|  |
| --- |
|  |
|  | -- SQL Query |
|  | -- |
|  |  |
|  |  |
|  | -- using the DDL in the file final\_exercise\_ddl.sql, perfom the following - |
|  |  |
|  |  |
|  | -- DQL Tasks |
|  |  |
|  | -- 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 'data\_of\_birth' > '1990-1-1' 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%' |
|  | -- 7) List all orders with products from category 3 by order of price, highest first. |
|  | SELECT \* FROM `orders` where category\_id = 3 ORDER BY price DESC |
|  | -- 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 'Customer Name', customer\_firstname AS 'Price', county AS 'County' FROM `orders` |
|  | -- 10) Select the average price, minimum price & maximum price for each category. |
|  | SELECT `category\_id`, 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('category\_id') AS 'Total Orders', SUM('price') AS 'Total Sales' FROM 'orders' JOIN 'categories' ON ( category\_id = 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 `trans\_date` AS 'Transaction Date', 'category\_name' AS 'Category', 'customer\_surname' AS 'Surname', 'customer\_firstname' AS 'Firstname', 'price' AS 'Order Price', 'category\_name' AS 'Category', 'discount'AS 'Discounted by' FROM 'orders' o JOIN 'categories' c ON ( o.category\_id = c.id ) JOIN 'promotions' p ON ( o.promo\_code = p.code ) |