**Типы данных**

Как вы уже заметили, для хранения информации в таблицах используются различные типы данных, рассмотрим основные из них.

**Целочисленные типы данных**

Целочисленные типы данных используются для хранения точных числовых данных, например, возраста человека или количества единиц определенного товара. Ниже представлена таблица целочисленных типов данных с указанием названия, диапазона допустимых значений и количества занимаемой памяти (Таблица 2.1).

Таблица 2.1 Целочисленные типы данных

|  |  |  |
| --- | --- | --- |
| Название | Диапазон | Память |
| **bigint** | от -9 223 372 036 854 775 808 до 9 223 372 036 854 775 807 | 8 байт |
| **int** | от -2 147 483 648 до 2 147 483 647 | 4 байта |
| **smallint** | от -32 768 до 32 767 | 2 байта |
| **tinyint** | от 0 до 255 | 1 байт |

Еще одним целочисленным типом данных является bit, он занимает один байт памяти и может хранить значения 0, 1 или NULL, фактически этот тип данных является аналогом логического типа данных в таких языках программирования как C++ или C#.

**Типы данных для хранения текста**

Для хранения текстовой информации в базе данных используются четыре основных типа данных:

* char(n) – предназначен для хранения строк фиксированной длины не в кодировке Unicode, где вместо n указывается возможное количество символов в строке от 1 до 8000, память при этом выделяется по одному байту на символ;
* varchar(n | max) – также предназначен для хранения строк не в кодировке Unicode, но переменной длины, допустимое количество символов в строке можно задать либо указав значение от 1 до 8000, либо использовать для этого слово max и в этом случае для хранения строки выделится память до 2 ГБ;
* nchar(n) – предназначен для хранения строк фиксированной длины в кодировке Unicode, при помощи n задается максимально возможное количество символов в строке от 1 до 4000, размер выделенной памяти по два байта на символ;
* nvarchar(n | max) – тип данных, который предназначен для хранения строк переменной длины в кодировке Unicode, максимальное количество символов в строке можно задать от 1 до 4000 или выделить память для хранения строки до 2 ГБ, если использовать ключевое слово max.

В чем же состоит разница между строкой переменной длины и фиксированной строкой?

Если, например, для хранения строки указан тип данных char или nchar с размером 10 символов (char(10) или nchar(10)), а сохраняется строка из шести символов, то количество выделенной памяти будет соответствовать заявленному размеру – 10 или 20 байт, соответственно.

Если для хранения строки указан тип данных varchar или nvarchar с размером 10 символов (varchar(10) или nvarchar(10)), а сохраняется строка из шести символов, то размер выделенной памяти будет соответствовать реальному количеству символов – 6 или 12 байт, соответственно.

Вы также можете встретить типы данных text и ntext, но эти типы данных являются устаревшими и их использование не рекомендовано.

**Вещественные типы данных**

Для хранения числовых типов данных с плавающей точкой используются вещественные типы данных float и real.

При помощи типа данных float можно хранить числа в диапазоне от - 1,79E+308 до -2,23E-308, 0 и от 2,23E-308 до 1,79E+308.

При использовании типа данных float(n), указывая значение n, вы можете устанавливать точность числа и естественно изменять размер выделяемой памяти. Если значение n с 1 по 24, то SQL Server трактует его как 24 и выделяет память в 4 байта для хранения числа с точность до 7 знаков. Если в n указывается значение с 25 по 53, то SQL Server трактует его как 53 и выделяет 8 байт памяти для хранения числа с точность до 15 знаков. По умолчанию значение n равно 53.

Типа данных real позволяет хранить числа в диапазоне от - 3,40E + 38 до -1,18E - 38, 0 и от 1,18E - 38 до 3,40E + 38.

Синонимом для типа данных real является float(24).

**Типы данных для хранения даты и времени**

Для хранения в таблицах даты и времени SQL Server предоставляет ряд типов данных, которыми вы можете воспользоваться в зависимости от поставленной задачи.

Тип данных datetime позволяет хранить дату и время в 24-часовом формате с указанием долей секунды, в диапазоне с 1 января 1753 года по 31 декабря 9999 года, например, 2017-12-28 15:20:35.693.

Тип данных datetime2 позволяет хранить дату и время в 24-часовом формате, но имеет большую точность долей секунды по сравнению с типом datetime и диапазон дат с 1 января 0001 года по 31 декабря 9999 года, например, 2017-12-28 15:20:35.6930000.

Тип данных datetimeoffset позволяет хранить дату и время в 24-часовом формате с указанием долей секунды с учетом часового пояса в диапазоне с 1 января 0001 года по 31 декабря 9999 года, например, 2017-12-28 15:20:35.6930000 +00:00.

Предыдущие типы данных выводят довольно точные значения времени, но, если вам достаточно получить только дату и время, например, время покупки товара, тогда вы можете использовать тип данных smalldatetime. Этот тип данных позволяет хранить дату и время в 24-часовом формате с секундами, всегда равными нулю, без долей секунды в диапазоне с 1 января 1900 года по 6 июня 2079 года, например, 2017-12-28 15:20:00.

Если поле вашей таблицы предназначено для хранения только даты, например, даты рождения человека, то вам больше подойдет тип данных date, который позволяет хранить дату в диапазоне с 1 января 0001 года по 31 декабря 9999 года, например, 2017-12-28.

Тип данных time позволяет хранить только время в 24-часовом формате с указанием долей секунды без учета часового пояса в диапазоне от 00:00:00.0000000 до 23:59:59.9999999, например, 15:20:35.6930000.

**Типы данных с фиксированной точкой**

Для хранения вещественных значений в более точном формате используются типы данных decimal(p,s) и numeric(p,s). Эти типы данных являются взаимозаменяемыми и позволяют хранить данные в диапазоне от -1038+1 до 1038-1.

При помощи параметра p задается общее количество цифр в числе, как целой, так и дробной части. Диапазон значений от 1 до 38 (по умолчанию 18), чем большее значение вы укажете, тем больше байт памяти будет выделено для хранения информации. Значение в параметре s определяет количество цифр дробной части числа в диапазоне от 0 до p (по умолчанию 0).

Тип decimal(6,2) мы использовали в качестве типа данных для поля Grants таблицы Students из раздела № 1. В этом случае общее количество цифр в числе равно 6, а цифр после запятой 2, то есть целая часть должна состоять из 4 цифр.

**Другие типы данных**

Для хранения денежных значений используются типы данных money и smallmoney, которые представляют собой вещественные числа, дробная часть которых предназначена для хранения мелочи (копейки, центы и т.д.) (Таблица 2.2).

Таблица 2.2 Денежные типы данных

|  |  |  |
| --- | --- | --- |
| Название | Диапазон | Память |
| **money** | от -922 337 203 685 477,5808 до 922 337 203 685 477,5807 | 8 байт |
| **smallmoney** | от -214 748,3648 до 214 748,3647 | 4 байта |

Для хранения различной информации в бинарном виде используются типы данных binary(n) и varbinary(n | max). Параметр n может принимать значения от 1 до 8000, если указать ключевое слово max, то выделится память для хранения 2 ГБ информации.

Если у вас есть необходимость хранить данные в двумерной системе координат, тогда вы можете использовать тип данных geometry.

Для хранения координат широты и долготы объекта предназначен тип данных geography.

Хранение XML-документов обеспечивает тип данных xml.

Временный результирующий набор данных, полученных в результате запроса, сохраняется в типе данных table.

Обычно при создании таблицы полям указываются такие типы данных, максимальный размер которых гарантировано, позволит сохранить требуемый диапазон значений, такой подход используется в целях уменьшения размера базы данных.