**DANNY’S DINER**

**BACKGROUND**

Danny seriously loves Japanese food so in the beginning of 2021, he decides to embark upon a risky venture and opens up a cute little restaurant that sells his 3 favorite foods: sushi, curry, and ramen.

Danny’s Diner is in need of your assistance to help the restaurant stay afloat - the restaurant has captured some very basic data from their few months of operation but have no idea how to use their data to help them run the business.

**PROBLEM STATEMENT**

Danny wants to use the data to answer a few simple questions about his customers:

* about their visiting patterns
* how much money they have spent
* which menu items are their favorite.

Having this deeper connection with his customers will help him deliver a better and more personalized experience for his loyal customers.

He plans on using these insights to help him decide whether he should expand the existing customer loyalty program - additionally he needs help to generate some basic datasets so his team can easily inspect the data without needing to use SQL.

Danny has provided you with a sample of his overall customer data due to privacy issues - but he hopes that these examples are enough for you to write fully functioning SQL queries to help him answer his questions!

**UNDERSTANDING THE DATA**

Danny has shared with you 3 key datasets for this case study:

* sales
* menu
* members

You can inspect the entity relationship diagram below:

Graphical user interface, application

Description automatically generated

To understand the dataset we will be working with, I studied the Entity Relationship Diagrams of the Database and ran a few SQLite commands to get an overview of the tables in the database

**DATABASE:** dannys\_diner

**TABLES:**

* Sales: Captures customer purchases using unique customer ID with corresponding order date and product ID
* Menu: Captures relevant data (product ID, name, and price) of food menu items
* Members: Captures date when customer joined the Danny’s Diner loyalty program

**Sales**

**PRAGMA table\_info(sales);**

By using the above statement, we describe the sales table.

**Result:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **cid** | **Name** | **Type** | **notnull** | **dflt\_value** | **pk** |
| 0 | Customer\_id | VARCHAR(1) | 0 | Null | 0 |
| 1 | Order\_date | DATE | 0 | Null | 0 |
| 2 | Product\_id | INTEGER | 0 | Null | 0 |

**Sample:**

**SELECT \***

**FROM sales;**

|  |  |  |
| --- | --- | --- |
| **Customer\_id** | **Order\_date** | **Product\_id** |
| A | 2021-01-01 | 1 |
| B | 2021-01-01 | 2 |
| C | 2021-01-01 | 3 |

**Menu**

**PRAGMA table\_info(menu);**

By using the above statement, we describe the menu table.

**Result:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **cid** | **Name** | **Type** | **notnull** | **dflt\_value** | **pk** |
| 0 | Product\_id | INTEGER | 0 | Null | 0 |
| 1 | Product\_name | VARCHAR(5) | 0 | Null | 0 |
| 2 | Price | INTEGER | 0 | Null | 0 |

**Sample:**

**SELECT \***

**FROM menu;**

|  |  |  |
| --- | --- | --- |
| **Product\_id** | **Product\_name** | **Price** |
| 1 | sushi | 10 |
| 2 | curry | 15 |
| 3 | ramen | 12 |

**Members**

**PRAGMA table\_info(members);**

By using the above statement, we describe the members table.

**Result:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **cid** | **Name** | **Type** | **notnull** | **dflt\_value** | **pk** |
| 0 | Customer\_id | VARCHAR(1) | 0 | Null | 0 |
| 1 | Join\_date | DATE | 0 | Null | 0 |

**Sample:**

**SELECT \***

**FROM members;**

|  |  |
| --- | --- |
| **Customer\_id** | **Join\_date** |
| A | 2021-01-07 |
| B | 2021-01-09 |

**CASE STUDY QUESTIONS**

1. What is the total amount each customer spent at the restaurant?

**SELECT customer\_id**

**,sum(price) AS amount\_spent**

**FROM sales**

**JOIN menu ON sales.product\_id = menu.product\_id**

**GROUP BY customer\_id;**

|  |  |
| --- | --- |
| **Customer\_id** | **Amount\_spent** |
| A | 76 |
| B | 74 |
| C | 36 |

1. How many days has each customer visited the restaurant?

**SELECT customer\_id**

**,COUNT(DISTINCT order\_date) AS days\_visited**

**FROM sales**

**GROUP BY customer\_id;**

|  |  |
| --- | --- |
| **Customer\_id** | **Amount\_spent** |
| A | 4 |
| B | 6 |
| C | 2 |

1. What was the first item from the menu purchased by each customer?

**WITH first\_item**

**AS (**

**SELECT sales.customer\_id**

**,menu.product\_name**

**,ROW\_NUMBER() OVER (**

**PARTITION BY sales.customer\_id ORDER BY sales.order\_date**

**) AS rank**

**FROM sales**

**JOIN menu ON sales.product\_id = menu.product\_id**

**)**

**SELECT \***

**FROM first\_item**

**WHERE rank = 1;**

|  |  |  |
| --- | --- | --- |
| **Customer\_id** | **Product\_name** | **Rank** |
| A | Sushi | 1 |
| B | Curry | 1 |
| C | Ramen | 1 |

1. What is the most purchased item on the menu and how many times was it purchased by all customers?

**SELECT sales.product\_id**

**,menu.product\_name**

**,count(sales.product\_id) AS num\_of\_purchases**

**FROM sales**

**JOIN menu ON sales.product\_id = menu.product\_id**

**GROUP BY sales.product\_id**

**,menu.product\_name**

**ORDER BY num\_of\_purchases DESC LIMIT 1;**

|  |  |  |
| --- | --- | --- |
| **Product\_id** | **Product\_name** | **Num\_of\_purchases** |
| 3 | Ramen | 8 |

1. Which item was the most popular for each customer?

**WITH fave\_item**

**AS (**

**SELECT sales.customer\_id**

**,menu.product\_name**

**,COUNT(sales.product\_id) AS popular\_order**

**,DENSE\_RANK() OVER (**

**PARTITION BY sales.customer\_id ORDER BY COUNT(sales.product\_id) DESC**

**) AS rank**

**FROM sales**

**JOIN menu ON sales.product\_id = menu.product\_id**

**GROUP BY sales.customer\_id**

**,menu.product\_name**

**)**

**SELECT \***

**FROM fave\_item**

**WHERE rank = 1;**

|  |  |  |  |
| --- | --- | --- | --- |
| **Customer\_id** | **Product\_name** | **Popular\_order** | **Rank** |
| A | Ramen | 3 | 1 |
| B | Sushi | 2 | 1 |
| B | Curry | 2 | 1 |
| B | Ramen | 2 | 1 |
| C | Ramen | 3 | 1 |

1. Which item was purchased first by the customer after they became a member?

**WITH first\_member\_item**

**AS (**

**SELECT sales.customer\_id**

**,sales.product\_id**

**,menu.product\_name**

**,members.join\_date**

**,DENSE\_RANK() OVER (**

**PARTITION BY sales.customer\_id ORDER BY**

**sales.order\_date ,sales.customer\_id**

**) AS rank**

**FROM sales**

**JOIN members ON sales.customer\_id = members.customer\_id**

**JOIN menu ON sales.product\_id = menu.product\_id**

**WHERE sales.order\_date >= members.join\_date**

**)**

**SELECT customer\_id**

**,product\_name**

**FROM first\_member\_item**

**WHERE rank = 1;**

|  |  |
| --- | --- |
| **Customer\_id** | **Product\_name** |
| A | Curry |
| B | Sushi |

1. Which item was purchased just before the customer became a member?

**WITH prior\_member\_item**

**AS (**

**SELECT sales.customer\_id**

**,sales.product\_id**

**,menu.product\_name**

**,sales.order\_date**

**,members.join\_date**

**,DENSE\_RANK() OVER (**

**PARTITION BY sales.customer\_id ORDER BY**

**sales.order\_date DESC**

**) AS rank**

**FROM sales**

**JOIN members ON sales.customer\_id = members.customer\_id**

**JOIN menu ON sales.product\_id = menu.product\_id**

**WHERE sales.order\_date < members.join\_date**

**)**

**SELECT customer\_id, product\_name**

**FROM prior\_member\_item**

**WHERE rank = 1;**

|  |  |
| --- | --- |
| **Customer\_id** | **Product\_name** |
| A | Sushi |
| A | Curry |
| B | Sushi |

1. What is the total items and amount spent for each member before they became a member ?

**SELECT sales.customer\_id**

**,COUNT(sales.product\_id) AS total\_purchases**

**,SUM(menu.price) AS total\_cost**

**FROM sales**

**JOIN menu ON sales.product\_id = menu.product\_id**

**JOIN members ON sales.customer\_id = members.customer\_id**

**WHERE sales.order\_date < members.join\_date**

**GROUP BY sales.customer\_id**

**ORDER BY sales.customer\_id;**

|  |  |  |
| --- | --- | --- |
| **Customer\_id** | **Total\_purchases** | **Total\_cost** |
| A | 2 | 25 |
| B | 3 | 40 |

1. If each $1 spent equates to 10 points and sushi has a 2x points multiplier - how many points would each customer have?

**WITH price\_points**

**AS (**

**SELECT \***

**,CASE**

**WHEN menu.product\_name = "sushi"**

**THEN menu.price \* 10 \* 2**

**ELSE price \* 10**

**END AS points\_gained**

**FROM menu**

**)**

**SELECT sales.customer\_id**

**,SUM(points\_gained) AS customer\_points**

**FROM price\_points**

**JOIN sales ON price\_points.product\_id = sales.product\_id**

**GROUP BY sales.customer\_id**

|  |  |
| --- | --- |
| **Customer\_id** | **Customer\_points** |
| A | 860 |
| B | 940 |
| C | 360 |

1. In the first week after a customer joins the program (including their join date) they earn 2x points on all items, not just sushi - how many points do customer A and B have at the end of January?

**WITH jan\_and\_first\_week**

**AS (**

**SELECT \***

**,DATEADD(DAY, 6, join\_date) AS first\_week**

**,EOMONTH('2021-01-31') AS jan\_end**

**FROM members**

**)**

**SELECT jfw.customer\_id**

**,SUM(CASE**

**WHEN menu.product\_name = 'sushi'**

**THEN 2 \* 10 \* menu.price**

**WHEN sales.order\_date BETWEEN jfw.join\_date**

**AND jfw.first\_week**

**THEN 2 \* 10 \* menu.price**

**ELSE 10 \* menu.price**

**END) AS total\_jan\_points**

**FROM jan\_and\_first\_week AS jfw**

**JOIN sales ON jfw.customer\_id = sales.customer\_id**

**JOIN menu ON sales.product\_id = menu.product\_id**

**WHERE sales.order\_date < jfw.jan\_end**

**GROUP BY jfw.customer\_id;**

|  |  |
| --- | --- |
| **Customer\_id** | **Total\_jan\_points** |
| A | 1370 |
| B | 820 |