databricks03 - Joins

Create Tables

Run the cell below to create tables for the questions in this notebook.

```
%run ../Utilities/03-CreateTables
```

Declared the following tables:
- **purchases**
- **prices**

Declared the following tables:
- **discounts**
- **products**

Declared the following tables:
- **stores**
- **articles**

Question 1: Joins

Summary

Perform a join of two tables `purchases` and `prices`.

Steps to complete

Write a SQL query that acheives the following:

- Performs an inner join on two tables `purchases` and `prices` on the column `itemId`

- Includes **only** 3 columns from the `purchases` table: `transactionId`, `itemId`, and `value`

```
DESCRIBE purchases
```

	col_name	data_type	comment
1	transactionId	int	null
2	itemName	string	null
3	itemId	bigint	null

Showing all 3 rows.

```
DESCRIBE prices
```

	col_name	data_type	comment
1	itemId	bigint	null
2	value	double	null

Showing all 2 rows.

```
SELECT
transactionId,
purchases.itemId,
prices.value
FROM
purchases
JOIN prices
ON
purchases.itemId = prices.itemId;
```

	transactionId	itemId	value
1	3072	721798132696267900	7639.71
2	3073	5274937845866294000	593.41
3	3074	1184079848143570000	495.42
4	3075	6625090293814504000	8683.67
5	3076	4094462754234261500	1989.74
6	3077	3719347016390063000	9250.49
7	3078	1769676752351633700	4980.62

Truncated results, showing first 1000 rows.

Question 2: Combine Tables

Summary

Perform an outer join on two tables `discounts` and `products` store the results into `q2Results` view.

Steps to complete

Write a SQL query that achieves the following:

- Performs an outer join on table `discounts` and `products` on the `itemName` column
- Ensures that the joined view only includes **one** `itemName` column that comes from the `products` table

The final schema and DataFrame should contain the following columns, though not necessarily in this order:

column	type
itemName	string
discountId	integer
discountCode	string
price	double
active	boolean
itemId	integer
amount	integer

```
SELECT
products.itemName,
discountId,
discountCode,
price,
active,
products.itemId,
products.amount
FROM
products
FULL OUTER JOIN discounts
ON
products.itemName = discounts.itemName;
```

	itemName	discountId	discountCode	price	active
1	Aliqua Dolore	182464	TKA8zpq7ylfH	3.52	false
2	Aliqua Ea	null	null	null	null
3	Aliqua Id	null	null	null	null
4	Aliqua Minim	968504	qI2a5pyV12PecVX2tFE	64.6	true
5	null	598656	mGljpnEVgxncz3cUvKL	43.81	true
6	null	175592	fK7MYqtKVf	54.56	false
7	Excepteur Enim	217000	PIrHip77dC1D	23.73	false

Showing all 15 rows.

Question 3: Cross Join Tables

Summary

Perform a cross join on two tables `stores` and `articles`.

Steps to complete

- Perform a cross join on two tables `stores` and `articles`

```
SELECT *
FROM stores CROSS JOIN articles;
```

	storeName	storeId	itemName	itemId	amount

1	Excepteur Aute	172	Pariatur Incididunt	76589	473
2	Excepteur Aute	172	Aliqua Ea	67697	323
3	Excepteur Aute	172	Aliqua Minim	21837	190
4	Excepteur Aute	172	Excepteur Officia	12881	89
5	Excepteur Aute	172	Aliqua Dolore	26631	2199
6	Excepteur Aute	172	Ut Sit	23081	131
7	Aliqua Dolore	164	Pariatur Incididunt	76589	473

Showing all 24 rows.