**Предметна област**

“TeddyPlay” е лидер в продажбите на лицензирани игри. Открита през 2001 г. в град София. Той има много магазини в цялата страна, с широка гама от игри от различни жанрове.

При продажбата на игри клиентът трябва да предостави кратка информация на себе си. Първо, представител на магазина за да се свържи с него в случай на проблеми. Второ, за да се предотврати незаконното разпространение на игри. Магазинът събира такива клиентски данни като име, адрес, град, email и телефонен номер на клиента. Освен това всеки клиент се инициализира с уникален ID за автоматизиране и ускоряване на работата в бъдеще, ако клиентът пристигне отново.

За всеки служител се запазват ЕГН, неговата длъжност, име, град, адрес, email, телефонен номер, заплата. Няколко служители могат да заемат една и съща длъжност.

Що се отнася до продуктите, всеки магазин съхранява информация за името на продукта (име на играта), единична цена, брой копия и жанр на играта. Може да има няколко игри в един и същ жанр едновременно.

Когато се случи продажба, магазинът съхранява информация за клиента, служителя, който извършил продажбата, и точното време, когато е извършена продажбата. Всяка продажба се инициализира със специален ID.

**Бизнес правила**

Един клиент може да направи повече от една поръчка. Една поръчка може да бъде направена само от един клиент;

Един служител може да направи повече от една продажба. Една продаюба може да бъде направена само от един служител;

Една игра може да участва в повече от една продажба. Една продажба може да съдържа повече от една игра;

Един служител може да работи само в една длъжност. Една длъжност може да принадлежи на няколко служители;

Една игра може да принадлежи на няколко жанра. Един жанр може да съдържа повече от една игра;

**"Таблица 1."**

**Начален E-R модел**

**1 М**



**М М**

1. **М** 

* **Клиенти**

ID\_Клиент

Име

Адрес

Град

Email

Телефон

* **Служители**

ЕГН

Име

Град

Адрес

Email

Телефон

* **Продажби**

ID\_Продажба

ЕГН\_Служител

ID\_Клиент

Дата\_продажба

* **Стоки**

ID\_Стока

Име

Ед.цена

Наличност

**Усъвършенстван E-R модел**

Една игра може да участва в повече от една продажба. Една продажба може да съдържа повече от една игра; *Грешка: връзка M:M невъзможна – създаваме нова таблица.*

Един служител може да работи само в една длъжност. Една длъжност може да принадлежи на няколко служители; *Създаваме нова таблица.*

Една игра може да принадлежи на няколко жанра. Един жанр може да съдържа повече от една игра; *Създаваме нова таблица.*

"**Таблица 2.",**

**1 М М**



**1**

**М** 

**М**

**1 1** 



**М** **М**

**1** **1**

1. **1**

* **Клиенти**

ID\_Клиент

Име

Адрес

Град

Email

Телефон

* **Служители**

ЕГН

ID\_Длъжност

Име

Град

Адрес

Email

Телефон

* **Продажби**

ID\_Продажба

ЕГН\_Служител

ID\_Клиент

Дата\_продажба

* **Стоки**

ID\_Стока

ID\_Категория

Име

Ед.цена

Наличност

* **Детали продажби**

ID\_Продажба

ID\_Стока

Количество

Цена

* **Длъжности**

ID\_Длъжност

Име

Заплата

* **Категории**

ID\_Категория

Име

**Преобразуване на E-R модел в релационен модел и връзки между релациите**

При трансформиране на концептуалния E–R модел в релационен модел могат да бъдат очертани следните стъпки и правила:

* базата от данни, която е представена като E–R диаграма съдържаща същности, може да бъде представена като съвкупност от таблици, представляващи релационен модел на базата от данни;
* атрибутите на типа същност се преобразуват директно в колони на таблицата;
* подсхемата (дефинирана потребителска представа като част от модела) са изразени чрез изрази директно;
* идентификаторът (първичният ключ) на типа същност става първичен ключ на таблицата;
* установените правила за интегритет на данни се свързват с колоните на таблиците;
* за връзки от типа 1:1 или 1:М не се създават нови таблици. Те се представят чрез механизма на външните ключове. Ако връзката е от типа 1:1, първичния ключ на единия тип същност се включва като колона (външен ключ) в таблицата, съответстваща на другия тип същност. Ако връзката е от типа 1:М, първичния ключ на типа същност, към която връзката е 1, се добавя като колона (външен ключ) в таблицата, съответстваща на типа същност, към която връзката е М;
* ако в E–R модела са останали връзки от типа М:М, то е необходимо да се създаде свързваща релация, в която да се включват първичните ключове на релациите, съответстващи на двата типа същности.
* Връзките се характеризират със степен, кардиналност, насоченост и зависимост.

Степента на връзката се определя от броя на релациите, с които е асоциирана. Тъй като в релационния модел не може да се използват комплексни (n – арни) връзки в нашата релационна база от данни връзките между таблиците са бинарни.

Кардиналността класифицира връзките между типовете същности като измерва колко същности от единия тип същност се свързват с една същност от същност. Може да бъде 1:1, 1:М, М:1, М:М (отпада при релационния модел).

Насочеността на връзката определя пораждащия тип релация в бинарна връзка:

* релация „родител” (пораждаща релация) – релацията, от която започва връзката;
* релация „дете” (породена релация) – релацията, където завършва връзката.

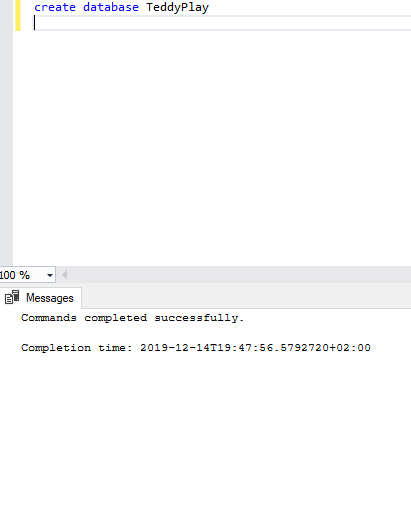
Посоката на връзката се определя от тяхната кардиналност. Във връзки 1:М, родителят е винаги релацията, към която, кардиналността е 1. Във връзки от типа 1:1 посоката е от независимия към зависимия тип същност. Когато съществуването на една релация зависи от съществуването на друга говорим за определящи и зависими релации.

**Интегритет на данните**

Всички връзки в базата данни притежават свойството Enforce Referential Integrity, което осигурява цялостност, интегрираност и съвместимост на данните. По този начин потребителят не може да въведе данни, в свързани релации, които не си съответстват по тип и стойности. Така се осигурява съвместимостта между първични ключове и външни ключове.

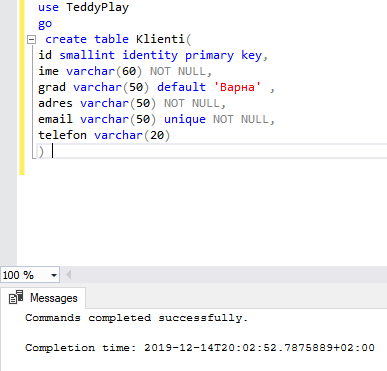
**Скриптове SQL за създаване на базата от данни и дефиниране на таблиците**

* **Създаване на БД**

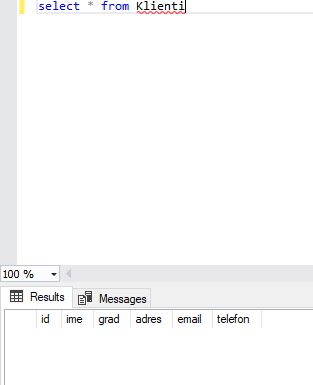
****

**"Фиг. 1. База данни".**

* **Таблица Клиенти**

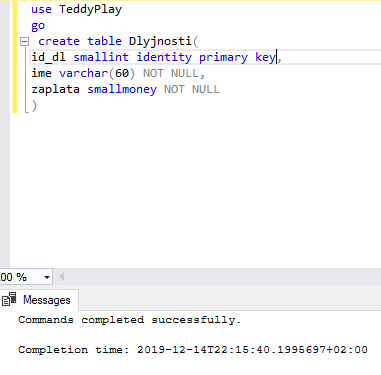


**"Фиг. 2. Клиент".**

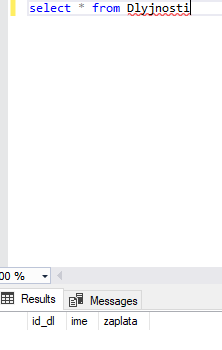


**"Фиг. 3. Резултат на създаване".**

* **Таблица Длъжности**

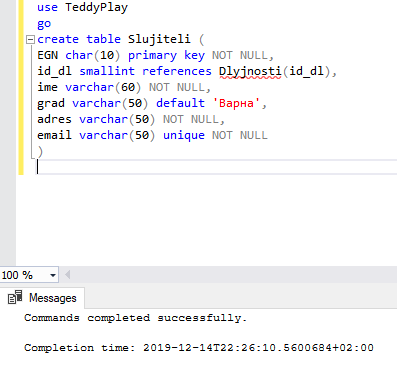


**"Фиг. 4. Длъжности".**

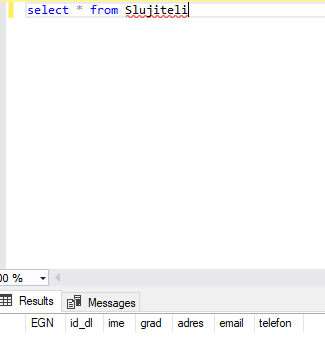


**"Фиг. 5. Резултат на създаване".**

* **Таблица Служители**

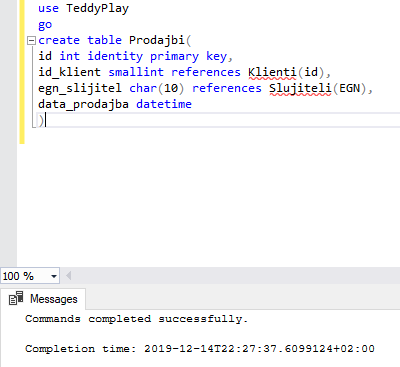


**"Фиг. 6. Служители".**

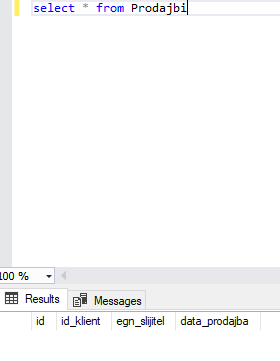


**"Фиг. 7. Резултат на създаване".**

* **Таблица Продажби**

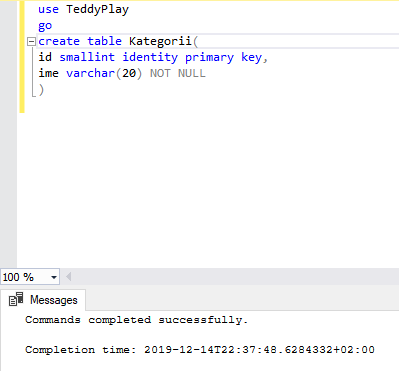


**"Фиг. 8. Продажби".**

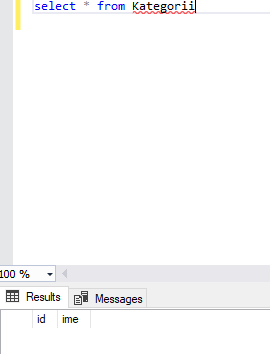


**"Фиг. 9. Резултат на създаване".**

* **Таблица Категории**

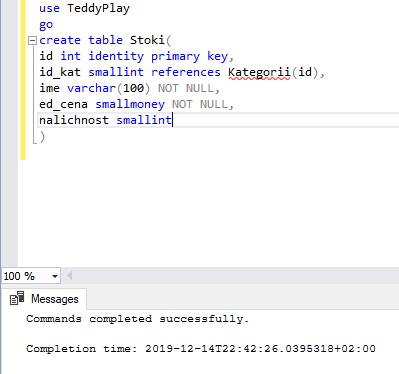


**"Фиг. 10. Категории".**

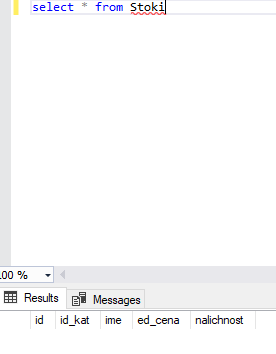


**"Фиг. 11. Резултат на създаване".**

* **Таблица Стоки**

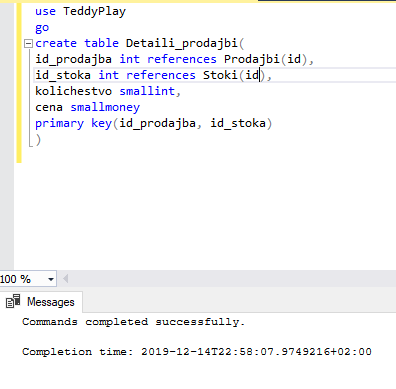


**"Фиг. 12. Стоки".**

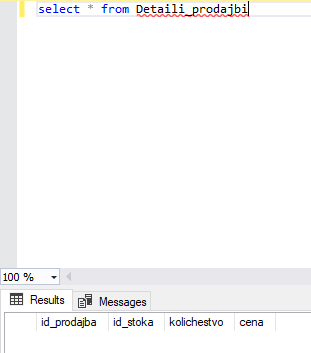


**"Фиг. 13. Резултат на създаване".**

* **Таблица Детайли продажба**

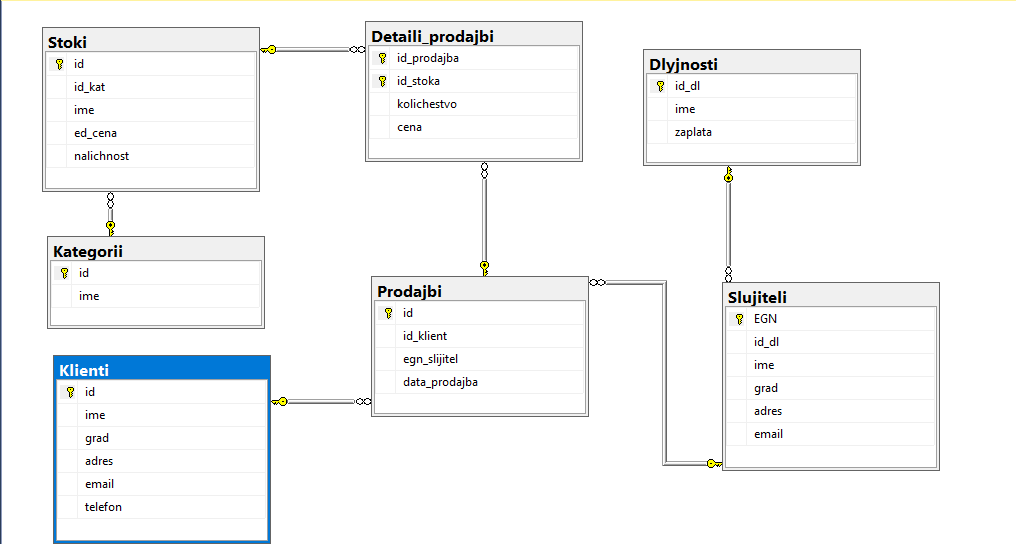


**"Фиг. 14. Детайли продажби".**



**"Фиг. 15. Резултат на създаване".**

* **Диаграма за връзките**

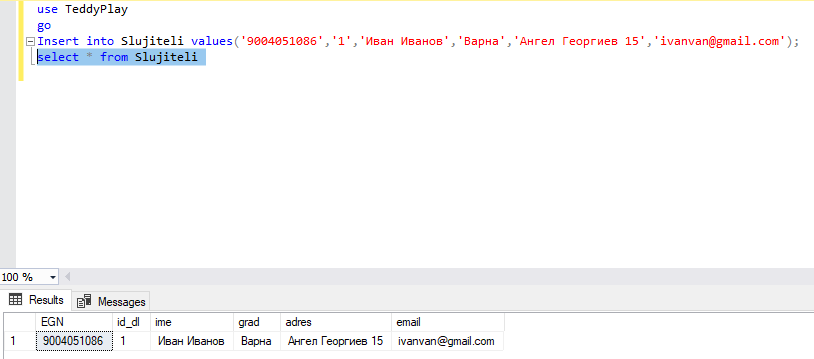


**"Фиг. 16. Диаграма".**

* **Например, по този начин се попълват таблиците Длъжности и Служители**

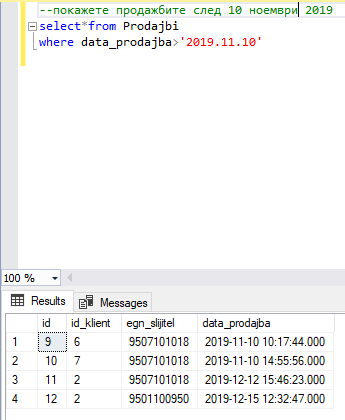


**"Фиг. 17. Попълване на таблиците".**

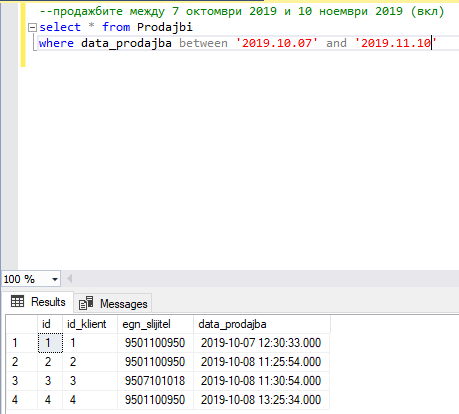


**"Фиг. 17.1. Попълване на таблиците".**

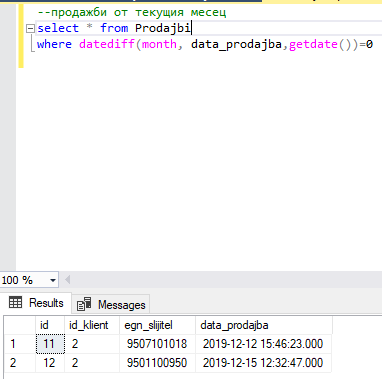
**SQL СКРИПТОВЕ ЗА МАНИПУЛИРАНЕ НА ДАННИ ОТ БАЗАТА ОТ ДАННИ**



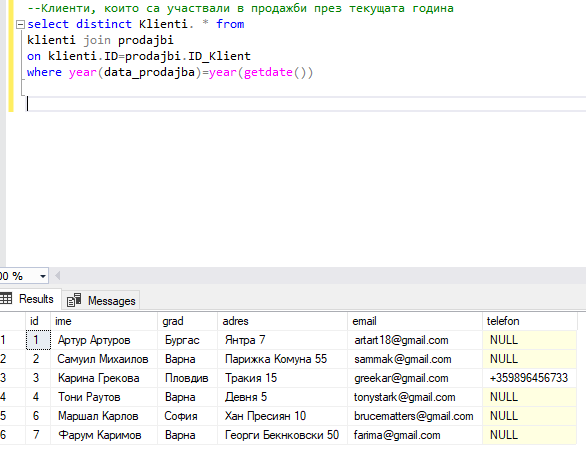
**"Фиг. 18. Манипулиране на таблица продажби".**

****

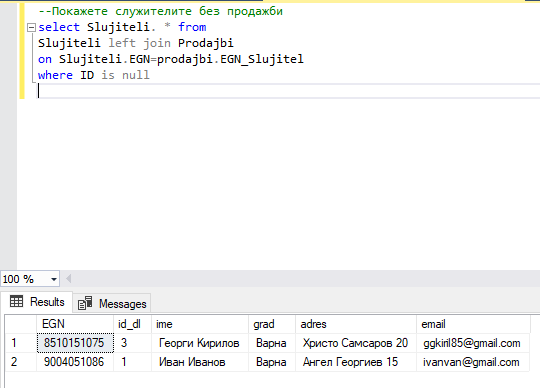
**"Фиг. 18.1. Манипулиране на таблица продажби".**

****

**"Фиг. 18.2. Манипулиране на таблица продажби".**

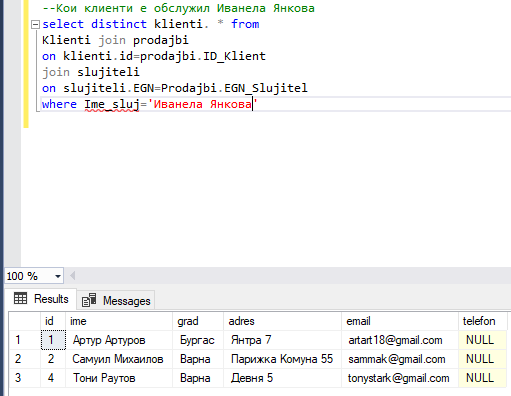
****

**"Фиг. 19. Манипулиране на таблиците през JOIN".**

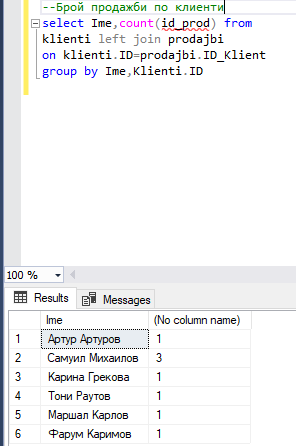
****

**"Фиг. 19.1. Манипулиране на таблиците през JOIN".**

**Тука променихме “ime” на “ime\_sluj”, щото имаше конфликт на стойностите на таблиците**

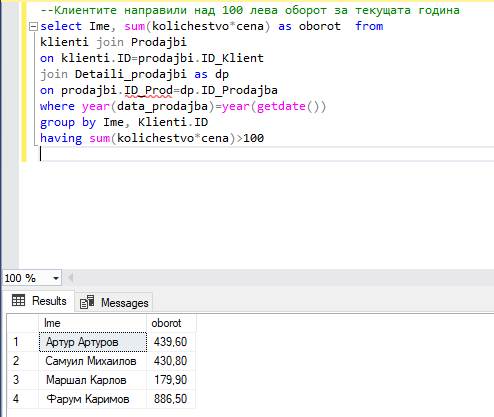


**"Фиг. 19.2. Манипулиране на таблиците през JOIN".**

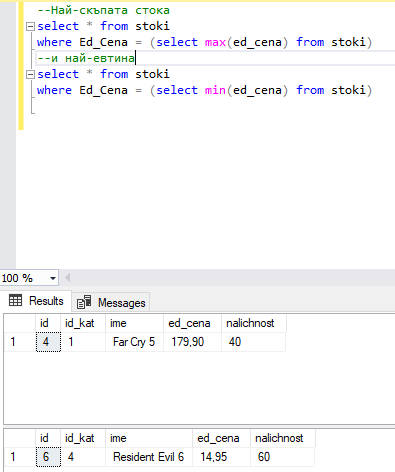
****

**"Фиг. 19.3. Манипулиране на таблиците през JOIN".**

**Тука променихме “id” на “id\_prod”, щото имаше конфликт на стойностите на таблиците**

****

**"Фиг. 20. Манипулиране на таблиците през JOIN, GROUP BY,HAVING".**

****

**"Фиг. 21. Манипулиране на таблиците през MIN, MAX".**