



MICROSOFT ACCESS 2016

Дефиниране на таблици и
свойства на полетата им.
Връзки между таблиците в
релационния модел

Създаване на таблица



Раздел **Create** ->

- **Table** – таблицата се създава в изглед **Datasheet View** и може да се започне с директно въвеждане на данните;
- **Table Design** – таблицата се създава в изглед **Design View**, който позволява първо да се дефинира нейната структура.



Създаване на таблицата в режим Design View

- Описание на полетата
 - Име на полето – **Field Name** (до 64 знака);
 - Тип на данните – **Data Type**;
 - Описание – **Description**;
 - Свойства на полето – **Field Properties**
(страници General и Lookup);
- Задаване на **първичен ключ (Primary Key)**;
- Съхраняване на таблицата в базата от данни с избрано име.

Дефиниране на полета



- **Името на полето** трябва да бъде съобразено със следните правила и препоръки:
 - максимална дължина – 64 символа;
 - не се разрешават дублирани имена на полета;
 - не трябва да се използват имена на вградени функции и свойства на обекти;
 - да се избягват дългите имена;
 - не е разрешено използването на символите **точка, квадратни скоби и удивителен знак.**

Дефиниране на полета



- името не може да започва с интервал;
- въпреки, че вътре в името **интервалът** е разрешен символ, е препоръчително да се избягва;
- препоръчително е имената на полетата да са мнемонично насочени към предназначението на полето;
- полетата, които се срещат в повече от 1 таблици, е добре да са навсякъде с еднакви имена.

Тип на данните и свойства на полето

- **Типът на данните** на полето определя важни негови качества, като:
 - какви стойности може да приема полето;
 - максималния размер на стойността на полето;
 - как полето може да се използва в изрази;
 - кои формати могат да се използват за показване на съдържанието на полето;
- **Свойствата на полето** определят характеристиките и поведението на данните, съхранявани в полето, съобразно избрания тип на данните.

Типове данни в MS Access



- **Short Text** – символни данни с размер до 255 символа;
- **Long Text** – символни данни до 65 535 символа;
- **Number** – числа (с размер 1,2,4 или 8 байта в зависимост от Field Size);
- **Date/Time** – дата и време (8 байта);
- **Currency** – валутни стойности (8 байта);
- **AutoNumber** – уникално цяло число, което се генерира автоматично от Access за всеки нов ред в таблицата. Такива стойности не могат да се променят и само 1 колона може да бъде от този тип.

Типове данни в MS Access



- **Calculated** – изчисляемо поле въз основа на други полета от таблицата;
- **HyperLink** – за хипервръзка (уеб документ, имейл);
- **Yes/No** – логически (булеви) данни (Да/Не; Yes/No; True/False);
- **OLE Object** – обекти от друго приложение (картини, графики, таблици и други);
- **Attachment** – прикрепване на файл, създаден с приложенията на MS Office.

Свойства на полетата (Field Properties)



❖ **Field Size** – размер на полето

- за тип **Text** - от 0 до 255; по подразбиране 255;
- за данни от тип **Number**
 - **Byte** - 1 байт, цели числа от 0 до 255
 - **Integer** - 2 байта, цели числа
 - **Long Integer** - 4 байта, цели числа
 - **Decimal** - 16 байта, десетични числа
 - **Single** - 4 байта, с плаваща точка
 - **Double** - 8 байта, с плаваща точка

Свойства на полетата (Field Properties)



❖ **Format** – формат на представяне

- за данни тип **Number**, **Currency**
- **General Number** – (по подразбиране), числата се показват така, както са въведени
- **Currency** – с валутен знак и точност 0.01 (от настройките в Control Panel / Region and Language)
- **Euro** – с валутен знак € и точност 0.01
- **Fixed** – най-малко една цифра пред и две след десетичната запетая
- **Percent** – процент, например 23.00%

Свойства на полетата (Field Properties)



❖ **Format** – формат на представяне

- За данни тип **Data/Time**
- **General Date** – (по подразбиране), **30.10.2018 11:30:00**
- **Long Date** – в зависимост от настройките в Region and Language, **30 октомври 2018**
- **Medium Date** – **30 октомври 18**
- **Short Date** – **30.10.2018**
- **Short Time** – **11:30**

❖ **Decimal Places** – място на десетичната точка

Свойства на полетата (Field Properties)



- ❖ **Input Mask** – входна редактираща маска, която контролира броя и вида на въвежданите в полето стойности.
 - **0** – Цифра (**изисква се** въвеждане на символ, + и – не са позволени)
 - **9** – Цифра или интервал (**не** изисква въвеждане, без + и -)
 - **#** – Цифра или интервал (**не** се изисква въвеждане; интервалите се показват в Edit режим, но се махат като се даде Save; + и – не са позволени)
 - **L** – Буква, изисква се въвеждане
 - **?** – Буква, въвеждането е незадължително
 - **A** – Буква или цифра (изисква се въвеждане)

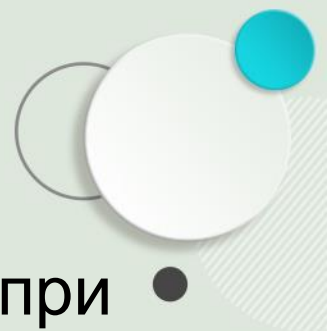
Примери за дефинирани шаблони



- **"+(359)"999"/"99\ 99\ 99! -> +(359)____/____ _ _ _**
- **"BG"9999999999999999 -> BG5671122357 (ИН по ДДС)**
- **за ЕГН 00000000000**
- **за регистрационен номер >?L\ 0000\ >L?**

Пример: CA 3619 PH

Свойства на полетата (Field Properties)



- ❖ **Caption** – синонимно име, което се използва при показване на данните от таблицата и замества името, зададено в Field Name

Например: Field Name е **ID_Customer**, а в Caption – **Номер клиент**

- ❖ **Default Value** – стойност по подразбиране, която се попълва автоматично, ако не се въведе стойност

Например: за колоната Дата на поръчка,
Default Value – Date()

Дефиниране на ограничения



- **Наборът от стойности**, които са **валидни** за дадена колона и дали да се разрешават неопределени стойности (NULL), се определя от **типа на данните** в колоната и допълнителни **ограничения**:
 - **Data Type**
 - Свойства в **General**:
 - **Field Size**
 - **Validation Rule**
 - **Required (Yes/No)**

Дефиниране на ограничения



- **Validation Rule** – правила за валидизация; чрез логически израз се задават ограничения, на които да отговаря съдържанието на колоната

Примери: за поле "оценка" ≥ 2 AND ≤ 6
за поле "цена" > 0

- **Validation Text** – съобщение, което се извежда, ако се нарушат правилата за валидизация
- **Required** – изисква задаване на стойност в полето, ако са избере Yes (**Not Null**); по подразбиране за всички, с изключение на колоната **PK** е No



Връзки между таблиците в релационния модел

Връзки между таблиците в БД



■ **Връзка** между таблиците в БД се осъществява на основата на **общи колони**, които:

- Имат един и същи смисъл
- Имат един и същи тип данни

Пример:

Authors (AuthorID, Author_Name)

Books (BookId, Title, Author, Publish_Year)

- Атрибут в една релация, който съответства на първичния или на друг възможен ключ в друга релация наричаме **външен ключ** (Foreign Key – FK)

Рекурсивен външен ключ



Рекурсивен външен ключ: външен ключ в релация, който съответства на първичния ключ в същата релация

Номер служител	Име	Номер отдел	Номер ръководител
14	Иван Стоянов	2	
16	Петя Живкова	3	14
17	Антон Павлов	5	14

Primary Key

Foreign Key

Изисквания към външния ключ

- Има **същия смисъл** като първичния ключ в другата релация;
- Има **същия тип на данните** както първичния ключ в другата релация;
- Може да приема стойности само измежду **съществуващите стойности** на кандидат ключа в другата релация или да остане **напълно неопределен**. Това изискване се нарича **референтна цялост**.

Интегритет (integrity) на данните

- *Дадена съвкупност от данни притежава свойството **интегритет (цялост)**, ако данните са логически съвместими, непротиворечиви и изпълняват определени ограничения.*
- Основни типове интегритет на данните
 - **Entity Integrity** (интегритет по същност)
 - **Domain Integrity** (домейн интегритет)
 - **Referential Integrity** (интегритет на връзките)

Интегритет по същност



- Всяка релация се идентифицира от **първичен ключ**, който удовлетворява изискванията към РК:
 - Уникалност – не може да приема дублиращи се стойности
 - Is Not Null - не може да приема неопределени стойности
 - Минималност

Интегритет на връзките

- Дефинира се за релации, между които съществува връзка на основата на общи колони.
- Определя какви стойности **може да приема** **ВЪНШНИЯТ КЛЮЧ** при **добавяне на нови редове** в релацията.
- Налага определени правила при
 - **изтриване** на редове от релацията, към която се обръща;
 - **промяна** стойността на **ключа**, към който се обръща.

Домейн интегритет (Domain Integrity)



- Дефинира се за отделни атрибути на релацията
- Специфицира **набор от стойности**, които са валидни за дадена колона и дали се разрешава атрибутът да приема неопределени стойности
- Наборът от допустими стойности се дефинира от типа на данните и зададените свойства на колоната – размер (Field size), формат (Format), стойност по подразбиране (Default value), Required и др.

Типове връзки

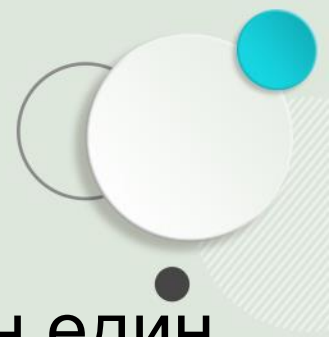
- **Едно : Едно (1:1)**
- **Едно : Много (1:M)**
- **Много : Едно (M:1)**
- **Много : Много (M:M)**



Типове връзки

- За две таблици **A** и **B** типът се определя като се отговори на следните **2 въпроса**:
 - Един ред от **A** с колко реда от **B** се свързва?
 - Един ред от **B** с колко реда от **A** се свързва?
- Ако отговорът на първия въпрос е “**много**”, а на втория “**един**”, то типът на връзката от **A** към **B** е **1:M**
- Ако отговорът и на двата въпроса е “**много**”, то типът на връзката от **A** към **B** е **M:M**

Примери за типове връзки



- „По всяка оферта може да бъде съставен един договор; всеки договор се съставя само въз основа на една оферта.“

Оферти:Договори = 1:1

- „Отделът има **много** служители; всеки служител работи само към **един** отдел.“

Отдели:Служители = 1:M

- „В една фактура могат да бъдат описани много продукти; един продукт може да присъства в няколко фактури.“

Фактури:Продукти= M:M

Представяне на връзки от тип 1:1



- Връзка от тип 1:1 се представя чрез включването на **първичния ключ** на едната таблица като **колона** (външен ключ) в **другата таблица**
- Пример: **Стоки : Наличности на стоки (1:1)**
Стоки (код, име, ед. мярка, цена)
Наличности на стоки (код, налично к-во)

Представяне на връзки от тип 1:M



- Връзка от тип 1:M се представя чрез включването на **първичния ключ** на таблицата към която **връзката е 1** като **колона** (външен ключ) в другата таблица, към която **връзката е M**.
- Пример: **Отдели: Служители** (1:M)

Отдели (код, наименование, ръководител)

Служители (номер-служ., име, ..., код-отдел)

Представяне на връзки от тип M:M

- Връзка от тип **M:M** между 2 таблици **A** и **B** се представя чрез създаване на нова таблица **C**, която свързва двете таблици и има общи колони с всяка от тях, т.е. включва **първичните ключове** на **A** и **B**, които се явяват **външни ключове** за **C**.

Фактури (№ фактура, ...) Стоки (код стока, ...)

Детайли (№ фактура, код стока, количество)

Представяне на връзки от тип M:M



- В новата таблица Детайли, възможности за РК:
 - **съставен РК** от (**№ фактура, код стока**):
Детайли (№ фактура, код стока, количество)
 - въвеждане на **изкуствен ключ**:
Детайли (№ детайл, № фактура, код стока, к-во)