**Amrita kaur Bajaj**

-- 1) How many orders were received for products with a category_id = 2

SELECT * FROM `orders` WHERE category_id=2

-- 2) How many orders were received with a category_id of either 2, 4, or 5

SELECT * FROM `orders` WHERE category_id IN (2,4,5)

-- 3) How many order are there with a price over £35.00

SELECT * FROM `orders` WHERE price > 35

-- 4) How many orders are there where the customer has a date of birth before 1st January 1980 and want to receive the newsletter

SELECT * FROM `orders` WHERE 'date_of_birth' < 1980-01-01 AND 'Newsletter'='1'

-- 5) How many customers named Davenport placed orders?

SELECT * FROM `orders` where customer_surname='Davenport'

-- 6) Which customer with a firstname starting with 'Br', had the most orders

SELECT * FROM `orders` where customer_firstname like 'Br%'

order by customer_firstname

-- 7) List all orders with products from category 3 by order of price, highest first.

SELECT category_id, MAX(`price`)

FROM `orders` where category_id=3 GROUP BY category_id

-- 8) Select the following fields from all orders (trans_date, price, promo_code) renaming the colum (field) headings ('Transaction Date', 'Price' & 'Promotion Code')

select trans_date as "Transaction Date", price as "Price", promo_code as "Promotion Code" from orders

-- 9) Select the following fields (customer_surname, customer_firstname, county) from all orders, with customer names in a single field named 'Customer Name' and in the format <Surname>, <Firstname>, with surname capitalised. The county field is to be renamed 'County'.

SELECT concat(`customer_Surname`, ' ', `customer_firstname`) as 'name', `county` as 'County' FROM `orders`;

-- 10) Select the average price, minimum price & maximum price for each category.

SELECT `category_id`, COUNT(*), SUM(`price`), AVG(`price`), MIN(`price`), MAX(`price`) FROM `orders`

GROUP BY `category_id`;

-- 11) Select the category_name (labelled 'Category', number of sales (labelled 'Total Orders') & total sales (labelled 'Total Sales') for each category.

select category_name as "Category", count(orders.id) as "Total Orders", sum(price) as "Total sales" from orders join categories on (orders.category_id = categories.id) group by category_name;

-- 12) List all orders with the following fields (with the labls given)

orders.trans_date('Transaction Date'), categories.category_name('Category'), orders.customer_surname('Surname'), orders.customer_firstname('Firstname'), orders.price('Order Price'), categories.category_name('Category'), promotions.discount('Discounted by')

select category_name as "Category", count(orders.id) as "Total Orders", sum(price) as "Total sales" from orders join categories on (orders.category_id = categories.id) group by category_name;