_period	cart_sessions	clickthroughs	cart_ctr	products_per_order	aov	rev_per_cart_session
▶ A. Pre_Cross_Sell						
B. Post_Cross_Sell						

# TEST YOUR SKILLS: CROSS-SELL ANALYSIS



## NEW MESSAGE

November 22, 2013

From: Cindy Sharp (CEO)

Subject: Cross-Selling Performance

Good morning,

On September 25<sup>th</sup> we started giving customers the **option to add a 2<sup>nd</sup> product while on the /cart page**. Morgan says this has been positive, but I'd like your take on it.

Could you please **compare the month before vs the month after the change?** I'd like to see **CTR from the /cart page, Avg Products per Order, AOV, and overall revenue per /cart page view.**

Thanks, Cindy

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See video for details.

-- STEP 1: Identify the relevant /cart page views and their sessions

-- STEP 2: See which of those /cart sessions clicked through to the shipping page

-- STEP 3: Find the orders associated with the /cart sessions. Analyze products purchased, AOV

-- STEP 4: Aggregate and analyze a summary of our findings

# TEST YOUR SKILLS: CROSS-SELL ANALYSIS



## NEW MESSAGE

November 22, 2013

From: Cindy Sharp (CEO)

Subject: RE: Cross-Selling Performance

Thanks!

It looks like the CTR from the /cart page didn't go down (I was worried), and that our **products per order, AOV, and revenue per /cart session** are all up slightly since the cross-sell feature was added.

Doesn't look like a game changer, but the trend looks positive. Great analysis!

-Cindy

Reply

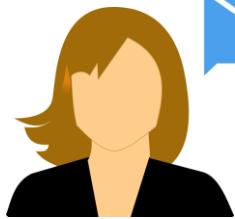
Forward

time_period	cart_sessions	clickthroughs	cart_ctr	products_per_order	aov	rev_per_cart_session
A. Pre_Cross_Sell	1830	1229	0.6716	1.0000	51.416380	18.318842
B. Post_Cross_Sell	1975	1351	0.6841	1.0447	54.251848	18.431894

## NEXT STEPS:

- Take some time to reflect on the product analyses you have done so far*
- Ask whether you think adding a second product and attempting cross sell was a good idea for the business. Do you think more products should be added?*

# TEST YOUR SKILLS: CROSS-SELL ANALYSIS



## NEW MESSAGE

January 12, 2014

From: Cindy Sharp (CEO)

Subject: Recent Product Launch

Good morning,

On December 12<sup>th</sup> 2013, we launched a third product targeting the birthday gift market (Birthday Bear).

Could you please run a **pre-post analysis comparing the month before vs. the month after**, in terms of **session-to-order conversion rate, AOV, products per order, and revenue per session?**

Thank you!

-Cindy

Reply

Forward

## Result Preview

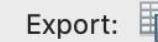
Result Grid



Filter Rows:



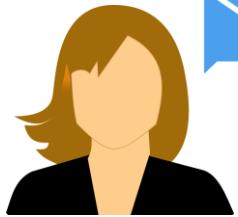
Search



Export:

time_period	conv_rate	aov	products_per_order	revenue_per_session
A. Pre_Birthday_Bear				
B. Post_Birthday_Bear				

# TEST YOUR SKILLS: PORTFOLIO EXPANSION ANALYSIS



## NEW MESSAGE

January 12, 2014

From: Cindy Sharp (CEO)

Subject: Recent Product Launch

Good morning,

On December 12<sup>th</sup> 2013, we launched a third product targeting the birthday gift market (Birthday Bear).

Could you please run a **pre-post analysis comparing the month before vs. the month after**, in terms of **session-to-order conversion rate, AOV, products per order, and revenue per session?**

Thank you!

-Cindy

Reply

Forward

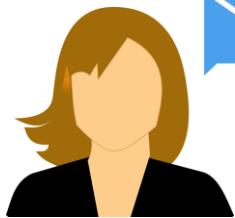
## Solution Query

```
SELECT
CASE
    WHEN website_sessions.created_at < '2013-12-12' THEN 'A. Pre_Birthday_Bear'
    WHEN website_sessions.created_at >= '2013-12-12' THEN 'B. Post_Birthday_Bear'
    ELSE 'uh oh...check logic'
END AS time_period,
COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
COUNT(DISTINCT orders.order_id) AS orders,
COUNT(DISTINCT orders.order_id)/COUNT(DISTINCT website_sessions.website_session_id) AS conv_rate,
SUM(orders.price_usd) AS total_revenue,
SUM(orders.items_purchased) AS total_products_sold,
SUM(orders.price_usd)/COUNT(DISTINCT orders.order_id) AS average_order_value,
SUM(orders.items_purchased)/COUNT(DISTINCT orders.order_id) AS products_per_order,
SUM(orders.price_usd)/COUNT(DISTINCT website_sessions.website_session_id) AS revenue_per_session

FROM website_sessions
LEFT JOIN orders
    ON orders.website_session_id = website_sessions.website_session_id

WHERE website_sessions.created_at BETWEEN '2013-11-12' AND '2014-01-12'
GROUP BY 1;
```

# TEST YOUR SKILLS: PORTFOLIO EXPANSION ANALYSIS



## NEW MESSAGE

January 12, 2014

From: Cindy Sharp (CEO)

Subject: RE: Recent Product Launch

Great – it looks like **all of our critical metrics have improved** since we launched the third product. This is fantastic!

I'm going to meet with Tom about increasing our ad spend now that we're driving more revenue per session, and we may also consider adding a fourth product.

Stay tuned,  
-Cindy

Reply

Forward

time_period	conv_rate	aov	products_per_order	revenue_per_session
A. Pre_Birthday_Bear	0.0608	54.226502	1.0464	3.298677
B. Post_Birthday_Bear	0.0702	56.931319	1.1234	3.998763

## NEXT STEPS:

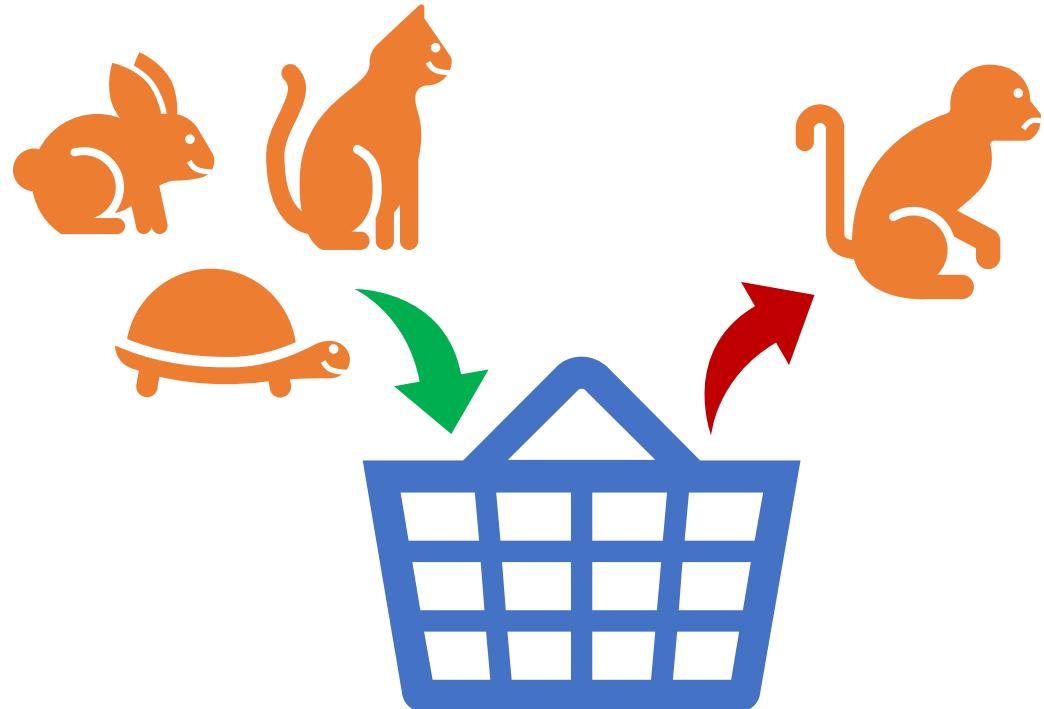
- Help keep an eye on paid traffic volume if Tom increases ad spend*
- Keep an eye out for future requests from Cindy as she gets ready to launch yet another product!*

# TEST YOUR SKILLS: PORTFOLIO EXPANSION ANALYSIS

# BUSINESS CONCEPT: PRODUCT REFUND ANALYSIS



Analyzing product refund rates is about **controlling for quality** and **understanding where you might have problems to address**



## COMMON USE CASES:

- Monitoring products from different suppliers
- Understanding refund rates for products at different price points
- Taking product refund rates and the associated costs into account when assessing the overall performance of your business

# PRODUCT REFUNDS

Keeping a eye on on refund rates is a great way to **analyze the relative quality of your products, track customer satisfaction, and keep a pulse on overall business health**

order_id	order_item_id	price_paid_usd	created_at	order_item_refund_id	refund_amount_usd	created_at
3489	3489	49.99	2013-03-03 09:51:10	NULL	NULL	NULL
27061	33000	49.99	2015-01-03 16:47:12	1505	49.99	2015-01-12 11:47:12
27061	33001	45.99	2015-01-03 16:47:12	1526	45.99	2015-01-19 13:47:12
32049	39671	49.99	2015-03-15 15:33:51	1728	49.99	2015-03-30 21:33:51
32049	39672	45.99	2015-03-15 15:33:51	NULL	NULL	NULL

# PRODUCT REFUNDS

- To analyze product refunds, we'll need to JOIN our order\_item data to the order\_item\_refunds table

- We'll want to track the total amount refunded, the % of time each product is refunded, and the overall impact on margin

## PRO TIP:

*Factoring in refunds can help identify problems AND make you look very sharp*

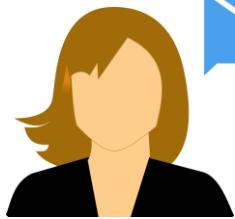
## MySQL QUERY IN ACTION:

SELECT

```
order_items.order_id,  
order_items.order_item_id,  
order_items.price_usd AS price_paid_usd,  
order_items.created_at,  
order_item_refunds.order_item_refund_id,  
order_item_refunds.refund_amount_usd,  
order_item_refunds.created_at  
FROM order_items  
LEFT JOIN order_item_refunds  
ON order_item_refunds.order_item_id = order_items.order_item_id  
WHERE order_items.order_id IN(3489,32049,27061)
```

## QUERY RESULTS:

order_id	order_item_id	price_paid_usd	created_at	order_item_refund_id	refund_amount_usd	created_at
3489	3489	49.99	2013-03-03 09:51:10	NULL	NULL	NULL
27061	33000	49.99	2015-01-03 16:47:12	1505	49.99	2015-01-12 11:47:12
27061	33001	45.99	2015-01-03 16:47:12	1526	45.99	2015-01-19 13:47:12
32049	39671	49.99	2015-03-15 15:33:51	1728	49.99	2015-03-30 21:33:51
32049	39672	45.99	2015-03-15 15:33:51	NULL	NULL	NULL



## NEW MESSAGE

October 15, 2014

From: Cindy Sharp (CEO)

Subject: Quality Issues & Refunds

Good morning,

Our Mr. Fuzzy supplier had some quality issues which weren't corrected until September 2013. Then they had a major problem where the bears' arms were falling off in Aug/Sep 2014. As a result, we replaced them with a new supplier on **September 16, 2014**.

Can you please pull **monthly product refund rates, by product, and confirm our quality issues are now fixed?**

-Cindy

Reply

Forward

## Result Preview

Result Grid



Filter Rows:

Search

Export:



yr	mo	p1_orders	p1_refund_rt	p2_orders	p2_refund_rt	p3_orders	p3_refund_rt	p4_orders	p4_refund_rt
----	----	-----------	--------------	-----------	--------------	-----------	--------------	-----------	--------------

2012	3								
2012	4								
2012	5								
2012	6								
2012	7								
2012	8								
2012	9								
2012	10								
2012	11								
2012	12								
2013	1								
2013	2								
2013	3								
2013	4								
2013	5								
2013	6								
2013	7								
2013	8								
2013	9								
2013	10								
2013	11								
2013	12								
2014	1								
2014	2								
2014	3								
2014	4								
2014	5								
2014	6								
2014	7								
2014	8								
2014	9								
2014	10								

# TEST YOUR SKILLS: PRODUCT REFUND RATES



## NEW MESSAGE

October 15, 2014

From: Cindy Sharp (CEO)

Subject: Quality Issues & Refunds

Good morning,

Our Mr. Fuzzy supplier had some quality issues which weren't corrected until September 2013. Then they had a major problem where the bears' arms were falling off in Aug/Sep 2014. As a result, we replaced them with a new supplier on **September 16, 2014**.

Can you please pull **monthly product refund rates, by product, and confirm our quality issues are now fixed?**

-Cindy

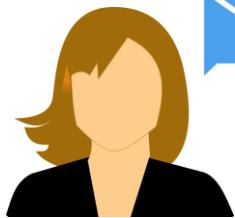
Reply

Forward

## Solution Query

```
SELECT
    YEAR(order_items.created_at) AS yr,
    MONTH(order_items.created_at) AS mo,
    COUNT(DISTINCT CASE WHEN product_id = 1 THEN order_items.order_item_id ELSE NULL END) AS p1_orders,
    COUNT(DISTINCT CASE WHEN product_id = 1 THEN order_item_refunds.order_item_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN product_id = 1 THEN order_items.order_item_id ELSE NULL END) AS p1_refund_rt,
    COUNT(DISTINCT CASE WHEN product_id = 2 THEN order_items.order_item_id ELSE NULL END) AS p2_orders,
    COUNT(DISTINCT CASE WHEN product_id = 2 THEN order_item_refunds.order_item_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN product_id = 2 THEN order_items.order_item_id ELSE NULL END) AS p2_refund_rt,
    COUNT(DISTINCT CASE WHEN product_id = 3 THEN order_items.order_item_id ELSE NULL END) AS p3_orders,
    COUNT(DISTINCT CASE WHEN product_id = 3 THEN order_item_refunds.order_item_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN product_id = 3 THEN order_items.order_item_id ELSE NULL END) AS p3_refund_rt,
    COUNT(DISTINCT CASE WHEN product_id = 4 THEN order_items.order_item_id ELSE NULL END) AS p4_orders,
    COUNT(DISTINCT CASE WHEN product_id = 4 THEN order_item_refunds.order_item_id ELSE NULL END)
        /COUNT(DISTINCT CASE WHEN product_id = 4 THEN order_items.order_item_id ELSE NULL END) AS p4_refund_rt
FROM order_items
LEFT JOIN order_item_refunds
    ON order_items.order_item_id = order_item_refunds.order_item_id
WHERE order_items.created_at < '2014-10-15'
GROUP BY 1,2
```

# TEST YOUR SKILLS: PRODUCT REFUND RATES



## NEW MESSAGE

October 15, 2014

From: Cindy Sharp (CEO)

Subject: RE: Quality Issues & Refunds

Thanks, this is helpful to see.

Looks like the refund rates for Mr. Fuzzy *did* go down after the initial improvements in September 2013, but refund rates were terrible in August and September, as expected (13-14%).

Seems like the new supplier is doing much better so far, and the other products look okay too.

-Cindy

Reply

Forward

yr	mo	p1_orders	p1_refund_rt	p2_orders	p2_refund_rt	p3_orders	p3_refund_rt	p4_orders	p4_refund_rt
2012	3	60	0.0167	0	NULL	0	NULL	0	NULL
2012	4	99	0.0505	0	NULL	0	NULL	0	NULL
2012	5	108	0.0370	0	NULL	0	NULL	0	NULL
2012	6	140	0.0571	0	NULL	0	NULL	0	NULL
2012	7	169	0.0828	0	NULL	0	NULL	0	NULL
2012	8	228	0.0746	0	NULL	0	NULL	0	NULL
2012	9	287	0.0906	0	NULL	0	NULL	0	NULL
2012	10	371	0.0728	0	NULL	0	NULL	0	NULL
2012	11	618	0.0744	0	NULL	0	NULL	0	NULL
2012	12	506	0.0593	0	NULL	0	NULL	0	NULL
2013	1	343	0.0496	47	0.0213	0	NULL	0	NULL
2013	2	336	0.0714	162	0.0123	0	NULL	0	NULL
2013	3	320	0.0563	65	0.0462	0	NULL	0	NULL
2013	4	459	0.0414	94	0.0106	0	NULL	0	NULL
2013	5	489	0.0634	82	0.0244	0	NULL	0	NULL
2013	6	503	0.0775	90	0.0556	0	NULL	0	NULL
2013	7	509	0.0727	95	0.0316	0	NULL	0	NULL
2013	8	510	0.0549	98	0.0102	0	NULL	0	NULL
2013	9	537	0.0428	98	0.0102	0	NULL	0	NULL
2013	10	603	0.0282	135	0.0148	0	NULL	0	NULL
2013	11	724	0.0345	174	0.0230	0	NULL	0	NULL
2013	12	818	0.0232	183	0.0219	139	0.0719	0	NULL
2014	1	728	0.0426	183	0.0219	200	0.0650	0	NULL
2014	2	584	0.0394	351	0.0171	211	0.0664	202	0.0099
2014	3	785	0.0306	193	0.0155	244	0.0697	205	0.0049
2014	4	917	0.0349	214	0.0187	267	0.0674	259	0.0154
2014	5	1030	0.0291	246	0.0163	299	0.0569	298	0.0067
2014	6	893	0.0571	245	0.0367	288	0.0556	249	0.0241
2014	7	961	0.0437	244	0.0369	276	0.0399	264	0.0152
2014	8	958	0.1378	237	0.0169	294	0.0680	303	0.0066
2014	9	1056	0.1326	251	0.0319	317	0.0662	327	0.0122
2014	10	513	0.0273	135	0.0074	165	0.0485	155	0.0323

## NEXT STEPS:

- *Keep an eye on product refund rates to in case there are more quality issues in the future*
- *Ask yourself how important you think product refund rates are? Is this worth ongoing analysis?*

TEST YOUR SKILLS: PRODUCT REFUND RATES

The screenshot displays the phpMyAdmin interface with the following sections:

- Left Sidebar:** Shows the database schema with tables like actor, address, category, city, country, customer, film, film\_actor, film\_category, film\_text, and inventory.
- Top Bar:** Shows "phpMyAdmin" and "Current server: phpMyAdmin demo - My".
- Query Results:** A table showing rental records from the rental table. The columns are rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update. The data includes rows for rentals on May 24, 2005, and May 25, 2005.
- Query Editor:** Displays the SQL query used to retrieve the data:

```

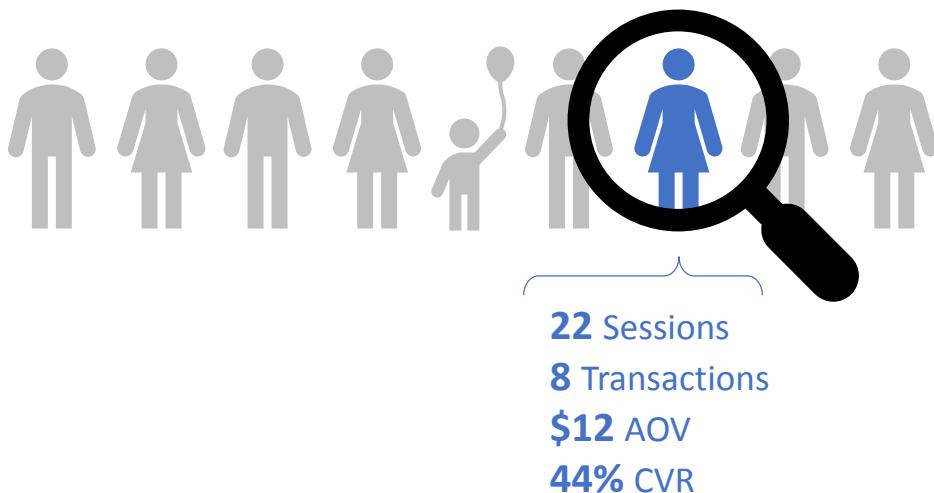
SELECT
    COUNT(DISTINCT rental_id) AS count
FROM
    rental
ORDER BY
    count DESC
  
```
- Table Structure:** A table showing the structure of the rental table with columns: rental\_id, rental\_date, inventory\_id, customer\_id, return\_date, staff\_id, and last\_update.
- Bottom Right:** A table showing the structure of the general table with columns: id, type, column, size, and overhead.

# USER ANALYSIS

# BUSINESS CONCEPT: ANALYZE REPEAT BEHAVIOR



Analyzing repeat visits helps you **understand user behavior and identify some of your most valuable customers**



## COMMON USE CASES:

- Analyzing repeat activity to see how often customers are coming back to visit your site
- Understanding which channels they use when they come back, and whether or not you are paying for them again through paid channels
- Using your repeat visit activity to build a better understanding of the value of a customer in order to better optimize marketing channels

# TRACKING REPEAT CUSTOMERS ACROSS MULTIPLE SESSIONS

Businesses track customer behavior across multiple sessions using **browser cookies**

**Cookies have unique ID values associated with them, which allows us to recognize a customer when they come back and track their behavior over time**

website_session_id	created_at	user_id	is_repeat_session	utm_source	utm_campaign	utm_content	device_type	http_referer
237966	2014-04-28 23:02:55	204524	0	gsearch	nonbrand	g_ad_1	desktop	https://www.gsearch.com
319940	2014-09-11 12:38:39	271374	0	bsearch	nonbrand	b_ad_1	desktop	https://www.bsearch.com
326645	2014-09-22 13:50:39	271374	1	gsearch	brand	g_ad_2	desktop	https://www.gsearch.com
325116	2014-09-19 11:42:44	275579	0	socialbook	desktop_targeted	social_ad_2	desktop	https://www.socialbook.com
349691	2014-10-26 19:24:17	275579	1	gsearch	brand	g_ad_2	desktop	https://www.gsearch.com
357769	2014-11-06 22:10:25	275579	1	NULL	NULL	NULL	mobile	https://www.gsearch.com
367395	2014-11-19 14:56:33	275579	1	NULL	NULL	NULL	mobile	NULL

# Comparing Dates and Datetimes with DATEDIFF()

## DATEDIFF()

Allows you to compare the time difference between two dates

### DATEDIFF( secondDate, firstDate)

This lets the SQL server know you are about to specify two dates and will want to know the time between them

This is where you prescribe dates to compare

#### Examples:

- `DATEDIFF( NOW(), born_on_date ) AS days_old`
- `DATEDIFF( second_session_created_at, first_session_created_at ) AS days_between_sessions`
- `DATEDIFF( ordered_at, ordered_at ) / 7 AS weeks_from_order_to_refund`



#### HEY THIS IS IMPORTANT!

`DATEDIFF()` subtracts the *second date* from the *first date*, so typically you will list the more recent date first.



#### PRO TIP:

MySQL's `DATEDIFF` returns a **number of days**, but you can convert to other time periods using division

# DATEDIFF()

- DATEDIFF() lets us compare the number of days between two dates
- We will use this for analyzing things like how long a user spends on a given page, how long between orders or website sessions, etc.

## PRO TIP:

Use DATEDIFF() with **GROUP BY** and aggregate functions for summary views

## MySQL QUERY IN ACTION:

SELECT

```
order_items.order_id,  
order_items.order_item_id,  
order_items.price_usd AS price_paid_usd,  
order_items.created_at,  
order_item_refunds.order_item_refund_id,  
order_item_refunds.refund_amount_usd,  
order_item_refunds.created_at,  
DATEDIFF(order_item_refunds.created_at, order_items.created_at) AS days_order_to_refund
```

FROM order\_items

LEFT JOIN order\_item\_refunds

ON order\_item\_refunds.order\_item\_id = order\_items.order\_item\_id

WHERE order\_items.order\_id IN(3489,32049,27061)

## QUERY RESULTS:

order_id	order_item_id	price_paid_usd	created_at	order_item_refund_id	refund_amount_usd	created_at	days_order_to_refund
3489	3489	49.99	2013-03-03 09:51:10	NULL	NULL	NULL	NULL
27061	33000	49.99	2015-01-03 16:47:12	1505	49.99	2015-01-12 11:47:12	9
27061	33001	45.99	2015-01-03 16:47:12	1526	45.99	2015-01-19 13:47:12	16
32049	39671	49.99	2015-03-15 15:33:51	1728	49.99	2015-03-30 21:33:51	15
32049	39672	45.99	2015-03-15 15:33:51	NULL	NULL	NULL	NULL



## NEW MESSAGE

November 01, 2014

From: **Tom Parmesan (Marketing Director)**

Subject: **Repeat Visitors**

Hey there,

We've been thinking about customer value based solely on their first session conversion and revenue. **But if customers have repeat sessions, they may be more valuable than we thought.** If that's the case, we might be able to spend a bit more to acquire them.

Could you please pull data on how many of our website visitors come back for another session? 2014 to date is good.

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter

repeat_sessions	users
0	[REDACTED]
1	[REDACTED]
2	[REDACTED]
3	[REDACTED]

0  
1  
2  
3

# TEST YOUR SKILLS: IDENTIFYING REPEAT VISITORS



## NEW MESSAGE

November 01, 2014

From: Tom Parmesan (*Marketing Director*)

Subject: Repeat Visitors

Hey there,

We've been thinking about customer value based solely on their first session conversion and revenue. **But if customers have repeat sessions, they may be more valuable than we thought.** If that's the case, we might be able to spend a bit more to acquire them.

Could you please pull data on how many of our website visitors come back for another session? 2014 to date is good.

Thanks, Tom

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See video for details.

-- STEP 1: Identify the relevant new sessions

-- STEP 2: User the user\_id values from Step 1 to find any repeat sessions those users had

-- STEP 3: Analyze the data at the user level (how many sessions did each user have?)

-- STEP 4: Aggregate the user-level analysis to generate your behavioral analysis

# TEST YOUR SKILLS: IDENTIFYING REPEAT VISITORS



## NEW MESSAGE

November 01, 2014

From: **Tom Parmesan (Marketing Director)**

Subject: **RE: Repeat Visitors**

Thanks, it's really interesting to see this breakdown.

Looks like a fair number of our customers do come back to our site after the first session.

Seems like we should learn more about this – I'll follow up with some next steps soon.

-Tom

Reply

Forward

repeat_sessions	users
0	126813
1	14086
2	315
3	4686

## NEXT STEPS:

- *Help Tom dig in to repeat customer behavior more*
- *Proactively think through the implications of having repeat visitors. What do you think?*

# TEST YOUR SKILLS: IDENTIFYING REPEAT VISITORS



## NEW MESSAGE

November 03, 2014

From: **Tom Parmesan (Marketing Director)**

Subject: **Deeper Dive on Repeat**

Ok, so the repeat session data was really interesting to see.

Now you've got me curious to better understand the behavior of these repeat customers.

Could you help me understand **the minimum, maximum, and average time between the first and second session** for customers who do come back? Again, **analyzing 2014 to date** is probably the right time period.

Thanks, Tom

Reply

Forward

## Result Preview

Result Grid



Filter Rows:

Search

Export:



avg_days_first_to_second	min_days_first_to_second	max_days_first_to_second
--------------------------	--------------------------	--------------------------

TEST YOUR SKILLS: ANALYZING REPEAT BEHAVIOR



## NEW MESSAGE

November 03, 2014

From: **Tom Parmesan (Marketing Director)**

Subject: **Deeper Dive on Repeat**

Ok, so the repeat session data was really interesting to see.

Now you've got me curious to better understand the behavior of these repeat customers.

Could you help me understand **the minimum, maximum, and average time between the first and second session** for customers who do come back? Again, **analyzing 2014 to date** is probably the right time period.

Thanks, Tom

Reply

Forward

## Solution Query

-- Solution is a multi-step query. See video for details.

```
-- STEP 1: Identify the relevant new sessions  
-- STEP 2: User the user_id values from Step 1 to find any repeat sessions those users had  
-- STEP 3: Find the created_at times for first and second sessions  
-- STEP 4: Find the differences between first and second sessions at a user level  
-- STEP 5: Aggregate the user level data to find the average, min, max
```

# TEST YOUR SKILLS: ANALYZING REPEAT BEHAVIOR



## NEW MESSAGE

November 03, 2014

From: **Tom Parmesan (Marketing Director)**  
Subject: **RE: Deeper Dive on Repeat**

Thanks!

Interesting to see that our repeat visitors are coming back about a month later, on average.

I think we should investigate the channels that these visitors are using. I'll follow up with some additional asks.

-Tom

Reply

Forward

## Result Grid



Filter Rows:



Search

Export:



avg_days_first_to_second	min_days_first_to_second	max_days_first_to_second
33.2622	1	69

## NEXT STEPS:

- *Keep an eye out for more repeat visitor analysis requests to help Tom understand behavior*
- *Think proactively, what else might you want to know?*

# TEST YOUR SKILLS: ANALYZING REPEAT BEHAVIOR



## NEW MESSAGE

November 05, 2014

From: **Tom Parmesan (Marketing Director)**

Subject: **Repeat Channel Mix**

Hi there,

Let's do a bit more digging into our repeat customers.

Can you help me understand the channels they come back through? Curious if it's all direct type-in, or if we're paying for these customers with paid search ads multiple times.

**Comparing new vs. repeat sessions by channel** would be really valuable, if you're able to pull it! 2014 to date is great.

Thanks, Tom

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

channel_group	new_sessions	repeat_sessions
► organic_search		
paid_brand		
direct_type_in		
paid_nonbrand		
paid_social		

# TEST YOUR SKILLS: NEW VS REPEAT CHANNEL PATTERNS



## NEW MESSAGE

November 05, 2014

From: **Tom Parmesan (Marketing Director)**

Subject: **Repeat Channel Mix**

Hi there,

Let's do a bit more digging into our repeat customers.

Can you help me understand the channels they come back through? Curious if it's all direct type-in, or if we're paying for these customers with paid search ads multiple times.

**Comparing new vs. repeat sessions by channel** would be really valuable, if you're able to pull it! 2014 to date is great.

Thanks, Tom

Reply

Forward

## Solution Query

```
SELECT
CASE
WHEN utm_source IS NULL AND http_referer IN ('https://www.gsearch.com', 'https://www.bsearch.com') THEN 'organic_search'
WHEN utm_campaign = 'nonbrand' THEN 'paid_nonbrand'
WHEN utm_campaign = 'brand' THEN 'paid_brand'
WHEN utm_source IS NULL AND http_referer IS NULL THEN 'direct_type_in'
WHEN utm_source = 'socialbook' THEN 'paid_social'
END AS channel_group,
COUNT(CASE WHEN is_repeat_session = 0 THEN website_session_id ELSE NULL END) AS new_sessions,
COUNT(CASE WHEN is_repeat_session = 1 THEN website_session_id ELSE NULL END) AS repeat_sessions
FROM website_sessions
WHERE created_at < '2014-11-05' -- the date of the assignment
    AND created_at >= '2014-01-01' -- prescribed date range in assignment
GROUP BY 1
ORDER BY 3 DESC
```

TEST YOUR SKILLS: NEW VS REPEAT CHANNEL PATTERNS



## NEW MESSAGE

November 05, 2014

From: **Tom Parmesan (Marketing Director)**  
Subject: RE: Repeat Channel Mix

Hi there,

So, it looks like when customers come back for repeat visits, they come mainly through **organic search, direct type-in, and paid brand**.

Only about 1/3 come through a paid channel, and brand clicks are cheaper than nonbrand. So all in all, we're not paying very much for these subsequent visits.

This make me wonder whether these convert to orders...

-Tom

Reply

Forward

channel_group	new_sessions	repeat_sessions
► organic_search	7139	11507
paid_brand	6432	11027
direct_type_in	6591	10564
paid_nonbrand	119950	0
paid_social	7652	0

## NEXT STEPS:

- *Keep an eye out for an order conversion request*
- *Take some time to think through the different channel patterns between new and repeat sessions. What do you think about this?*

# TEST YOUR SKILLS: NEW VS REPEAT CHANNEL PATTERNS



## NEW MESSAGE

November 08, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **Top Website Pages**

Hi there!

Sounds like you and Tom have learned a lot about our repeat customers. Can I trouble you for one more thing?

I'd love to do a **comparison of conversion rates and revenue per session for repeat sessions vs new sessions.**

Let's continue using data from **2014, year to date.**

Thank you!

-Morgan

Reply

Forward

## Result Preview

### Result Grid



Filter Rows:

Search

is_repeat_session	sessions	conv_rate	rev_per_session
0			
1			

# TEST YOUR SKILLS: NEW VS REPEAT PERFORMANCE



## NEW MESSAGE

November 08, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **Top Website Pages**

Hi there!

Sounds like you and Tom have learned a lot about our repeat customers. Can I trouble you for one more thing?

I'd love to do a **comparison of conversion rates and revenue per session for repeat sessions vs new sessions.**

Let's continue using data from **2014, year to date.**

Thank you!

-Morgan

Reply

Forward

## Solution Query

```
SELECT
    is_repeat_session,
    COUNT(DISTINCT website_sessions.website_session_id) AS sessions,
    COUNT(DISTINCT orders.order_id)/COUNT(DISTINCT website_sessions.website_session_id) AS conv_rate,
    SUM(price_usd)/COUNT(DISTINCT website_sessions.website_session_id) AS rev_per_session
FROM website_sessions
LEFT JOIN orders
    ON website_sessions.website_session_id = orders.website_session_id
WHERE website_sessions.created_at < '2014-11-08' -- the date of the assignment
    AND website_sessions.created_at >= '2014-01-01' -- prescribed date range in assignment
GROUP BY 1
```

# TEST YOUR SKILLS: NEW VS REPEAT PERFORMANCE



## NEW MESSAGE

November 08, 2014

From: **Morgan Rockwell (Website Manager)**

Subject: **Top Website Pages**

Hey!

This is so interesting to see. Looks like repeat sessions are **more likely to convert**, and **produce more revenue per session**.

I'll circle up with Tom on this one. Since we aren't paying much for repeat sessions, we should probably take them into account when bidding on paid traffic.

Thanks!  
-Morgan

Reply

Forward

is_repeat_session	sessions	conv_rate	rev_per_session
0	149787	0.0680	4.343754
1	33577	0.0811	5.168828

## NEXT STEPS:

- *Keep an eye out for future requests on repeat users*
- *Think proactively; If you wanted to incorporate the value of these repeat sessions into figuring out the value of your paid clicks, how might you do that?*

# TEST YOUR SKILLS: NEW VS REPEAT PERFORMANCE

# INTRODUCING THE FINAL COURSE PROJECT

## THE SITUATION

Cindy is close to securing Maven Fuzzy Factory's next round of funding, and she needs your help to tell a compelling story to investors. You'll need to pull the relevant data, and help your CEO craft a story about a data-driven company that has been producing rapid growth.

## THE OBJECTIVE

### Use SQL to:

Extract and analyze traffic and website performance data to craft a growth story that your CEO can sell. Dive in to the marketing channel activities and the website improvements that have contributed to your success to date, and use the opportunity to flex your analytical skills for the investors while you're at it.

As an Analyst, the first part of your job is extracting and analyzing the data. The next (equally important) part is communicating the story effectively to your stakeholders.

# INTRODUCING THE FINAL PROJECT

 **NEW MESSAGE**  
March, 20 2015

From: Cindy Sharp (CEO)  
Subject: Need Help w/ Fundraising

Good morning!

Now that we've been in market for 3 years, we've generated enough growth to raise a much larger round of venture capital funding. We're close to securing a large round from one of the best West Coast firms.

I need your analytical skills to help me paint a picture of high growth, and data-driven performance optimization.

Can you help?  
-Cindy

[Reply](#) [Forward](#)

## YOUR OBJECTIVES:

- Tell the story of your company's growth, using trended performance data
- Use the database to explain how you've been able to produce growth, by diving in to channels and website optimizations
- Flex your analytical muscles so the VCs know your company is a serious data-driven shop

# FINAL COURSE PROJECT QUESTIONS

- 1** First, I'd like to show our volume growth. Can you pull overall session and order volume, trended by quarter for the life of the business? Since the most recent quarter is incomplete, you can decide how to handle it.

~ 0:54
- 2** Next, let's showcase all of our efficiency improvements. I would love to show quarterly figures since we launched, for session-to-order conversion rate, revenue per order, and revenue per session.

~ 2:40
- 3** I'd like to show how we've grown specific channels. Could you pull a quarterly view of orders from Gsearch nonbrand, Bsearch nonbrand, brand search overall, organic search, and direct type-in?

~ 5:27
- 4** Next, let's show the overall session-to-order conversion rate trends for those same channels, by quarter. Please also make a note of any periods where we made major improvements or optimizations.

~ 8:45

# FINAL COURSE PROJECT QUESTIONS

5

We've come a long way since the days of selling a single product. Let's pull monthly trending for revenue and margin by product, along with total sales and revenue. Note anything you notice about seasonality.

~ 11:18

6

Let's dive deeper into the impact of introducing new products. Please pull monthly sessions to the /products page, and show how the % of those sessions clicking through another page has changed over time, along with a view of how conversion from /products to placing an order has improved.

~ 14:53

7

We made our 4<sup>th</sup> product available as a primary product on December 05, 2014 (it was previously only a cross-sell item). Could you please pull sales data since then, and show how well each product cross-sells from one another?

~ 18:40

8

In addition to telling investors about what we've already achieved, let's show them that we still have plenty of gas in the tank. Based on all the analysis you've done, could you share some recommendations and opportunities for us going forward? No right or wrong answer here – I'd just like to hear your perspective!

~ 24:04