Case Study - 6

Wednesday, February 9, 2022 2:26 PM

Section A

Enterprise Relationship Diagram

Click Here Link For Diagram

https://dbdiagram.io/d/6203629e85022f4ee5612771

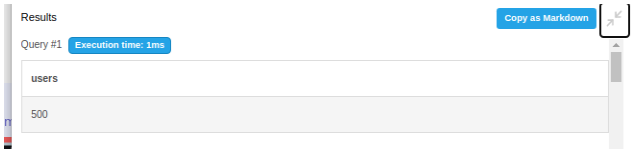
```
Table users as U {
  user_id int [pk, increment] // auto-increment
  cookie_id varchar(6)
  start_date timestamp
}


Table events as E {
  visit_id varchar(6)
  cookie_id varchar(6)
  page_id int
  event_type integer
  sequence_number integer

}

Table Page_hierarchy {
  page_id int [pk,increment]
  page_name varchar(14)
  product_category varchar(19)
  product_id integer
}


Table event_Identifier {
  event_type integer
  event_name varchar(13)
}

Table compaign_identifier {
  compaign_id int [pk,increment]
  products varchar(3)
  compaign_name varchar(33)
  start_date timestamp
  end_date timestamp
}

Ref onetomany {
users.user_id < events.cookie_id
}

Ref manytoone {
events.event_type > event_Identifier.event_type
}

ref manytoone {
  events.page_id > Page_hierarchy.page_id
}
```

Digital Analysis

1. How many users are there?

Choose file No file chosen

```
select count(Distinct user_id) As users from clique_bait.users;
```

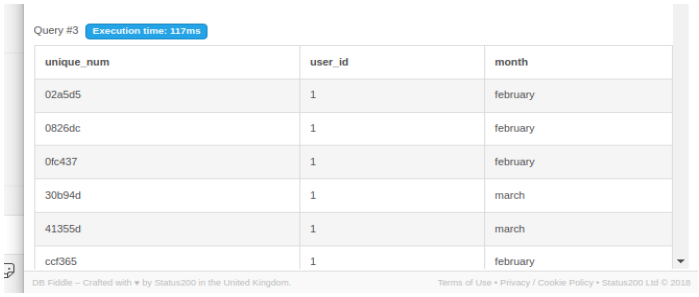


2. How many cookies does each user have on average?

```
select Round(avg(total_cookies),2) as AVG_cookie
        from (select Distinct(user_id) As users,count(cookie_id) as total_cookies from  clique_bait.users group by users
order by users) as a;
```

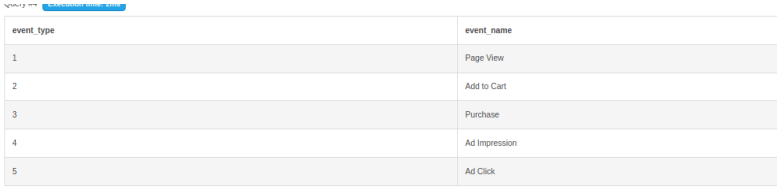
3. What is the unique number of visits by all users per month?

```
select distinct(e.visit_id) as Unique_num,
        u.user_id,
        to_char(e.event_time,'month') as month
        from clique_bait.users u inner join clique_bait.events e  on e.cookie_id=u.cookie_id group by
        month,user_id,Unique_num order by user_id;
```



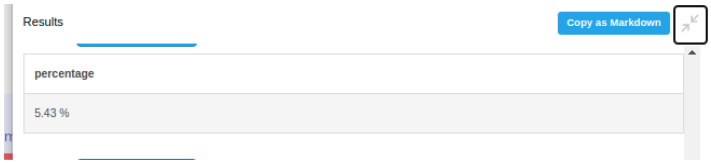
4. What is the number of events for each event type?

```
select * from clique_bait.event_identifier;
```



5. What is the percentage of visits which have a purchase event?

```
select Round((sum(CASE WHEN ei.event_name='Purchase' then 1
    ELSE 0
END)::NUMERIC /
 count(e.visit_id))*100,2) || ' %' as Percentage from clique_bait.events e inner join clique_bait.event_identifier ei on
e.event_type=ei.event_type;
```



6. What is the percentage of visits which view the checkout page but do not have a purchase event?

```
select Round((sum(CASE WHEN p.page_name='Checkout' and ei.event_name!='Purchase' then 1
    ELSE 0
END)::NUMERIC /
 count(e.visit_id))*100,2) || ' %' as Percentage
 from clique_bait.events e inner join clique_bait.event_identifier ei on e.event_type=ei.event_type inner join
clique_bait.page_hierarchy p on
```

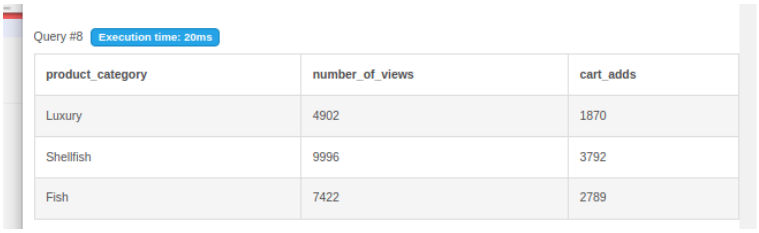
e.page_id=p.page_id;

7. What are the top 3 pages by number of views?

```
select page_name from
 (select p.page_name as page_name,count(e.page_id) as total
 from
 clique_bait.page_hierarchy p inner join clique_bait.events e
 on e.page_id=p.page_id group by page_name order by total DESC LIMIT 3) t;
```

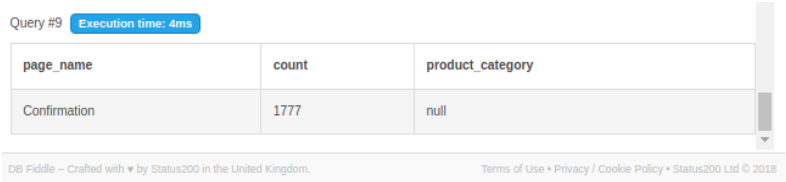
8. What is the number of views and cart adds for each product category?

```
select p.product_category,count(e.page_id) as Number_of_views,
sum(CASE WHEN e.event_type=2 then 1 ELSE 0 END)as Cart_Adds
from  clique_bait.page_hierarchy p inner join clique_bait.events e
on e.page_id=p.page_id where product_category is not null group by product_category;
```



9. What are the top 3 products by purchases?

```
select p.page_name,count(e.visit_id),product_category
from  clique_bait.page_hierarchy p inner join clique_bait.events e
on e.page_id=p.page_id  inner join clique_bait.event_identifier ei
on  e.event_type=ei.event_type where ei.event_name='Purchase' group  by  page_name,product_category LIMIT 3;
```



Product Funnel Analysis

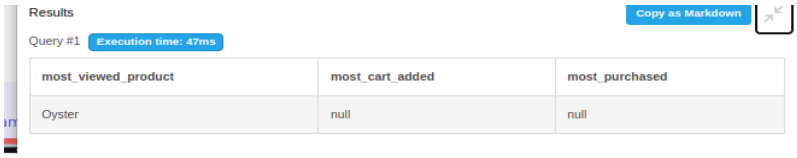
Using a single SQL query - create a new output table which has the following details:

- How many times was each product viewed?
- How many times was each product added to cart?
- How many times was each product added to a cart but not purchased (abandoned)?
- How many times was each product purchased?

1. Which product had the most views, cart adds and purchases?

```
WITH Product_CTE as (
SELECT p.page_name As products,p.product_id,
```

```
SUM(CASE WHEN ei.event_name='Page View' then 1 ELSE 0 END) AS product_Viewed,
SUM(CASE WHEN ei.event_name='Add to Cart' then 1 ELSE 0 END) AS Added_to_cart,
SUM(CASE WHEN ei.event_name='Add to Cart' and ei.event_name!='Purchase' then 1 ELSE 0 END) AS Added_to_Cart___But_Not_Purchased,
SUM(CASE WHEN ei.event_name='Purchase' then 1 ELSE 0 END) AS Product_Purchased
from  clique_bait.page_hierarchy p inner join clique_bait.events e
on e.page_id=p.page_id  inner join clique_bait.event_identifier ei
on  e.event_type=ei.event_type
Where product_id is NOT NULL
group by page_name,product_id order by product_id)
select
(select products from Product_CTE where product_Viewed=(select max(Product_Viewed) from Product_CTE))As Most_Viewed_product,
(select products from Product_CTE where product_Viewed=(select max(Added_to_cart) from Product_CTE))As Most_Cart_added,
(select products from Product_CTE where product_Viewed=(select max(Product_Purchased) from Product_CTE))As Most_Purchased;
```



Results — Copy as Markdown

Query #1 — Execution time: 47ms

most_viewed_product	most_cart_added	most_purchased
Oyster	null	null

2. Which product was most likely to be abandoned?

```
WITH Product_CTE as (
SELECT p.page_name As products,p.product_id,
SUM(CASE WHEN ei.event_name='Page View' then 1 ELSE 0 END) AS product_Viewed,
SUM(CASE WHEN ei.event_name='Add to Cart' then 1 ELSE 0 END) AS Added_to_cart,
SUM(CASE WHEN ei.event_name='Add to Cart' and ei.event_name!='Purchase' then 1 ELSE 0 END) AS Added_to_Cart___But_Not_Purchased,
SUM(CASE WHEN ei.event_name='Purchase' then 1 ELSE 0 END) AS Product_Purchased
from  clique_bait.page_hierarchy p inner join clique_bait.events e
on e.page_id=p.page_id  inner join clique_bait.event_identifier ei
on  e.event_type=ei.event_type
Where product_id is NOT NULL
group by page_name,product_id order by product_id)

select products from product_CTE order by Added_to_Cart___But_Not_Purchased DESC LIMIT 1;
```

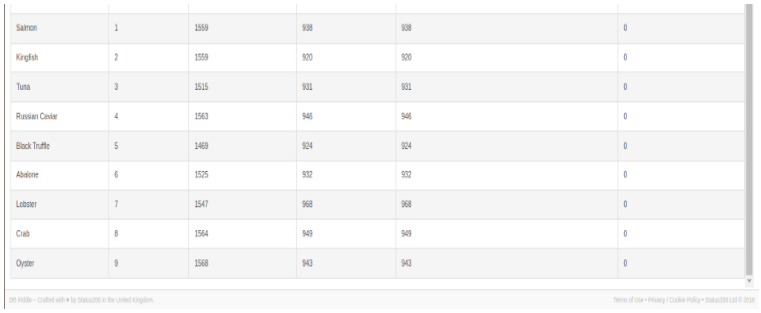
Query #2 — Execution time: 39ms

products
Lobster

3. Which product had the highest view to purchase percentage?

```
WITH Product_CTE as (
SELECT p.page_name As products,p.product_id,
SUM(CASE WHEN ei.event_name='Page View' then 1 ELSE 0 END) AS product_Viewed,
SUM(CASE WHEN ei.event_name='Add to Cart' then 1 ELSE 0 END) AS Added_to_cart,
SUM(CASE WHEN ei.event_name='Add to Cart' and ei.event_name!='Purchase' then 1 ELSE 0 END) AS Added_to_Cart___But_Not_Purchased,
SUM(CASE WHEN ei.event_name='Purchase' then 1 ELSE 0 END) AS Product_Purchased
from  clique_bait.page_hierarchy p inner join clique_bait.events e
on e.page_id=p.page_id  inner join clique_bait.event_identifier ei
on  e.event_type=ei.event_type
Where product_id is NOT NULL
group by page_name,product_id order by product_id)

select * from Product_CTE;
```

It's Not Possible Because No products Purchased Yet so values of Purchased column all are Zeroes.



Salmon	1	1559	938	938	0
Kingfish	2	1559	920	920	0
Tuna	3	1515	931	931	0
Russian Caviar	4	1563	946	946	0
Black Truffle	5	1469	924	924	0
Abalone	6	1525	932	932	0
Lobster	7	1547	968	968	0
Crab	8	1564	949	949	0
Oyster	9	1568	943	943	0

4. What is the average conversion rate from view to cart add?

```
WITH Product_CTE as (

SELECT p.page_name As products,p.product_id,

        SUM(CASE WHEN ei.event_name='Page View' then 1 ELSE 0 END) AS product_Viewed,
        SUM(CASE WHEN ei.event_name='Add to Cart' then 1 ELSE 0 END) AS Added_to_cart,
        SUM(CASE WHEN ei.event_name='Add to Cart' and ei.event_name!='Purchase' then 1 ELSE 0 END) AS
Added_to_Cart___But_Not_Purchased,
        SUM(CASE WHEN ei.event_name='Purchase' then 1 ELSE 0 END) AS Product_Purchased
        from  clique_bait.page_hierarchy p inner join clique_bait.events e
        on e.page_id=p.page_id  inner join clique_bait.event_identifier ei
        on  e.event_type=ei.event_type
        Where product_id is NOT NULL
        group by page_name,product_id order by product_id)

        select step, Total as user_count, Round(coalesce(1.0 - Total::float/lag(Total, 1) over () , 1)::NUMERIC,2) as drop_off_rate
---Conversion_rate is simply 100% - drop-off rate
        from (select  'View Product' as step, sum(Product_viewed)as Total  from Product_CTE
UNION ALL
  select 'Add to Cart' as step, sum(Added_to_cart) as Total from Product_CTE) t;
```

Query #4 **Execution time: 41ms**

step	user_count	drop_off_rate
View Product	13869	1.00
Add to Cart	8451	0.39

5. What is the average conversion rate from cart add to purchase?

```
WITH Product_CTE as (

SELECT p.page_name As products,p.product_id,

        SUM(CASE WHEN ei.event_name='Page View' then 1 ELSE 0 END) AS product_Viewed,
        SUM(CASE WHEN ei.event_name='Add to Cart' then 1 ELSE 0 END) AS Added_to_cart,
        SUM(CASE WHEN ei.event_name='Add to Cart' and ei.event_name!='Purchase' then 1 ELSE 0 END) AS
Added_to_Cart___But_Not_Purchased,
        SUM(CASE WHEN ei.event_name='Purchase' then 1 ELSE 0 END) AS Product_Purchased
        from  clique_bait.page_hierarchy p inner join clique_bait.events e
        on e.page_id=p.page_id  inner join clique_bait.event_identifier ei
        on  e.event_type=ei.event_type
        Where product_id is NOT NULL
        group by page_name,product_id order by product_id)

        select step, Total as user_count, CASE WHEN Total>0 then Round(coalesce(1.0 - Total::float/lag(Total, 1) over () ,
1)::NUMERIC,2) ELSE 0 END as drop_off_rate   ---Conversion_rate is simply 100% - drop-off rate
        from (select  'View Product' as step, sum(Product_viewed)as Total  from Product_CTE

UNION ALL
         select  'Add to Cart' as step, sum(Added_to_cart)as Total  from Product_CTE
UNION ALL
  select 'Product_Purchased' as step, sum(Product_Purchased) as Total from Product_CTE)t;
```

Query #5 Execution time: 40ms

step	user_count	drop_off_rate
View Product	13869	1.00
Add to Cart	8451	0.39
Product_Purchased	0	0

DB Fiddle – Crafted with ♥ by Status200 in the United Kingdom. Terms of Use • Privacy / Cookie Policy • Status200 Ltd © 2018

Campaigns Analysis

Generate a table that has 1 single row for every unique visit_id record and has the following columns:

- user_id
- visit_id
- visit_start_time: the earliest event_time for each visit
- page_views: count of page views for each visit
- cart_adds: count of product cart add events for each visit
- purchase: 1/0 flag if a purchase event exists for each visit
- campaign_name: map the visit to a campaign if the visit_start_time falls between the start_date and end_date
- impression: count of ad impressions for each visit
- click: count of ad clicks for each visit
- **(Optional column)** cart_products: a comma separated text value with products added to the cart sorted by the order they were added to the cart (hint: use the sequence_number)

```
 with CTE as
(select
Distinct(e.visit_id) as visit,
   u.user_id,
   min(e.event_time) as Visit_Start_time,
   count(p.page_id) as Page_views,
   sum(case when ei.event_name='Add to Cart' then 1 Else 0 end) as Cart_adds,
sum(case when ei.event_name='Purchase' then 1 Else 0 end)as Purchase,
   ci.campaign_name,
   sum(case when ei.event_name='Ad Impression' then 1 Else 0 end)as impression,
   sum(case when ei.event_name='Ad Click' then 1 Else 0 end) as Click,
STRING_AGG(CASE WHEN e.event_type = 2 THEN p.page_name ELSE NULL END, ',' ORDER BY      e.sequence_number) AS
cart_products
   from clique_bait.users u inner join clique_bait.events e on
   u.cookie_id=e.cookie_id inner join
   clique_bait.event_identifier ei on
   e.event_type=ei.event_type inner join
   clique_bait.page_hierarchy p on
   e.page_id=p.page_id
   left join clique_bait.campaign_identifier ci on
   e.event_time between ci.start_date and ci.end_date
   group by visit,user_id,campaign_name
order by user_id)

   select * from CTE;
```

Query #1 Execution time: 283ms

visit	user_id	visit_start_time	page_views	cart_adds	purchase	campaign_name	impression	click	cart_products
02a5d5	1	2020-02-26T16:57:26.260Z	4	0	0	Half Off - Treat Your Shelf(ish)	0	0	null
0826dc	1	2020-02-26T05:58:37.918Z	1	0	0	Half Off - Treat Your Shelf(ish)	0	0	null
0fc437	1	2020-02-04T17:49:49.602Z	19	6	1	Half Off - Treat Your Shelf(ish)	1	1	Tuna,Russian Caviar,Black Truffle,Abalone,Crab,Oyster
30b94d	1	2020-03-15T13:12:54.023Z	19	7	1	Half Off - Treat Your Shelf(ish)	1	1	Salmon,Kingfish,Tuna,Russian Caviar,Abalone,Lobster,Crab
41355d	1	2020-03-25T00:11:17.860Z	7	1	0	Half Off - Treat Your Shelf(ish)	0	0	Lobster
ccf365	1	2020-02-04T19:16:09.182Z	11	3	1	Half Off - Treat Your Shelf(ish)	0	0	Lobster,Crab,Oyster
eaffde	1	2020-03-25T20:06:32.342Z	21	8	1	Half Off - Treat Your Shelf(ish)	1	1	Salmon,Tuna,Russian Caviar,Black Truffle,Abalone,Lobster,Crab,Oyster
f7c798	1	2020-03-15T02:23:26.312Z	13	3	1	Half Off - Treat Your Shelf(ish)	0	0	Russian Caviar,Crab,Oyster
0635fb	2	2020-02-16T06:42:42.735Z	14	4	1	Half Off - Treat Your Shelf(ish)	0	0	Salmon,Kingfish,Abalone,Crab
1f1198	2	2020-02-01T21:51:55.078Z	1	0	0	Half Off - Treat Your Shelf(ish)	0	0	null
3b5871	2	2020-01-18T10:16:32.158Z	18	6	1	25% Off - Living The Lux Life	1	1	Salmon,Kingfish,Russian Caviar,Black Truffle,Lobster,Oyster

49d73d	2	2020-02-16T06:21:27.138Z	23	9	1	Half Off - Treat Your Shelf(ish)	1	1	Salmon,Kingfish,Tuna,Russian Caviar,Black Truffle,Abalone,Lobster,Crab,Oyster
910d9a	2	2020-02-01T10:40:46.875Z	9	1	0	Half Off - Treat Your Shelf(ish)	0	0	Abalone
c5c0ee	2	2020-01-18T10:35:22.765Z	1	0	0	25% Off - Living The Lux Life	0	0	null

DB Fiddle – Crafted with ♥ by Status200 in the United Kingdom. Terms of Use • Privacy / Cookie Policy • Status200 Ltd © 2018