# Построить схему хранения в Базе Данных

Первое задание я начал с продумывания всех полей, их типов и всех необходимых ограничений (CONSTRAINT).

У меня получился вот такой рисунок:



Изначальный вариант выглядел немного иначе, но потом я пересмотрел все изначальные данные и пришел к текущему варианту.

**Создание таблиц:**

PrintingDevices (устройства печати)

|  |
| --- |
| CREATE TABLE PrintingDevice (  Id INT IDENTITY,  Name NVARCHAR(100),  ConnectionType VARCHAR(15),  MACAdress CHAR(12),  CONSTRAINT PK\_PrintingDevice\_Id PRIMARY KEY (Id),  CONSTRAINT UQ\_PrintingDevice\_MACAdress UNIQUE (MACAdress)  ) |

|  |  |  |  |
| --- | --- | --- | --- |
| Поле Id (индентификатор) | Поле Name (Наименование устройства печати) | Поле ConnectionType (Тип подключения) | Поле MACAdress (MAC-адрес) |
| Уникальный первичный ключ типа INT | Поле с названием со всеми возможностями UNICODE типа NVARCHAR, макс длина 100 символов | Поле с типом подключения с типом VARCHAR и длиной 15 символов. Так как нам известно, что всего 2 варианта заполнения поля: локальное и сетевое. | Поле с уникальным физическим адресом устройства, поэтому оно ограниченной длины и уникально (12 символов и UNIQUE) |

Ограничения:

- первичный ключ Id

- уникальные значения MACAdress

Branch (филиал)

|  |
| --- |
| CREATE TABLE Branch (  Id INT IDENTITY,  PrintingDeviceId INT,  Number INT,  IsDefault VARCHAR(10),  CONSTRAINT PK\_Branch\_Id PRIMARY KEY (Id),  CONSTRAINT FK\_Branch\_PrintingDeviceId FOREIGN KEY (PrintingDeviceId) REFERENCES PrintingDevice (Id) ON DELETE CASCADE,  CONSTRAINT CK\_Branch\_Number CHECK(Number > 0)  ) |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

Ограничения:

-

-

-

DeviceInstallation (Инсталляция устройств)

|  |
| --- |
| CREATE TABLE DeviceInstallation (  Id INT IDENTITY,  Name NVARCHAR(100),  Location NVARCHAR(100),  CONSTRAINT PK\_DeviceInstallation\_Id PRIMARY KEY (Id)  ) |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Ограничения:

-

PrintSession (Сеансы печати)

|  |
| --- |
| CREATE TABLE PrintSession (  Id INT IDENTITY,  DeviceInstallationId INT,  Employee NVARCHAR(100),  BranchId INT,  PrintablePages INT,  Status VARCHAR(10),  CONSTRAINT PK\_PrintSession\_Id PRIMARY KEY (Id),  CONSTRAINT FK\_PrintSession\_DeviceInstallationId FOREIGN KEY (DeviceInstallationId) REFERENCES DeviceInstallation (Id) ON DELETE CASCADE,  CONSTRAINT FK\_PrintSession\_BranchId FOREIGN KEY (BranchId) REFERENCES Branch (Id) ON DELETE CASCADE,  CONSTRAINT CK\_PrintSession\_PrintablePages CHECK(PrintablePages > 0)  ) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Ограничения:

-

-

-

CREATE TABLE PrintingDevice (

Id INT IDENTITY,

Name NVARCHAR(100),

ConnectionType VARCHAR(15),

MACAdress CHAR(12),

CONSTRAINT PK\_PrintingDevice\_Id PRIMARY KEY (Id),

CONSTRAINT UQ\_PrintingDevice\_MACAdress UNIQUE (MACAdress)

)

CREATE TABLE Branch (

Id INT IDENTITY,

PrintingDeviceId INT,

Number INT,

IsDefault VARCHAR(10),

CONSTRAINT PK\_Branch\_Id PRIMARY KEY (Id),

CONSTRAINT FK\_Branch\_PrintingDeviceId FOREIGN KEY (PrintingDeviceId) REFERENCES PrintingDevice (Id) ON DELETE CASCADE,

CONSTRAINT CK\_Branch\_Number CHECK(Number > 0)

)

CREATE TABLE DeviceInstallation (

Id INT IDENTITY,

Name NVARCHAR(100),

Location NVARCHAR(100),

CONSTRAINT PK\_DeviceInstallation\_Id PRIMARY KEY (Id)

)

CREATE TABLE PrintSession (

Id INT IDENTITY,

DeviceInstallationId INT,

Employee NVARCHAR(100),

BranchId INT,

PrintablePages INT,

Status VARCHAR(10),

CONSTRAINT PK\_PrintSession\_Id PRIMARY KEY (Id),

CONSTRAINT FK\_PrintSession\_DeviceInstallationId FOREIGN KEY (DeviceInstallationId) REFERENCES DeviceInstallation (Id) ON DELETE CASCADE,

CONSTRAINT FK\_PrintSession\_BranchId FOREIGN KEY (BranchId) REFERENCES Branch (Id) ON DELETE CASCADE,

CONSTRAINT CK\_PrintSession\_PrintablePages CHECK(PrintablePages > 0)

)

**Проверка:**

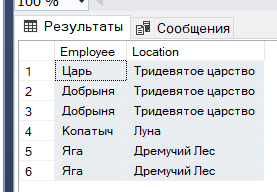
После завершения проектирования (если так можно сказать, учитывая, что уже было указано, какие именно поля и таблицы нужны), я подумал проверить результат. А именно попробовал получить те же самые таблицы, что были предоставлены в качестве первичных сведений.

Заполнение всех таблиц:

|  |
| --- |
| INSERT INTO PrintingDevice (Name, ConnectionType, MACAdress) VALUES  ('Папирус', 'Локальное', 093289230023),  ('Бумага', 'Локальное', 674323001143),  ('Камень', 'Сетевое', 234234453434)  INSERT INTO Branch (PrintingDeviceId, Number, IsDefault) VALUES  ( 1, 1, 'Да'),  ( 2, 2, 'Нет'),  ( 2, 3, 'Нет'),  ( 3, 1, 'Да'),  ( 2, 3, 'Нет'),  ( 1, 2, 'Да')  INSERT INTO DeviceInstallation (Name, Location) VALUES  ('Дворец', 'Тридевятое царство'),  ('Конюшни', 'Тридевятое царство'),  ('Оружейная', 'Тридевятое царство'),  ('Кратер', 'Луна'),  ('Избушка', 'Дремучий Лес'),  ('Топи', 'Дремучий Лес')  INSERT INTO PrintSession (DeviceInstallationId, Employee, BranchId, PrintablePages, Status) VALUES  ( 1, 'Царь', 1, 4, 'Успех'),  ( 2, 'Добрыня', 2, 7, 'Неудача'),  ( 3, 'Добрыня', 3, 5, 'Успех'),  ( 4, 'Копатыч', 4, 11, 'Неудача'),  ( 5, 'Яга', 5, 14, 'Неудача'),  ( 6, 'Яга', 6, 9, 'Успех') |

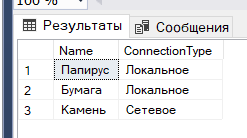
Проверка исходной таблицы сотрудников и локаций филиалов:

|  |
| --- |
| SELECT PrintSession.Employee, DeviceInstallation.Location  FROM PrintSession  INNER JOIN DeviceInstallation ON DeviceInstallation.Id = PrintSession.DeviceInstallationId |



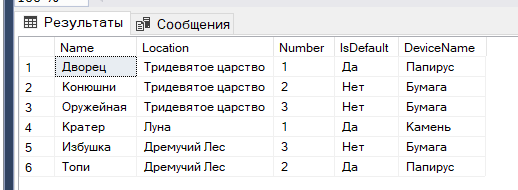
Проверка таблицы с устройствами печати:

|  |
| --- |
| SELECT Name, ConnectionType FROM PrintingDevice |



Проверка таблицы со сведения об использовании устройств в филиалах компании:

|  |
| --- |
| SELECT DeviceInstallation.Name AS Name,  DeviceInstallation.Location AS Location,  Branch.Number AS Number,  Branch.IsDefault AS IsDefault,  PrintingDevice.Name AS DeviceName  FROM PrintSession  INNER JOIN DeviceInstallation ON DeviceInstallation.Id = PrintSession.DeviceInstallationId  INNER JOIN Branch ON Branch.Id = PrintSession.BranchId  INNER JOIN PrintingDevice ON PrintingDevice.Id = Branch.PrintingDeviceId |



# Вывод справочников