LAB - Integer Data Types

In this lab, you will learn how about the integer data types and how to use them effectively to store integer values in the database.

SQL Server support standard SQL integer types including `BIGINT`, `INT`, `SMALLINT`, and `TINYINT`. The following table illustrates the range and storage of each integer type:

Data type	Range	Storage
BIGINT	-263 (-9,223,372,036,854,775,808) to 263-1 (9,223,372,036,854,775,807)	8 Bytes
INT	-231 (-2,147,483,648) to 231-1 (2,147,483,647)	4 Bytes
SMALLINT	-215 (-32,768) to 215-1 (32,767)	2 Bytes
TINYINT	0 to 255	1 Byte

It is a good practice to use the smallest integer data type that can reliably contain all possible values. For example, to store the number of children in a family, `TINYINT` is sufficient because nowadays no one could have more than 255 children. However, `TINYINT` is would not be sufficient for storing the stories of a building because a building can have more than 255 stories.

Example

The following statement creates a new table that consists of four integer columns:

```
CREATE TABLE sql_server_integers (
    bigint_col bigint,
    int_col INT,
    smallint_col SMALLINT,
    tinyint_col tinyint
);
```

The following `INSERT` statement adds the maximum integers of `BIGINT`, `INT`, `SMALLINT`, and `TINYINT` to the corresponding columns of the table:

```
INSERT INTO sql_server_integers (
    bigint_col,
    int_col,
    smallint_col,
    tinyint_col
)
VALUES
    (
        9223372036854775807,
        2147483647,
        32767,
        255
    );
```

To show the values stored in the `sql_server_integers` table, you use the following `SELECT` statement:

```
SELECT
    bigint_col,
    int_col,
    smallint_col,
    tinyint_col
FROM
    sql_server_integers;
```

In this basic lab, you have learned various SQL Server integer data types and how to use them to store integers in the database.