

# 8. Външни ключове

Лекционен курс “Бази от данни”

## Въвеждащ пример

REGION_ID	REGION_NAME	POPULATION
1	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000

Нека имаме таблица в нашата БД, съдържаща географски региони.

Искаме да разширим БД с нова таблица, която да съдържа държавите.

COUNTRY_ID	COUNTRY_NAME
AR	Argentina
BE	Belgium
BG	Bulgaria
US	United States of America
CN	China



Как да запишем  
кой държава в кой  
регион се намира?

COUNTRY_ID	COUNTRY_NAME	REGION	POPULATION
AR	Argentina	Americas	500 000 000
BE	Belgium	Western Europe	1 000 000 000
BG	Bulgaria	Eastern Europe	2 500 000 000
US	United States of America	Americas	1 000 000 000
GR	Greece		

Дублиране на информация?



REGION_ID	REGION_NAME	POPULATION
1	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000

COUNTRY_ID	COUNTRY_NAME	REGION	POPULATION
AR	Argentina	Americas	500 000 000
BE	Belgium	Western Europe	1 000 000 000
BG	Bulgaria	Eastern Europe	2 500 000 000
US	United States of America	Americas	1 000 000 000
GR	Greece		

---

---

---

---

---

---

---

---

Решение



REGION_ID	REGION_NAME	POPULATION
1	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000



COUNTRY_ID	COUNTRY_NAME	REGION_ID
AR	Argentina	2
BE	Belgium	5
BG	Bulgaria	1
US	United States of America	2
GR	Greece	

---

---

---

---

---

---

---

---

Да разгледаме атрибута REGION\_ID от релацията COUNTRIES:

- Една дадена стойност за този атрибут (напр. 1) може да се появи, само ако тази стойност се появява като стойност на първичния ключ REGION\_ID в релацията REGIONS;
- В противен случай БД не притежава свойството цялостност (референциална цялостност).

---

---

---

---

---

---

---

---

Защо липсва  
свойството  
цялостност?



REGION_ID	REGION_NAME	POPULATION
1	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000

COUNTRY_ID	COUNTRY_NAME	REGION_ID
AR	Argentina	2
BE	Belgium	5
BG	Bulgaria	1
US	United States of America	2
GR	Greece	6

Определение

**Външен ключ:** нека R е базова релация. Външен ключ (ВК) в R е едно подмножество на множеството от атрибутите  $BK = \{A_1^R, A_2^R, \dots, A_n^R\}$  така, че:

- съществува една базова релация  $R_1$  с ключ-кандидат;
- във всеки един момент стойностите на ВК в R са равни на стойности на КК в някой запис на  $R_1$ .

Важно е да се отбележи, че не е задължително R и  $R_1$  да са различни релации.

Пример

За да разберем дали атрибута REGION\_ID от релацията COUNTRIES отговаря на условията за външен ключ трябва да проверим дали отговаря на условията от дефиницията.

1. Той очевидно е подмножество от атрибутите на релацията.

COUNTRY_ID	COUNTRY_NAME	REGION_ID
AR	Argentina	2
BE	Belgium	5
BG	Bulgaria	1
US	United States of America	2
GR	Greece	

Пример

2. Съществува релация REGIONS с ключ-кандидат REGION\_ID и стойностите на атрибута COUNTRIES(REGION\_ID) са измежду стойностите на атрибута REGIONS(REGION\_ID).

REGION_ID	REGION_NAME	POPULATION
1	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000

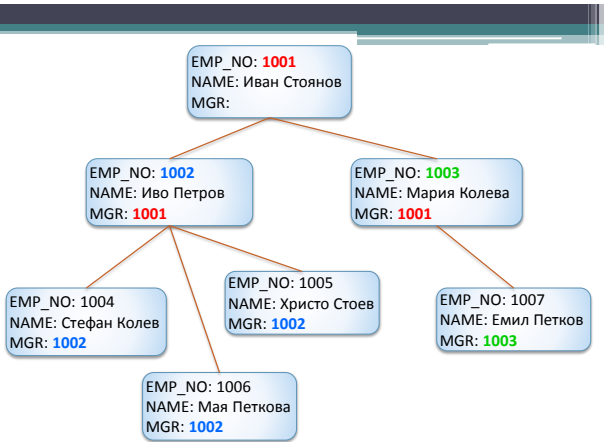
COUNTRY_ID	COUNTRY_NAME	REGION_ID
AR	Argentina	2
BE	Belgium	5
BG	Bulgaria	1
US	United States of America	2
GR	Greece	

Следователно в релацията COUNTRIES атрибутът REGION\_ID отговаря на условията за външен ключ.

В дефиницията беше подчертано, че двете релации R и R<sub>1</sub> не е необходимо да се различават.

Като пример за това можем да споменем релацията EMPLOYEE, в която се съхраняват данните за служителите и началниците на определена компания.

Това е типична дървовидна структура, която може да бъде съхранена в таблица.



EMPLOYEE

EMP_NO	NAME	SALARY	MGR
1001	Иван Стоянов	800	
1002	Иво Петров	630	1001
1003	Мария Колева	600	1001
1004	Стефан Колев	430	1002
1005	Христо Стоев	450	1002
1006	Мая Петкова	390	1002
1007	Емил Петков	300	1003

Тук EMP\_NO е първичният ключ, а MGR е външният ключ, който референцира първичния ключ на същата релация EMPLOYEE.

Логиката е следната – служителът с номер 1001 е и мениджър на служителите с номера 1002 и 1003.

Свойства на външните ключове

По дадената дефиниция можем да характеризираме външните ключове както следва:

- ВК са също множества от атрибути;
- По дефиниция всяка стойност на даден ВК във всеки един момент трябва да се появява като стойност на съответния референциран първичен ключ;
- Обратното не се изисква - т.е. КК може да има стойност, която не се появява във ВК;
- Един ВК ще бъде съставен, ако и само ако кореспондиращият КК е също съставен;

- Ако КК е прост - ВК също ще е прост;
- Всеки компонент на един ВК трябва да бъде дефиниран върху същия домейн, върху който е дефиниран кореспондиращият компонент от КК;
- Не се изисква ВК да е съставна част на ПК или на някой КК;
- Една стойност на ВК представлява връзка (референция) към n-торката, съдържаща съответстващата стойност на ПК.

Референциална цялостност: проблемът за осигуряване, че БД не съдържа невалидни стойности на ВК.

Референциално ограничение: ограничението, че стойностите на един ВК трябва да съответстват на стойностите на съответния ПК.

Релация:

- референцираща - релацията, която съдържа ВК
- референцирана (целева) - релацията на ПК

---

---

---

---

---

---

---

Референциални диаграми

Нека разгледаме таблиците за държави и региони отново. Можем да представим референциалните ограничения чрез следната референциална диаграма:

REGION\_ID  
**COUNTRIES → REGIONS**

Всяка стрелка показва наличие на външен ключ в релацията, от която тръгва, референциращ съответен първичен ключ в релацията, към която сочи. Добра идея е да се указват атрибутите, съставлящи външния ключ.

---

---

---

---

---

---

---

Референциалните диаграми се използват за представяне на референциални ограничения.

Една и съща релация може да бъде едновременно референцираща и референцирана:

**R3 → R2 → R1**

---

---

---

---

---

---

---

Референциален път



$$R_n \rightarrow R_{n-1} \rightarrow \dots \rightarrow R_2 \rightarrow R_1$$

Нека релациите от  $R_n$  до  $R_1$  са такива, че има референциално ограничение от  $R_n$  към  $R_{n-1}$ , от  $R_{n-1}$  към  $R_{n-2}$  и т.н. до  $R_1$ .

Последователността от референциални ограничения (стрелките) от  $R_n$  до  $R_1$  представя **референциален път** от  $R_n$  до  $R_1$ .

---

---

---

---

---

---

---

$R_1$  и  $R_2$  не е задължително да са различни, т.е. може да са еквивалентни. Една такава релация се нарича самореференцираща се.

Напр. релацията EMPLOYEE:

**EMPLOYEE (EMP\_NO, ..., EMP\_MGR)**  
**PRIMARY KEY(EMP\_NO),**  
**FOREIGN KEY(EMP\_MGR)**  
**REFERENCES EMPLOYEE**

---

---

---

---

---

---

---

Референциален цикъл



$$R_n \rightarrow R_{n-1} \rightarrow \dots \rightarrow R_2 \rightarrow R_1$$

Нека релациите от  $R_n$  до  $R_1$  са такива, че има референциално ограничение от  $R_n$  към  $R_{n-1}$ , от  $R_{n-1}$  към  $R_{n-2}$  и т.н. до  $R_1$ , като  $R_1$  също има външен ключ към  $R_n$ . Така представени, релациите формират **референциален цикъл**.

Самореференциращите се релации са частен случай на референциален цикъл.

По-точно, референциален цикъл съществува, ако има референциален път от  $R_n$  до  $R_n$ .

---

---

---

---

---

---

---

- Понякога се казва, че външен-към-кандидат-ключ съответствията са „връзките“, които държат базата данни „цяла“;
- Казано по друг начин, те представят взаимоотношения между отделните записи;
- Важно е тук да се отбележи, че не всички взаимоотношения се представят с такива съответствия. Напр. може да имаме взаимоотношение “съгражданство” между доставчици и детайли, представено от атрибутите CITY в двете релации, без да е налице външен ключ.

---

---

---

---

---

---

---

### Спецификация на ВК

FOREIGN KEY (element-list)  
REFERENCES base-relation

---

---

---

---

---

---

---

### Правила за ВК

Правило за референциалната цялостност: БД не трябва да съдържа стойности на ВК, които нямат съответствие - т.е. няма съществуваща стойност за ПК в целевата релация.

Всяко състояние на БД, което не удовлетворява правилото за референциалната цялостност, по дефиниция е некоректно.

Но правилото не казва как да се предпазим от такива некоректни състояния.

Възможности:

- Системата да отхвърли всяка операция, която би довела до некоректно състояние, ако бъде изпълнена;
- Системата извършва операцията, а след това извършва допълнителни компенсиращи операции.

---

---

---

---

---

---

---



За всеки ВК съществуват два основни въпроса, на които трябва да се отговори:

- 1. Какво ще се случи при опит да се изтрие целевата на ВК референция?  
Например при опит да се изтрие регион, за който съществува поне една държава.

Съществуват два подхода:

- **RESTRICTED** - операцията изтриване е ограничена само за случая, когато няма свързани данни; ако такива са налице, изтриване не се извършва;
- **CASCADES** - операцията се разширява каскадно и изтрива също така и свързаните записи.

RESTRICTED

Операцията се отхвърля

REGION_ID	REGION_NAME	POPULATION
1	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000

COUNTRY_ID	COUNTRY_NAME	REGION_ID
AR	Argentina	2
BE	Belgium	5
BG	Bulgaria	1
US	United States of America	2
GR	Greece	1

CASCADES



REGION_ID	REGION_NAME	POPULATION
1	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000

COUNTRY_ID	COUNTRY_NAME	REGION_ID
AR	Argentina	2
BE	Belgium	5
BG	Bulgaria	1
US	United States of America	2
GR	Greece	1

Операцията се разширява каскадно

2. Какво ще стане при опит да се промени един ПК, който е целева референция на ВК?  
Например - опит да се промени идентификатор на регион, за който съществува поне една държава.

Отново се използват същите два подхода:

- **RESTRICTED** - операцията промяна е ограничена само за случая, когато няма държава за региона; ако такава е налице, промяна не се извършва;
- **CASCADES** - операцията се разширява каскадно и променя също така и записа за държавата.

RESTRICTED

Операцията се отхвърля

REGION_ID	REGION_NAME	POPULATION
100	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000

COUNTRY_ID	COUNTRY_NAME	REGION_ID
AR	Argentina	2
BE	Belgium	5
BG	Bulgaria	1
US	United States of America	2
GR	Greece	1

CASCADES

REGION_ID	REGION_NAME	POPULATION
100	Eastern Europe	500 000 000
2	Americas	1 000 000 000
3	Asia	2 500 000 000
4	Middle East and Africa	1 000 000 000
5	Western Europe	500 000 000

COUNTRY_ID	COUNTRY_NAME	REGION_ID
AR	Argentina	2
BE	Belgium	5
BG	Bulgaria	100
US	United States of America	2
GR	Greece	100

Операцията се разширява каскадно

## Разширение на синтаксиса на ВК

FOREIGN KEY ( ... ) REFERENCES base-relation

DELETE option

UPDATE option

option ::= RESTRICTED | CASCADES

## Забележки

Опциите за правилата DELETE и UPDATE за ВК не изчерпват възможностите - те по-точно представят най-общите, които се изискват от практиката.

По принцип съществуват повече възможности. Например, опит за изтриване на един регион възможностите могат да бъдат:

- инициализира се диалог с крайните потребители;
- информацията може да се архивира (какво ще се прави);
- държавата може да се причисли към друг регион.

## Внимание!

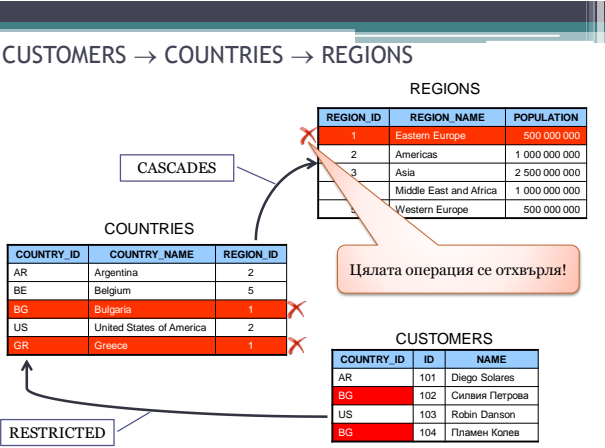
Нека  $R_2$  и  $R_1$  са референцираща и целева  $R_2 \rightarrow R_1$  и нека delete-правилото за това референциално ограничение е CASCADES.

Следователно изтриването на п-торка в  $R_1$  ще предизвика изтриване на п-торка в  $R_2$ .

Да допуснем, че  $R_2$  е целева за  $R_3$ :  $R_3 \rightarrow R_2 \rightarrow R_1$ . Тогава изтриването на п-торка зависи и от delete-правилото между  $R_3$  и  $R_2$ .

Ако то забранява изтриване, тогава не се изтрива нищо и БД остава непроменена.

И т.н. до произволно ниво.



---

---

---

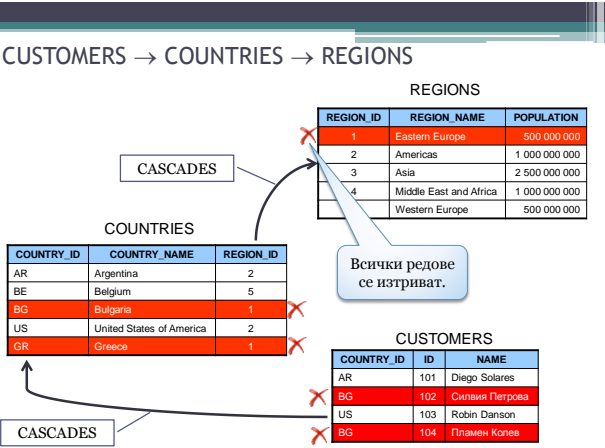
---

---

---

---

---



---

---

---

---

---

---

---

---