A Digital Analysis

Using the available datasets - answer the following questions using a single query for each one:

1. *How many users are there?*

```
Select count(distinct(user_id)) as number_of_users
From users;
```

number_of_users
500

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

```
Select round(count(distinct(cookie_id))/count(distinct(user_id)),0) as average_cookies
From users;
```

average_cookies
4

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

```
Select month(event_time) as number_of_month, count(distinct(visit_id)) as number_of_visits
From events
Group by number_of_month
Order by number_of_month;
```

number_of_month	number_of_visits
1	876
2	1488
3	916
4	248
5	36

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

```
Select e.event_type, i.event_name, count(e.visit_id) as number_of_events
From events as e
Join event_identifier as i On i.event_type = e.event_type
Group by event_type
Order by event_type;
```

event_type	event_name	number_of_events
1	Page View	20928
2	Add to Cart	8451
3	Purchase	1777
4	Ad Impression	876
5	Ad Click	702

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

```
SELECT round(100 * COUNT(DISTINCT e.visit_id)/(SELECT COUNT(DISTINCT visit_id) FROM events),2) AS percentage_purchase
FROM events AS e
JOIN event_identifier AS i ON e.event_type = i.event_type
WHERE i.event_name = 'Purchase';
```

percentage_purchase
49.86

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

```
Create View Checkout_and_purchase As
Select visit_id, max(case
when event_type = 1 and page_id =12 Then 1 Else 0 End )as view_the_checkout_page,
max(case when event_type = 3 Then 1 Else 0 End) as purchase_event
From events
Group by visit_id
Order by visit_id;
```

visit_id	view_the_checkout_page	purchase_event
001597	1	1
002809	0	0
0048b2	1	0
004aaf	1	1
005fe7	1	1
006a61	1	1

```
SELECT 100-round((100*sum(purchase_event)/sum(view_the_checkout_page)),2) as percentage_checkout_page_not_purchase
FROM Checkout_and_purchase;
```

percentage_checkout_page_not_purchase
15.50

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

```
Select e.page_id, h.page_name, count(e.visit_id) as number_of_views
From events as e
Join page_hierarchy as h on e.page_id = h.page_id
Where e.event_type = 1
Group by e.page_id
Order by number_of_views DESC Limit 3;
```

page_id	page_name	number_of_views
2	All Products	3174
12	Checkout	2103
1	Home Page	1782

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

```
Select h.product_category, sum(
case when e.event_type = 1 Then 1 Else 0 End) as number_of_views,
sum(case when e.event_type = 2 Then 1 Else 0 End) as number_of_cart_adds
From events as e
Join page_hierarchy as h On e.page_id = h.page_id
Where h.product_category is not null
Group by h.product_category
Order by h.product_category;
```

product_category	number_of_views	number_of_cart_adds
Fish	4633	2789
Luxury	3032	1870
Shellfish	6204	3792

9. *What are the top 3 products by purchases?*

product_name	purchases
Lobster	754
Oyster	726
Crab	719

The table was created below.

B. 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?*

```sql
Create View page_view_cart_add As
SELECT e.visit_id, h.product_id, h.page_name AS product_name, h.product_category,
SUM(CASE WHEN e.event_type = 1 THEN 1 ELSE 0 END) AS page_view,
SUM(CASE WHEN e.event_type = 2 THEN 1 ELSE 0 END) AS cart_add
FROM events AS e
JOIN page_hierarchy AS h ON e.page_id = h.page_id
WHERE h.product_id IS NOT NULL
GROUP BY e.visit_id, h.product_id, h.page_name, h.product_category;

Create View purchase_table As
SELECT DISTINCT visit_id
FROM events
WHERE event_type = 3;

Create View page_card_purchase As
SELECT pv.visit_id, pv.product_id, pv.product_name, pv.product_category, pv.page_view, pv.cart_add,
CASE WHEN pt.visit_id IS NOT NULL THEN 1 ELSE 0 END AS purchase
FROM page_view_cart_add AS pv
LEFT JOIN purchase_table AS pt ON pv.visit_id = pt.visit_id;

Create View combined_table As
SELECT product_name, product_category, SUM(page_view) AS views, SUM(cart_add) AS cart_adds,
SUM(CASE WHEN cart_add = 1 AND purchase = 0 THEN 1 ELSE 0 END) AS abandoned,
SUM(CASE WHEN cart_add = 1 AND purchase = 1 THEN 1 ELSE 0 END) AS purchases
FROM page_card_purchase
GROUP BY product_id, product_name, product_category;
```

```sql
Select *
From combined_table
Order by product_name;
```

product_name	product_category	views	cart_adds	abandoned	purchases
Kingfish	Fish	1559	920	213	707
Lobster	Shellfish	1547	968	214	754
Oyster	Shellfish	1568	943	217	726
Russian Caviar	Luxury	1563	946	249	697
Salmon	Fish	1559	938	227	711
Tuna	Fish	1515	931	234	697

Additionally, create another table which further aggregates the data for the above points but this time for each product category instead of individual products.

```sql
Create View combined_table_2 As
SELECT product_category, SUM(page_view) AS views, SUM(cart_add) AS cart_adds,
SUM(CASE WHEN cart_add = 1 AND purchase = 0 THEN 1 ELSE 0 END) AS abandoned,
SUM(CASE WHEN cart_add = 1 AND purchase = 1 THEN 1 ELSE 0 END) AS purchases
FROM page_card_purchase
GROUP BY product_category;
```

product_category	views	cart_adds	abandoned	purchases
Luxury	3032	1870	466	1404
Shellfish	6204	3792	894	2898
Fish	4633	2789	674	2115

Use your 2 new output tables - answer the following questions:

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

```
Select product_name, views, cart_adds, purchases
From combined_table
Order by views DESC, cart_adds DESC, purchases DESC LIMIT 1;

Select product_name, views, cart_adds, purchases
From combined_table
Order by cart_adds DESC, views DESC, purchases DESC LIMIT 1;
```

product_name	views	cart_adds	purchases
Oyster	1568	943	726

product_name	views	cart_adds	purchases
Lobster	1547	968	754

11. Which product was most likely to be abandoned?

```
Select product_name, abandoned as most_likely_to_be_abandoned
From combined_table
Order by most_likely_to_be_abandoned DESC Limit 1;
```

product_name	most_likely_to_be_abandoned
Russian Caviar	249

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

```
Select product_name, round(100* purchases/views,2) as purchase_percentage
From combined_table
Order by purchase_percentage DESC LIMIT 1;
```

product_name	purchase_percentage
Lobster	48.74

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

```
Select round(100* sum(cart_adds)/sum(views) ,2) as average_view_to_cart_add_conversion_rate
From combined_table;
```

average_view_to_cart_add_conversion_rate
60.93

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

```
Select round(100* sum(purchases)/sum(cart_adds) ,2) as average_view_to_cart_add_conversion_rate
From combined_table;
```

average_view_to_cart_add_conversion_rate
75.93