



# Usage Funnels with Warby Parker

Analyze Data with SQL  
Àngels Albiol Masip  
15/12/2021

# Table of Contents

1. STYLE QUIZ
2. QUIZ FUNNEL
3. HOME TRY-ON FUNNEL
4. ANALYZING THE DATA

# 1. STYLE QUIZ

# 1. Style Quiz

1. "What are you looking for?"
2. "What's your fit?"
3. "Which shapes do you like?"
4. "Which colors do you like?"
5. "When was your last eye exam?"

The users' responses are stored in a table called **survey**.

column1	column2	column3
QUESTION	USER_ID	RESPONSE

## 2. QUIZ FUNNEL

## 2. QUIZZ FUNNEL

- **What is the number of responses for each question?**

- Question 1: 500
- Question 2: 475
- Question 3: 380
- Question 4: 361
- Question 5: 270

- **Which question(s) of the quiz have a lower completion rates?**

Question 5, with 74.79%, have the lower completion rates. Also, question 3, with 80%, has a low completion rate.

- **What do you think is the reason?**

There may be different reasons why people are reluctant to answer question 5. Either they do not like to answer questions about medical exams, or they just simply cannot remember when it was.

question	users	percentage
1. What are you looking for?	500	100%
2. What's your fit?	475	95%
3. Which shapes do you like?	380	80%
4. Which colors do you like?	361	95%
5. When was your last eye exam?	270	74.79%

```
SELECT question,  
       COUNT(DISTINCT user_id) AS 'users'  
FROM survey  
GROUP BY 1;
```

# 3. HOME TRY-ON FUNNEL

# 1. Home try-on funnel

- QUIZ

column1	column2	column3	column4	column5
user_id	style	fit	shape	color

- HOME TRY-ON

column1	column2	column3
user_id	number_of_pairs	address

- PURCHASE

column1	column2	column3	column4	column5	column6
user_id	product_id	style	model_name	color	price



## 2. QUIZZ FUNNEL

```
SELECT DISTINCT q.user_id,  
               h.user_id IS NOT NULL AS 'is_home_try_on',  
               h.number_of_pairs,  
               p.user_id IS NOT NULL AS 'is_purchase'  
FROM quiz AS 'q'  
LEFT JOIN home_try_on AS 'h'  
      ON q.user_id = h.user_id  
LEFT JOIN purchase AS 'p'  
      ON q.user_id = p.user_id  
LIMIT 5;
```

user_id	is_home_try_on	number_of_pairs	is_purchase
4e8118dc-bb3d-49bf-85fc-cca8d83232ac	1	3 pairs	0
291f1cca-e507-48be-b063-02b14906468	1	3 pairs	1
75122300-0736-4087-b6d8-c0c5373a1a04	0	∅	0
75bc6ebd-40cd-4e1d-a301-27ddd93b12e2	1	5 pairs	0
ce965c4d-7a2b-4db6-9847-601747fa7812	1	3 pairs	1

## 4. ANALYZING THE DATA

## 4. ANALYZING THE DATA

**Compare conversion from quiz→home\_try\_on  
and home\_try\_on→purchase.**

```
WITH q AS(
  SELECT '1-quiz' AS 'stage',
  COUNT(DISTINCT user_id) AS
  'distinct_users'
  FROM quiz),
h AS(
  SELECT '2-home_try_on' AS 'stage',
  COUNT(DISTINCT user_id) AS
  'distinct_users'
  FROM home_try_on),
p AS(
  SELECT '3-purchase' AS 'stage',
  COUNT(DISTINCT user_id) AS
  'distinct_users'
  FROM purchase)
SELECT *
FROM q
UNION
SELECT *
FROM h
UNION
SELECT *
FROM p;
```

stage	distinct_users	percentage
1-quiz	1000	100%
2-home_try_on	750	75%
3-purchase	495	66%

## 4. ANALYZING THE DATA

**Calculate the difference in purchase rates between customers who had 3 number\_of\_pairs with ones who had 5.**

num_of_pairs	num_users	num_purchase	purchase_rate
3 pairs	379	201	53%
5 pairs	371	294	79%

```
SELECT h.number_of_pairs,  
       COUNT(h.user_id) AS 'num_users',  
       COUNT(p.user_id) AS 'num_purchase'  
FROM home_try_on AS 'h'  
LEFT JOIN purchase AS 'p'  
  ON h.user_id = p.user_id  
GROUP BY 1;
```

As we can see on the table, users who received 5 pairs to try on at home, have a higher purchase rate. Probably, because they had more options to choose from and find a better fit.