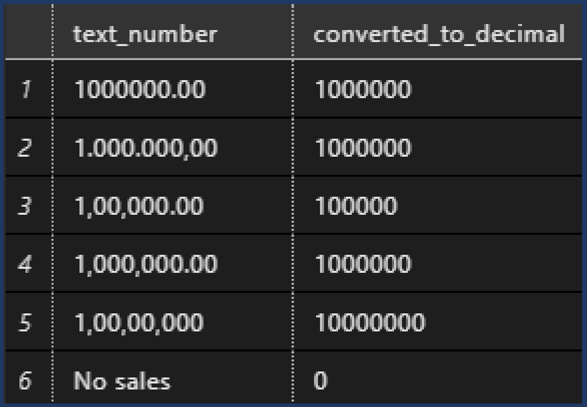
SQL: TRY_CAST and TRY_PARSE

```
INSERT INTO numbers_stored_as_text (text_number)
VALUES  ('1000000.00'),('1.000.000,00'),
        ('1,00,000.00'),('1,000,000.00'),
        ('1,00,00,000'),('No sales');

SELECT  text_number,
        COALESCE(
            TRY_PARSE(text_number AS decimal),
            TRY_PARSE(text_number AS decimal USING 'de-DE'),
            0
        ) AS converted_to_decimal
FROM numbers_stored_as_text;
```



	text_number	converted_to_decimal
1	1000000.00	1000000
2	1.000.000,00	1000000
3	1,00,000.00	100000
4	1,000,000.00	1000000
5	1,00,00,000	10000000
6	No sales	0

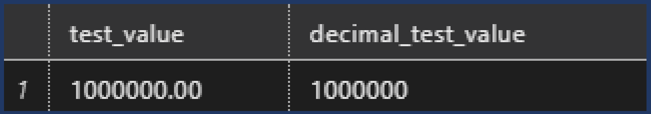
Numbers stored as text can be a problem

CAST is the ANSI standard function for converting from one data type to another

```
DROP TABLE IF EXISTS #testvalues;
CREATE TABLE #testvalues (test_value nvarchar(12));
INSERT INTO #testvalues (test_value)
VALUES
('1000000.00'),
('1.000.000,00'),
('1,00,000.00'),
('1,000,000.00'),
('1,00,00,000'),
('No sales');
```

CAST generates an error for these rows and will not return the row at all

```
SELECT test_value, CAST(test_value AS decimal) AS decimal_test_value
FROM #testvalues;
```



	test_value	decimal_test_value
1	1000000.00	1000000

CAST fails if a number stored as text contains anything other than a decimal separator for the current region settings

TRY_CAST returns NULL if the cast fails

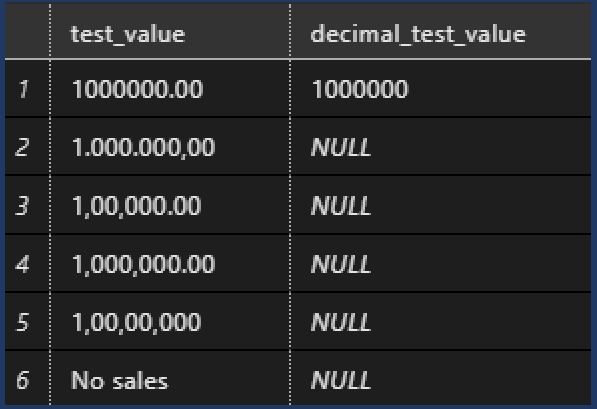
TRY_CAST is available in Snowflake, SQL Server, Db2 and various Azure Analytics products

```
DROP TABLE IF EXISTS #testvalues;
CREATE TABLE #testvalues (test_value nvarchar(12));
INSERT INTO #testvalues (test_value)
VALUES
('1000000.00'),
('1.000.000,00'),
('1,00,000.00'),
('1,000,000.00'),
('1,00,00,000'),
('No sales');
```

These values cannot be converted with TRY_CAST, so the query returns NULL

```
SELECT test_value, TRY_CAST(test_value AS decimal) AS decimal_test_value
FROM #testvalues;
```

TRY_CAST ensures the row is still returned if the cast has failed



	test_value	decimal_test_value
1	1000000.00	1000000
2	1.000.000,00	NULL
3	1,00,000.00	NULL
4	1,000,000.00	NULL
5	1,00,00,000	NULL
6	No sales	NULL

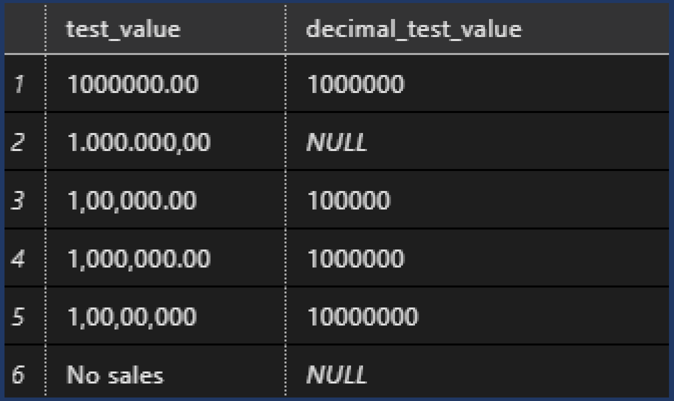
When TRY_CAST fails, we can use TRY_PARSE

```sql
DROP TABLE IF EXISTS #testvalues;
CREATE TABLE #testvalues (test_value nvarchar(12));
INSERT INTO #testvalues (test_value)
VALUES
('1000000.00'),
('1.000.000,00'),
('1,00,000.00'),
('1,000,000.00'),
('1,00,00,000'),
('No sales');

SELECT test_value, TRY_PARSE(test_value AS decimal) AS decimal_test_value
FROM #testvalues;
```

In the current culture, en-US, comma is the group-separator and period is the decimal separator, so some values are correctly parsed.

TRY_PARSE uses the current culture to identify group separators and decimal separators.



	test_value	decimal_test_value
1	1000000.00	1000000
2	1.000.000,00	*NULL*
3	1,00,000.00	100000
4	1,000,000.00	1000000
5	1,00,00,000	10000000
6	No sales	*NULL*

We can add a USING modifier to TRY_PARSE

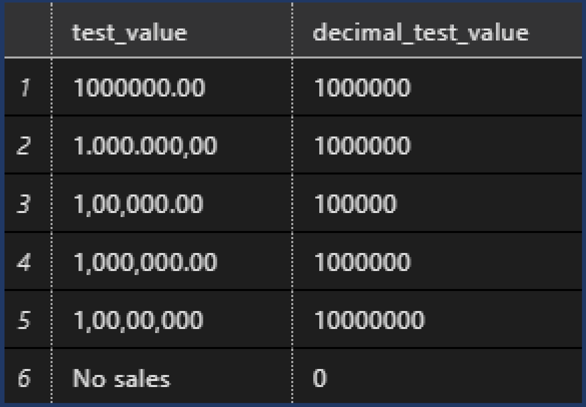
```sql
INSERT INTO #testvalues (test_value)
VALUES
('1000000.00'),
('1.000.000,00'),
('1,00,000.00'),
('1,000,000.00'),
('1,00,00,000'),
('No sales');

SELECT  test_value,
        COALESCE(
            TRY_PARSE(test_value AS decimal),
            TRY_PARSE(test_value AS decimal USING 'de-DE'),
            0
        ) AS decimal_test_value
FROM #testvalues;
```

If we don’t specify a culture in TRY_PARSE, the current culture is used. The first TRY_PARSE returns null if the current culture cannot parse the text-number

Replace any remaining NULL values with zero

If the current culture can’t parse the number, we try parsing the number with another culture, such as de-DE, which uses a period as the grouping separator and a comma as the decimal separator



	test_value	decimal_test_value
1	1000000.00	1000000
2	1.000.000,00	1000000
3	1,00,000.00	100000
4	1,000,000.00	1000000
5	1,00,00,000	10000000
6	No sales	0

Takeaways

1. Using CAST to convert numbers stored as text will fail if any non-numeric character other than the current culture’s decimal separator is present

2. Using TRY_CAST will return NULL if the number stored as text cannot be cast

3. Using TRY_PARSE will allow for grouping separators and allow us to specify an alternate culture to use in the conversion