1.

- This code outputs 10 rows from the table ”survey”

```
SELECT *
 FROM survey
 LIMIT 10;
```

Query Results

question	user_id	response
1. What are you looking for?	005e7f99-d48c-4fce-b605-10506c85aaf7	Women's Styles
2. What's your fit?	005e7f99-d48c-4fce-b605-10506c85aaf7	Medium
3. Which shapes do you like?	00a556ed-f13e-4c67-8704-27e3573684cd	Round
4. Which colors do you like?	00a556ed-f13e-4c67-8704-27e3573684cd	Two-Tone
1. What are you looking for?	00a556ed-f13e-4c67-8704-27e3573684cd	I'm not sure. Let's skip it.
2. What's your fit?	00a556ed-f13e-4c67-8704-27e3573684cd	Narrow
5. When was your last eye exam?	00a556ed-f13e-4c67-8704-27e3573684cd	<1 Year
3. Which shapes do you like?	00bf9d63-0999-43a3-9e5b-9c372e6890d2	Square
5. When was your last eye exam?	00bf9d63-0999-43a3-9e5b-9c372e6890d2	<1 Year
2. What's your fit?	00bf9d63-0999-43a3-9e5b-9c372e6890d2	Medium

2.

- This code selects the “question” column and counts unique user IDs from the table “survey”
- The output is grouped by the “question” column

```
SELECT question, COUNT(DISTINCT user_id)
FROM survey
GROUP BY question;
```

Query Results	
question	COUNT(DISTINCT user_id)
1. What are you looking for?	500
2. What's your fit?	475
3. Which shapes do you like?	380
4. Which colors do you like?	361
5. When was your last eye exam?	270

3.

- The first code does the same as the previous slide
- The second code only selects unique user IDs
- Percent Completing this Question was calculated by dividing the query results from first table by the query results in second table

```
SELECT question, COUNT(DISTINCT user_id)
FROM survey
GROUP BY question;

SELECT COUNT(DISTINCT user_id)
FROM survey;
```

Question Number	Percent Completing this Question
1	100%
2	95%
3	76%
4	72%
5	54%

4.

- This code selects 5 rows from 3 separate tables

```
SELECT *
FROM quiz
LIMIT 5;

SELECT *
FROM home_try_on
LIMIT 5;

SELECT *
FROM purchase
LIMIT 5;
```

Query Results

user_id	style	fit	shape	color
4e8118dc-bb3d-49bf-85fc-cca8d83232ac	Women's Styles	Medium	Rectangular	Tortoise
291f1cca-e507-48be-b063-002b14906468	Women's Styles	Narrow	Round	Black
75122300-0736-4087-b6d8-c0c5373a1a04	Women's Styles	Wide	Rectangular	Two-Tone
75bc6ebd-40cd-4e1d-a301-27ddd93b12e2	Women's Styles	Narrow	Square	Two-Tone
ce965c4d-7a2b-4db6-9847-601747fa7812	Women's Styles	Wide	Rectangular	Black

user_id	number_of_pairs	address
d8addd87-3217-4429-9a01-d56d68111da7	5 pairs	145 New York 9a
f52b07c8-abe4-4f4a-9d39-ba9fc9a184cc	5 pairs	383 Madison Ave
8ba0d2d5-1a31-403e-9fa5-79540f8477f9	5 pairs	287 Pell St
4e71850e-8bbf-4e6b-accc-49a7bb46c586	3 pairs	347 Madison Square N
3bc8f97f-2336-4dab-bd86-e391609dab97	5 pairs	182 Cornelia St

user_id	product_id	style	model_name	color	price
00a9dd17-36c8-430c-9d76-df49d4197dcf	8	Women's Styles	Lucy	Jet Black	150
00e15fe0-c86f-4818-9c63-3422211baa97	7	Women's Styles	Lucy	Elderflower Crystal	150
017506f7-aba1-4b9d-8b7b-f4426e71b8ca	4	Men's Styles	Dawes	Jet Black	150
0176bfb3-9c51-4b1c-b593-87edab3c54cb	10	Women's Styles	Eugene Narrow	Rosewood Tortoise	95
01fdf106-f73c-4d3f-a036-2f3e2ab1ce06	8	Women's Styles	Lucy	Jet Black	150

5.

- This code matches the user_id row in the quiz table with the user_id row in the home_try_on table
- Furthermroe, it matches the user_id rows in the purchase and quiz tables

```
SELECT DISTINCT quiz.user_id, home_try_on.user_id IS
NOT NULL AS 'is_home_try_on',
home_try_on.number_of_pairs, purchase.user_id IS NOT
NULL AS 'is_purchase'
FROM quiz
LEFT JOIN home_try_on
          ON        quiz.user_id =
home_try_on.user_id
LEFT JOIN purchase
          ON purchase.user_id = quiz.user_id
LIMIT 10;
```

Query Results

user_id	is_home_try_on	number_of_pairs	is_purchase
4e8118dc-bb3d-49bf-85fc-cca8d83232ac	1	3 pairs	0
291f1cca-e507-48be-b063-002b14906468	1	3 pairs	1
75122300-0736-4087-b6d8-c0c5373a1a04	0	Ø	0
75bc6ebd-40cd-4e1d-a301-27ddd93b12e2	1	5 pairs	0
ce965c4d-7a2b-4db6-9847-601747fa7812	1	3 pairs	1
28867d12-27a6-4e6a-a5fb-8bb5440117ae	1	5 pairs	1
5a7a7e13-fbcf-46e4-9093-79799649d6c5	0	Ø	0
0143cb8b-bb81-4916-9750-ce956c9f9bd9	0	Ø	0
a4ccc1b3-cbb6-449c-b7a5-03af42c97433	1	5 pairs	0
b1dded76-cd60-4222-82cb-f6d464104298	1	3 pairs	0

6.

- This code sums the total users that browsed the website, the total users that tried the glasses on, and the total users that made a purchase
- The code then calculates ratios of total glasses tried on to the total number of users and total purchases to total glasses tried on

```
WITH funnel AS(
SELECT DISTINCT quiz.user_id, home_try_on.user_id IS
NOT NULL AS 'is_home_try_on',
home_try_on.number_of_pairs, purchase.user_id IS NOT
NULL AS 'is_purchase'
FROM quiz
LEFT JOIN home_try_on
          ON        quiz.user_id =
home_try_on.user_id
LEFT JOIN purchase
          ON purchase.user_id = quiz.user_id
)

SELECT COUNT(*) AS 'total_browse',
SUM(is_home_try_on) AS 'total_try_on',
SUM(is_purchase) AS 'purchase', 1.0 *
SUM(is_home_try_on) / COUNT(user_id) AS
'total_try_on:total_user', 1.0 * SUM(is_purchase) /
SUM(is_home_try_on) AS 'total_purchase:total_try_on'
FROM funnel;
```

Query Results

total_browse	total_try_on	purchase	total_try_on:total_user	total_purchase:total_try_on
1000	750	495	0.75	0.66