Decimal Data Types

Hive and Impala support three decimal data types: FLOAT, DOUBLE, and DECIMAL. These types represent numbers that can include fractional parts.

The FLOAT and DOUBLE types represent floating-point numbers, which do not have a predetermined number of digits after the decimal point. DOUBLE offers greater range and precision than FLOAT, but processing DOUBLEs requires more memory than processing FLOATs, so in general, choose FLOAT unless you need the range or precision that DOUBLE offers.

Because of the binary system used to store numbers, both FLOAT and DOUBLE data types can produce unexpected inaccuracies, even with seemingly simple arithmetic like 0.1 + 0.2. (See The Floating Point Guide for more about this.)

DOUBLE is more accurate because DOUBLE uses 64 bits to store each number, while FLOAT uses only 32 bits. (So DOUBLE has *double* the number of bits.) This means FLOAT is typically accurate up to 7 digits, while DOUBLE is accurate up to 15 or maybe 16 digits.

The DECIMAL type represents numbers with fixed precision and scale. When you create a DECIMAL column, you specify the precision, *p*, and scale, *s*. Precision is the total number of digits, regardless of the location of the decimal point. Scale is the number of digits after the decimal place. To represent the number 8.54 without a loss of precision, you would need a DECIMAL type with precision of at least 3, and scale of at least 2.

The table below illustrates the difference in precision using results for π. Note that the DECIMAL(17,16) type means there is a total of 17 digits, with 16 of them after the decimal point.

Data Type	Result for π (bold are accurate)

FLOAT	3.1415927410125732
DOUBLE	3.1415926535897931
DECIMAL(17,16)	3.1415926535897932

Using the DECIMAL type, it is possible to represent numbers with greater precision than the FLOAT or DOUBLE types can represent. The maximum allowed precision and scale of the DECIMAL type are both 38. (Hive can allow larger values of precision and scale, but Impala does not support them.)

The table below describes the range of DOUBLE, FLOAT, and DECIMAL(38,0). The ranges described below are the largest negative and largest positive number that each data type can represent.

Data Type	Range
FLOAT	-3.40282346638528860 * 10^38 to 3.40282346638528860 * 10^38
DOUBLE	-1.79769313486231570 x 10^308 to 1.79769313486231570 x 10^308
DECIMAL(38,0)	-10^38 + 1 to 10^38 - 1

For representing currency, you should use DECIMAL instead of FLOAT or DOUBLE; this prevents loss of precision, which is typically of paramount importance with financial data. Another choice for currency is to use an integer type to represent, for example, the number of cents, instead of storing dollars with fractional parts.