Introduction to Snowflake SQL

INTRODUCTION TO SNOWFLAKE SQL
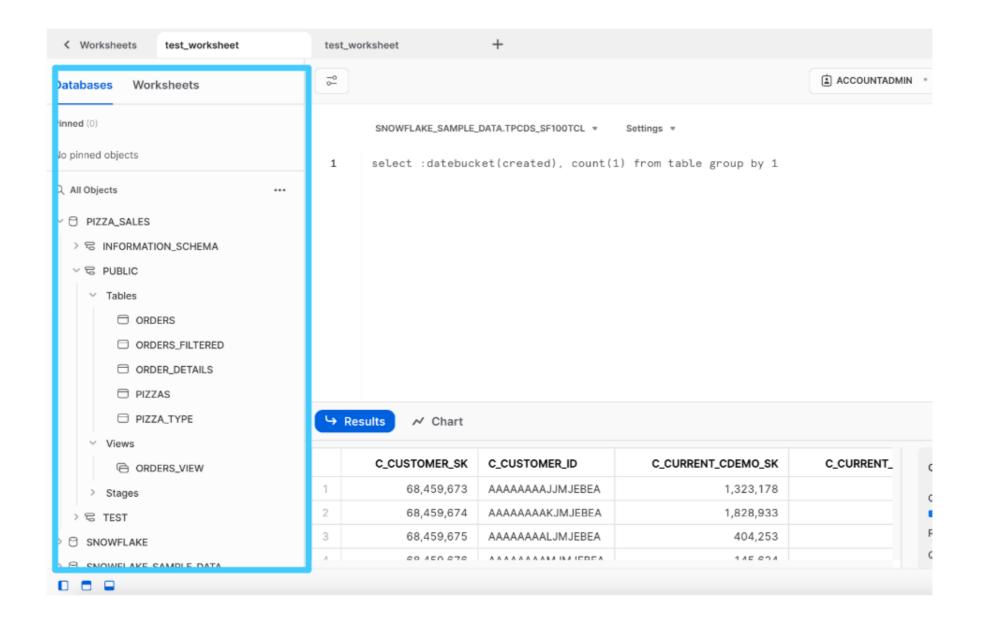
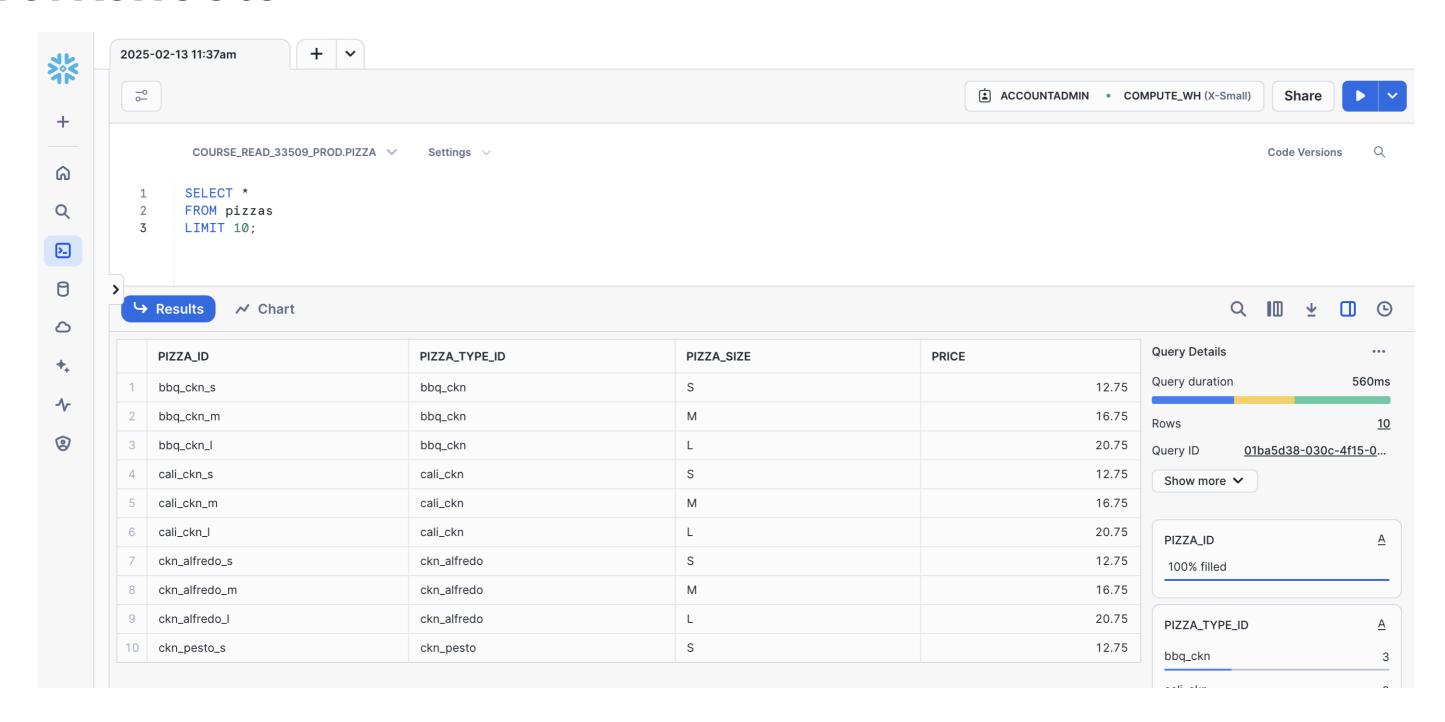**George Boorman**
Senior Curriculum Manager, DataCamp

datacamp
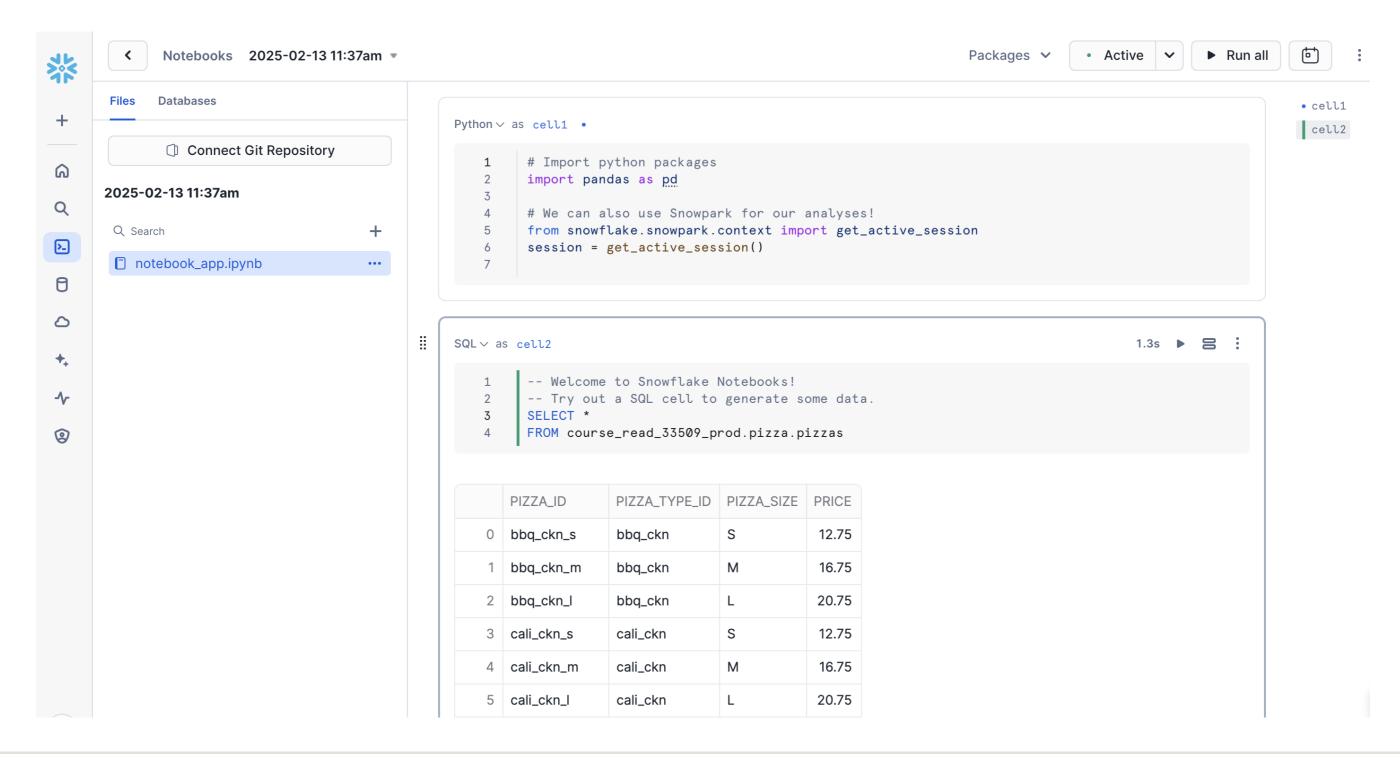
Connecting to Snowflake

- Snowsight: Snowflake Web Interface

Worksheets

Notebooks

Connecting to Snowflake: Drivers

Drivers & Connectors

- ODBC (Open Database Connectivity)

- JDBC (Java Database Connectivity)

- Connectors : Python, Spark, and more

[1] https://docs.snowflake.com/en/developer-guide/drivers

Connecting to Snowflake: Snowflake CLI

Snowflake CLI

- Command-line client
  - Installed on Linux, Windows, or Mac

[1] https://docs.snowflake.com/en/user-guide/snowsql

SQL flavors

- Snowflake uses Snowflake SQL

- Other popular SQL flavors:
  - PostgreSQL

  - T-SQL

  - MySQL

- Some differences in data types, functions, and general syntax

Common syntax

- `SELECT`

- `FROM`

- `WHERE`

- `GROUPBY`

- `ORDER BY`

- `AVG()` , `COUNT()` , `SUM()` (etc)

- `DISTINCT`

- `HAVING`

- `JOIN` *

*Some differences exist, more on this later!

Let's practice!

INTRODUCTION TO SNOWFLAKE SQL

Snowflake SQL data types

INTRODUCTION TO SNOWFLAKE SQL

George Boorman
Senior Curriculum Manager, DataCamp

Common data types

Category	Data types
Text/string	`VARCHAR` , `CHAR` , `TEXT`

Common data types

Category	Data types
Text/string	`VARCHAR` , `CHAR` , `TEXT`
Numeric	`INTEGER`

Common data types

Category	Data types
Text/string	`VARCHAR` , `CHAR` , `TEXT`
Numeric	`INTEGER`
Boolean	`BOOLEAN`

Common data types

Category	Data types
Text/string	`VARCHAR` , `CHAR` , `TEXT`
Numeric	`INTEGER`
Boolean	`BOOLEAN`
Date/time	`DATE` , `TIME` , `TIMESTAMP`

[1] https://docs.snowflake.com/en/sql-reference/intro-summary-data-types

Snowflake SQL data types - NUMBER

`NUMBER(p, s)`

- `NUMERIC` works in Snowflake as an alias for `NUMBER`

- `p` = precision; `s` = scale

- Max `p` and `s` values: `38`
  - Exceeding will cause rounding!

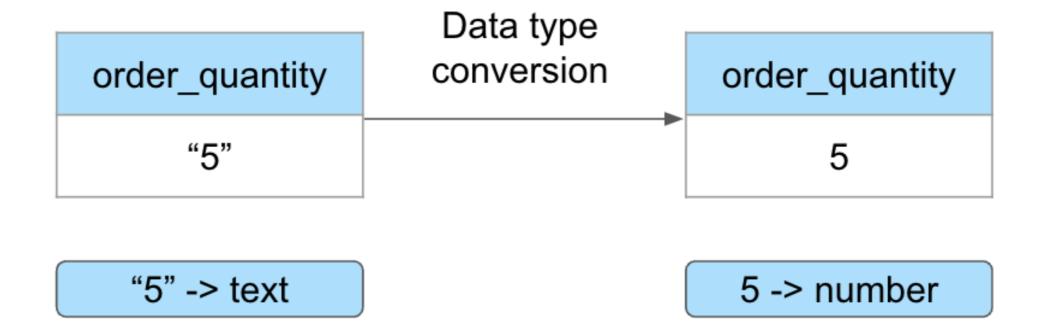# Snowflake SQL data types - TIMESTAMP_LTZ

- `TIMESTAMP_LTZ`
  - Combines `DATE` and `TIME` with local time zone
  - Format: `YYYY-MM-DD HH:MI:SS`

```
CREATE TABLE orders (
  -- Timestamp with local time zone
  order_timestamp TIMESTAMP_LTZ
)
```

	ORDER_TIMESTAMP
1	2015-01-01 11:38:36.000 -0800
2	2015-01-01 11:57:40.000 -0800
3	2015-01-01 12:12:28.000 -0800
4	2015-01-01 12:16:31.000 -0800
5	2015-01-01 12:21:30.000 -0800

Data type conversion - What?

- Converting data from one type to another

Data type conversion - Why?

- Improving performance

- Data accuracy and consistency

- Data quality

Data type conversion - How?

1. CAST *Syntax:*
   - `CAST( <source_data/column> AS <target_data_type> )`

   - `CAST('80' AS INT)`

2. :: *Syntax:*
   - `<source_data/column>::<target_data_type>`

   - `'80'::INT`

CAST

```
SELECT CAST(order_timestamp AS DATE)
        AS order_date
FROM orders
```

CAST results

	ORDER_TIMESTAMP
1	2015-01-01 11:38:36.000 -0800
2	2015-01-01 11:57:40.000 -0800
3	2015-01-01 12:12:28.000 -0800
4	2015-01-01 12:16:31.000 -0800
5	2015-01-01 12:21:30.000 -0800

	ORDER_DATE
1	2015-01-01
2	2015-01-02
3	2015-01-03
4	2015-01-04
5	2015-01-05

Conversion functions

- Examples: `TO_VARCHAR` , `TO_DATE` , etc.

TO_VARCHAR

- `TO_VARCHAR( <expr> )`
  - expr - numeric, timestamp, etc.
  - Result: `VARCHAR`

Example:

```
SELECT TO_VARCHAR(86)
```

Result:

`86`

Checking data types

```
DESC TABLE orders
```

```
name        type            kind     null?   default   primary key
ORDER_ID    NUMBER(38,0)    COLUMN   N       null      Y
ORDER_DATE  DATE            COLUMN   Y       null      N
ORDER_TIME  TIME(9)         COLUMN   Y       null      N
```

Let's practice!

INTRODUCTION TO SNOWFLAKE SQL

Functions, sorting, and grouping

INTRODUCTION TO SNOWFLAKE SQL

George Boorman
Senior Curriculum Manager, DataCamp

String functions - INITCAP

Syntax: `INITCAP( <expr> )`

- Capitalize each word in a string

```sql
SELECT INITCAP(category) AS capitalized_category
FROM pizza_type
```

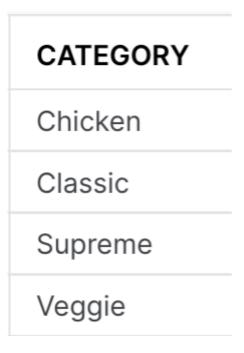| A CAPITALIZED_NAMES |
| --- |
| The Barbecue Chicken Pizza |
| The California Chicken Pizza |
| The Chicken Alfredo Pizza |
| The Chicken Pesto Pizza |

String functions - CONCAT

- Combines the expressions

Syntax:

```
CONCAT( <expr1> [ , <exprN> ... ] )
```

Before Concat:

CATEGORY
Chicken
Classic
Supreme
Veggie

- Combining `category` with ' - Pizza'

```
SELECT CONCAT(category, ' - Pizza')
    AS pizza_category
FROM pizza_type
```

After Concat:

PIZZA_CATEGORY
Chicken - Pizza
Classic - Pizza
Supreme - Pizza
Veggie - Pizza

DATE & TIME functions

- `CURRENT_DATE()` or `CURRENT_DATE`
- `CURRENT_TIME()` or `CURRENT_TIME`

```
SELECT CURRENT_DATE
```

```
SELECT CURRENT_TIME
```

CURRENT_DATE
2023-08-15

CURRENT_TIME
13:35:58

EXTRACT

Syntax

- EXTRACT(<date_or_time_part> FROM <date_or_time_expr>)
  - <date_or_time_part> - year , month , day , weekday ,etc.

```sql
SELECT EXTRACT(MONTH FROM order_date) AS order_month,
    COUNT(*) AS num_orders
FROM orders
GROUP BY order_month
```

# ORDER_MONTH	# NUM_ORDERS
1	1845
2	1685
3	1840

SORTING and GROUPING

- SORTING: `ORDER BY`

- GROUPING: `GROUP BY`
  - Snowflake: `GROUP BY ALL`

GROUP BY ALL

- `GROUP BY column1, column2`

- `GROUP BY ALL`

```sql
SELECT
    pizza_type_id,
    size,
    AVG(price) AS average_price
FROM
    pizzas
GROUP BY
    pizza_type_id, -- explicit columns
    size
ORDER BY
    pizza_type_id, average_price DESC
```

```sql
SELECT
    pizza_type_id,
    size,
    AVG(price) AS average_price
FROM
    pizzas
GROUP BY ALL -- Don't specify columns
ORDER BY
    pizza_type_id, average_price DESC
```

Summary

Function/keyword	Use
`INITCAP()`	Capitalize each word in a string
`CONCAT()`	Combine multiple strings
`CURRENT_DATE`	Get the current date
`CURRENT_TIME`	Get the current time
`EXTRACT`	Pull a date/time element, e.g., `month` from a date
`ORDER BY`	Sort query results
`GROUP BY ALL`	Group query by all (non-aggregated) columns

Let's practice!

INTRODUCTION TO SNOWFLAKE SQL