—First I’m going to query a sample of all the tables and look for my primary and forgin keys

```
SELECT *
FROM browse
LIMIT 5;
SELECT *
FROM checkout
LIMIT 5;
SELECT *
FROM purchase
LIMIT 2;
```

–This returns the data for all three tables like the one below:

user_id	browse_date	item_id
336f9fdc-aaeb-48a1-a773-e3a935442d45	2017-12-20	3
336f9fdc-aaeb-48a1-a773-e3a935442d45	2017-12-20	22

–Next I’m going to join these tables using aliases for simplicity and limit the view to 50

```
select distinct b.browse_date,
b.user_id,
c.user_id IS NOT NULL AS 'is_checkout',
p.user_id IS NOT NULL AS 'is_purchase'
from browse as b
left join checkout as c
on b.user_id = c.user_id
left join purchase as p
on c.user_id = p.user_id
limit 3;
```

browse_date	user_id	is_checkout	is_purchase
2017-12-20	336f9fdc-aaeb-48a1-a773-e3a935442d45	0	0
2017-12-20	4596bb1a-7aa9-4ac9-9896-022d871cdcde	0	0
2017-12-20	2fdb3958-ffc9-4b84-a49d-5f9f40e9469e	1	1

–Next I will use a with clause to add aggregates to calculate the number of browsers, checkouts, and purchases as well as the percentages of browsers to checkout and checkout to purchases.

```
WITH funnels AS (
 SELECT DISTINCT b.browse_date,
    b.user_id,
    c.user_id IS NOT NULL AS 'is_checkout',
    p.user_id IS NOT NULL AS 'is_purchase'
 FROM browse AS 'b'
 LEFT JOIN checkout AS 'c'
   ON c.user_id = b.user_id
 LEFT JOIN purchase AS 'p'
   ON p.user_id = c.user_id)
SELECT COUNT(*) as 'num_browse',
sum(is_checkout) as 'num_checkout',
sum(is_purchase) as 'num_purchase',
1.0 * SUM(is_checkout) / COUNT(user_id) as 'browse_to_checkout',
1.0 * SUM(is_purchase) / SUM(is_checkout) as 'checkout_to_purchase'
FROM funnels;
```

num_browse	num_checkout	num_purchase	browse_to_checkout	checkout_to_purchase
775	183	163	0.236129032258065	0.890710382513661

–Finally we can select browse_date and group and order by that to get a more in depth look at the daily level.

```
WITH funnels AS (
 SELECT DISTINCT b.browse_date,
    b.user_id,
    c.user_id IS NOT NULL AS 'is_checkout',
    p.user_id IS NOT NULL AS 'is_purchase'
 FROM browse AS 'b'
 LEFT JOIN checkout AS 'c'
   ON c.user_id = b.user_id
 LEFT JOIN purchase AS 'p'
   ON p.user_id = c.user_id)
SELECT DISTINCT browse_date,
COUNT(*) AS 'num_browse',
  SUM(is_checkout) AS 'num_checkout',
  SUM(is_purchase) AS 'num_purchase',
  1.0 * SUM(is_checkout) / COUNT(user_id) AS 'browse_to_checkout',
  1.0 * SUM(is_purchase) / SUM(is_checkout) AS 'checkout_to_purchase'
FROM funnels
GROUP BY browse_date
ORDER BY browse_date;
```

–Our query returned the following view, from this we can see that our conversions went from 80% on 12-20 to 94% on 12-23

browse_date	num_browse	num_checkout	num_purchase	browse_to_checkout	checkout_to_purchase
2017-12-20	100	20	16	0.2	0.8
2017-12-21	150	33	28	0.22	0.848484848484849
2017-12-22	250	62	55	0.248	0.887096774193548
2017-12-23	275	68	64	0.247272727272727	0.941176470588235