SQL data types reference

Snowflake supports most basic SQL data types (with some restrictions) for use in columns, local variables, expressions, parameters, and any other appropriate locations.

Note

You can also load unstructured data into Snowflake. For more information, see Introduction to unstructured data.

In some cases, data of one type can be converted to another type. For example, INTEGER data can be converted to FLOAT data.

Some conversions are lossless, but others might lose information. The amount of loss depends upon the data types and the specific values. For example, converting a FLOAT value to an INTEGER value removes the digits after the decimal place. (The value is rounded to the nearest integer.)

In some cases, the user must specify the desired conversion, such as when passing a VARCHAR value to the TIME_SLICE function, which expects a TIMESTAMP or DATE argument. We call this ***explicit casting***.

In other cases, data types are converted automatically, such as when adding a float and an integer. We call this ***implicit casting*** (or ***coercion***). In Snowflake, data types are automatically coerced whenever necessary and possible.

For more information about explicit and implicit casting, see Data type conversion.

- Numeric data types
- String & binary data types
- Logical data types
- Date & time data types
- Semi-structured data types
- Structured data types
- Unstructured data types
- Geospatial data types
- Vector data types
- Unsupported data types
- Data type conversion