

Увод у релационе базе података

12а



Саша Малков
Универзитет у Београду
Математички факултет
2023/2024

[PM13]
Увод у РБП
Саша Малков



Тема 9

Физички модел базе података

[PM13] Увод у релационе базе података – Саша Малков – 2023/24 – час 12

1

Физички модел базе података



Физички модел базе података

- Физички модел је најнижи модел БП
 - Прави се у односу на дефинисан логички модел
- Описује конкретну имплементацију
 - Прилагођава логички модел условима имплементације и употребе
 - Фокусира се на техничке аспекте имплементације
 - Важну улогу има процена оптерећења
- Потребна су другачија знања и вештине
 - више техничка него пословна
- Физичко моделирање се често назива „моделирање података“
 - (енгл. *data modeling*)

Универзитет у Београду - Математички факултет

[PM13] Увод у релационе базе података – Саша Малков – 2023/24 – час 12

2

Физички модел базе података



Елементи физичког модела

- Елементи физичког модела се називају и *елементима физичке структуре* базе података
- Структуре података
 - Уобичајено је да се на физичком нивоу говори о табелама и колонама, а не о релацијама и атрибутима
- Интерна (физичка) организација података
 - простори за табеле
 - контејнери
 - странице
 - бафери
 - ...
- Помоћне компоненте
 - индекси

Универзитет у Београду - Математички факултет

[PM13] Увод у релационе базе података – Саша Малков – 2023/24 – час 12

3



Претпоставке физичког модела

- Физички модел мора да узима у обзир
 - потребе логичког модела
 - хардверске и софтверске могућности рачунарског система
 - процењено оптерећење базе података
- Улога процене оптерећења је суштинска
 - зато ћемо јој посветити посебну пажњу
- Процена оптерећења мора да утврди
 - под каквим је условима могуће остварити логички модел у пракси
 - какви су минимални и препоручени захтеви у односу на рачунарски систем
 - да ли постоје претпостављена ограничења рачунарског система
 - (ако се планира имплементација на постојећем рач. систему)
 - и какве су последице
 - који аспекти употребе представљају “уско грло” система
 - да ли је потребно да се предузима додатна оптимизација и каква



Процена оптерећења

- Процена оптерећења и перформанси обухвата
 - модел обраде података (енгл. *process model*)
 - неструктурне захтеве
 - матрицу ентитета и процеса (ПЧМБ)
 - захтеване перформансе
 - препознате циљне имплементације СУБП, ОС,...
 - препозната евентуална ограничења простора
 - препознате евентуалне развојне проблеме



Модел обраде података

- Разматрамо следеће околности
 - Околности додавања нових редова
 - Околности ажурирања постојећих редова
 - Околности брисања редова
 - Околности читања редова
- Разматрамо их
 - за сваку појединачну табелу (и поглед)
 - за групе повезаних табела
 - за целу базу података



Модел обраде података (2)

- Околности додавања нових редова
 - колико се редова додаје у просеку (нпр. дневно)
 - колико се редова додаје при највећем оптерећењу (нпр. у секунди)
 - карактеристике података који се додају
 - да ли су примарни кључеви нових редова међусобно слични?
 - да ли подаци зависе од времена додавања?
 - да ли се и које табеле често реферишу у додаваним редовима?
- ...



Модел обраде података (3)

- ...
- **Околности ажурирања постојећих редова**
 - колико се редова мења у просеку (нпр. дневно)
 - колико се редова мења при највећем оптерећењу (нпр. у секунди)
 - карактеристике података који се мењају
 - колика је вероватноћа да се редови са сличним ПК истовремено користе или мењају (због закључавања)
 - ...
- ...



Модел обраде података (4)

- ...
- **Околности брисања редова**
 - колико се редова брише у просеку (нпр. дневно)
 - колико се редова брише при највећем оптерећењу (нпр. у секунди)
 - карактеристике података који се бришу
 - да ли се бришу појединачно или у групама?
 - да ли обрисани редови реферишу сличне податке?
 - ...
- ...



Модел обраде података (5)

- ...
- **Околности читања редова**
 - учесталост читања
 - на нивоу табеле али и појединачних колона
 - колико редова се чита једним упитом
 - критеријуми одабира редова
 - које колоне се користе за одабир редова
 - које друге табеле се често користе заједно са посматраном
 - у подупима?
 - у спајањима?



Неструктурни захтеви

- Трајање података
 - колико дуго се подаци задржавају у табели пре брисања или архивирања?
- Обим података
 - колико ће редова бити у табели при пуштању у рад и како ће се број редова мењати током времена?
- Распоживост података
 - да ли су подаци потребни стално или повремено, колико често и дуго подаци могу/смеју да буду недоступни корисницима?
- Ажурност података
 - колико ажурни морају да буду подаци који се користе?
 - (да ли могу да се раздвоје копије за мењање и читање,...)
- Безбедносни захтеви



Апликативни захтеви

- Матрица ентитета и процеса
 - описује у којим се процесима користе које табеле и на који начин
- Евентуални развојни проблеми
 - да ли су потребни неки посебни развојни стручњаци
 - нпр. за сложене алгоритме, извештаје, анализе и сл.



Остале информације

- Захтеване перформансе
 - обично у облику времена одзива кориснику након задавања акције
- Циљне имплементације СУБД, ОС,...
 - не само производ, него и конкретна верзија
- Евентуална ограничења простора
 - да ли постоје потенцијални проблеми са складишним простором



Основни методи оптимизације БП

- Оптимизација на нивоу интерне организације података
 - не уводи се разлика између физичког и логичког модела
 - остварује се кроз управљање интерном организацијом података, помоћним компонентама и ресурсима
 - управља се појединосима имплементације табела
- Оптимизација на нивоу упита
 - не уводи се разлика између физичког и логичког модела
 - врши се писање упита на начин који омогућава њихово ефикасније извршавање
 - овај вид оптимизације постепено губи на значају
 - савремени СУБД имају добру аутоматску оптимизацију
- Оптимизација на нивоу структуре података
 - **уводи се** разлика између физичког и логичког модела
 - организација табела се мења у односу на логички модел



Основни методи оптимизације БП

- Данас ћемо се бавити првенствено елементима физичке структуре чијим се исправним конфигурисањем остварује оптимизација на нивоу интерне организације података
- Остале видове оптимизације остављамо за наредни час



Интерна организација података

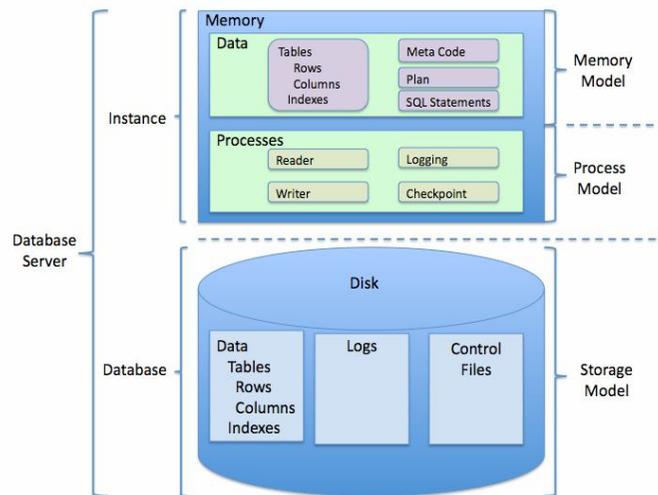
- Физичка организација података
 - простори за табеле, странице, бафери страница, партиције
- Управљање ресурсима
 - пре свега меморија
- Помоћне компоненте
 - индекси



Физичка организација података

- Логичка организација као основно место чувања података види релацију, тј. табелу
- Физичка организација иде и даље од тога и описује
 - начин записивања табела
 - начин приступања подацима
 - могућности управљања физичком организацијом
- СУБП обично омогућавају веома прецизно управљање елементима физичке организације података

Елементи физичке организације



IBM DB2

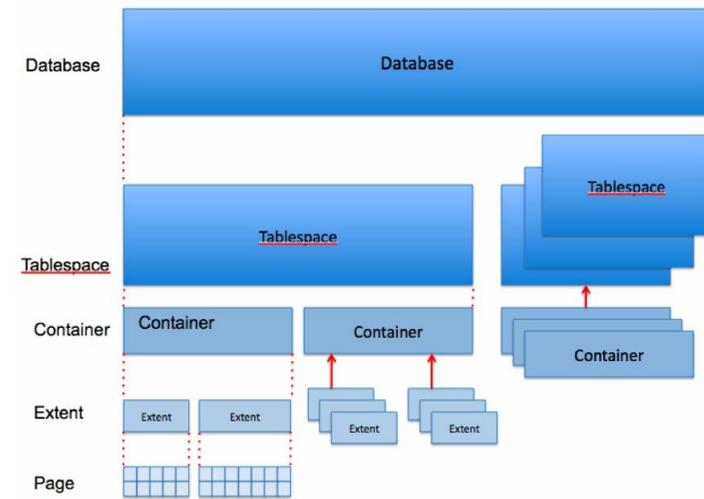
- Сваки СУБП има донекле специфичан начин физичке организације података и могућности управљања физичком организацијом података
- Елементе физичке организације релационих база података ћемо размотрити на примеру РСУПБ *IBM DB2*



Простор за табеле

- Основни складишни простор се назива *простор за табеле* (енгл. *table space*)
 - Свака база података мора да има један или више простора за табеле
 - Простор за табеле припада једној бази података
 - Један простор за табеле може да садржи више табела
 - У партиционисаним базама података једна табела може да буде и у више простора за табеле (хор. фрагментација)
- Простор за табеле представља *лоџички ниво* физичке организације
 - одређују се особине и начин употребе простора за табеле
 - не одређује се непосредно где се подаци налазе

Простор за табеле (2)



(wikimedia.org)



Простор за табеле (3)

- На нивоу простора за табеле дефинишу се:
 - величина физичке странице (енгл. *page size*, 4, 8, 16 или 32 KB)
 - величина јединице читања (енгл. *extent size*), обично у распону 2-256 страница
 - за ОЛТП се користе мање странице и јединице читања, а за складишта података веће странице и јединице читања
 - да ли се подаци чувају компримовано или не
 - да ли и како може да се аутоматски повећава
 - начин и услови баферисања страница
 - сваком простору за табеле се додељује тзв. бафер-пул
- Физичка локација складиштења се одређује као скуп *контејнера*
 - контејнер може да буде директоријум, партиција диска, диск, преалоциран фајл, али и тзв. *складишна група* (енгл. *storage group*)
 - складишна група предефинише скуп контејнера који затим може да се употребљава као целина
 - СУБП самостално одређује која страница које табеле ће бити у ком контејнеру



Прављење простора за табеле

```
CREATE [ LARGE | REGULAR | [ SYSTEM | USER ] TEMPORARY ] TABLESPACE <ime>
[ IN [ DATABASE PARTITION GROUP ] <ime grupe> ]
[ PAGESIZE <n> [ K ] ]
[ MANAGED BY AUTOMATIC STORAGE [ <group> <attrs> ] ]
[ MANAGED BY SYSTEM <kontejner> ]
[ MANAGED BY DATABASE <kontejner> <attrs> ]
[ EXTENTSIZE <int> [ K | M ] ]
[ BUFFERPOOL <ime> ]
...
```



Прављење простора за табеле (2)

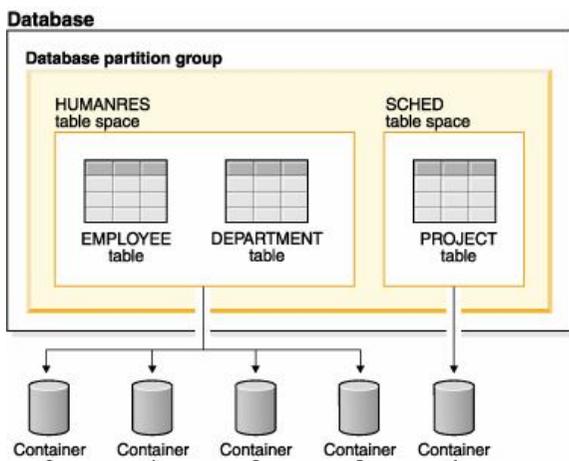
- На самом почетку се одређују врста и име простора за табеле
 - LARGE – општег типа, величине до 64TB, омогућава више од 255 редова по страници
 - REGULAR – општег типа, величине до 512GB
 - SYSTEM TEMPORARY – за системске привремене табеле
 - USER TEMPORARY – за корисничке привремене табеле
- Може да се бира група партиција у којој се прави
- Бира се величина странице у KB (4, 8, 16, 32)
- Опцијом EXTENTSIZE се бира величина јединице читања; задаје се бројем страница или величином у KB / MB
- Бира се начин старања о промени величине контејнера
 - Опција MANAGED BY AUTOMATIC STORAGE наглашава да се о промени величине контејнера стара СУБП (уз додатне опције)
 - Опција MANAGED BY SYSTEM наглашава да се о промени величине стара ОС (уз додатне опције)
 - Опција MANAGED BY SYSTEM наглашава да се о промени величине стара СУБП (у дефинисаним оквирима) или корисник мануелно



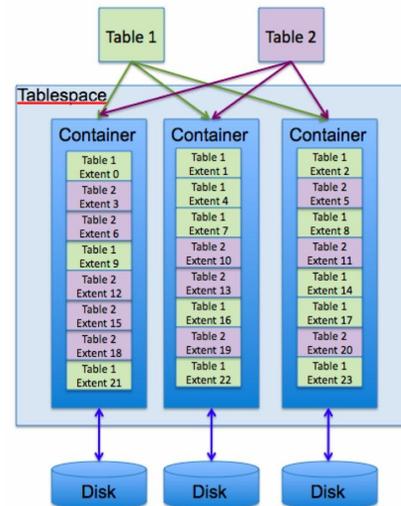
Контејнери

- Физичка локација складиштења је *контејнер*
 - Контејнер може да буде директоријум, партиција диска, диск, преалоциран фајл, али и тзв. *складишна група* (енгл. *storage group*)
 - складишна група предефинише скуп контејнера који затим може да се употребљава као целина
 - Простор за табеле се састоји од једног или више контејнера
 - Један контејнер припада само једном простору за табеле
 - СУБП самостално одређује која страница које табеле ће бити у ком контејнеру изабраног простора за табеле
 - странице се распоређују у групама, према величини јединице читања (*extent size*)

Контејнери (2)



Контејнери (3)





Прављење контејнера

- Контејнери се праве у оквиру наредби *CREATE TABLESPACE* и *ALTER TABLESPACE*
- У случају *"AUTOMATIC STORAGE"* праве се и одржавају аутоматски
- У случају *"MANAGED BY SYSTEM"* праве се тзв. *системски контејнери*:
 - контејнери се дефинишу као директоријуми са релативном или апсолутном путањом
 - *USING ('cont_path_1', 'cont_path_2',...)*
- У случају *"MANAGED BY DATABASE"* праве се тзв. *контејнери базе података*:
 - могу да буду
 - преалоцирани фајлови
 - уређаји (дискони или партиције)
 - *USING (FILE | DEVICE 'cont_string_1' n_of_pages, ...)*



Страница

- *Страница* је основни елемент физичког записа табеле
- све операције се физички одвијају над страницама
- увек се чита са диска или записује на диск цела страница
 - штавише, увек се читају групе страница, као *јединице за читање* (енгл. *extent*)
 - јединица писања је страница
- Свака табела (и сваки индекс) се састоји од страница
 - тачније, свака табела се записује у скупу страница једног простора за табеле
 - те странице се интерно распоређују по контејнерима тог простора за табеле



Страница (2)

- Величина странице је једна од карактеристика простора за табеле
 - све странице једног простора за табеле су исте величине
- Ако су странице велике
 - могу да садрже више редова
 - мање се тражи по диску
 - мања је дубина индекса
 - редови табеле могу да буду већи
 - обично један ред мора да стане у страницу
 - ако и може да се "прелије" у више редова, то смањује ефикасност
 - уобичајено за складишта података и аналитичке базе података
- Ако су странице мале
 - већа је искоришћеност простора на диску
 - мање је непотребно читања и писања
 - уобичајено за трансакционе базе података



Страница (3)

- Страница се састоји од
 - заглавља (енгл. *header*) и
 - записа (енгл. *record*)
- Не мора бити попуњена до краја
 - ако табела има много непопуњених страница, онда је повећано заузеће простора и успорен рад
 - оптимизација попуњености страница може да се изводи аутоматски или мануелно, али има релативно високу цену, зато што се индекси реферишу на странице
 - код индекса је слично, али неке врсте индекса увек имају оптималну организацију
 - нпр. алгоритми ажурирања индекса заснованих на Б-стаблима увек одржавају попуњеност страница у опсегу 50%-100%

DB2 REORGCHK

...
 Table statistics:
 F1: 100 * OVERFLOW / CARD < 5
 F2: 100 * (Effective Space Utilization of Data Pages) > 70
 F3: 100 * (Required Pages / Total Pages) > 80

SCHEMA.NAME	CARD	OV	NP	FP	ACTBLK	TSIZE	F1	F2	F3	REORG
Table: IP.DISORDER_NUMERIC	4088695	0	7198	7198	-	2.33e+08	0	99	100	---
Table: IP.MAIN	704	0	18	18	-	548416	0	100	100	---
Table: IP.CDS	4357	1169	16	21	-	535911	26	86	80	*--
Table: IP.RIPITI_IC	3.6e+07	0	32190	32195	-	9.08e+08	0	86	99	---
Table: IP.RPT_IN	0	0	0	1	-	0	0	-	0	---
Table: IP.SPIDER2_ALL_2	527752	0	841	1698	-	27443104	0	49	49	**
Table: IP.SPIDER2_ALL	527758	0	829	2509	-	26915658	0	32	33	**

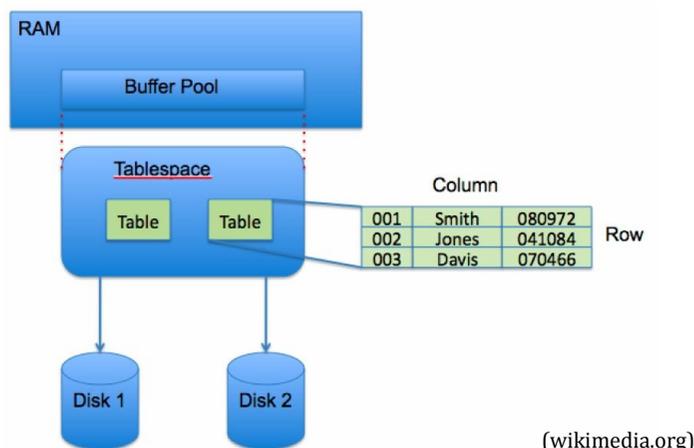
...
 Број редова који су премештени у другу страницу због величине
 Број непразних страница
 Број страница

Бафер за странице



- Бафер за странице (енгл. *buffer pool*) је меморијски простор предвиђен за чување копија једног броја страница ради омогућавања бржег приступа подацима
 - Одређује се на нивоу простора за табеле
- Што је бафер за странице већи, то је број приступа диску мањи

Бафер за странице



Бафер за странице (2)



- Све операције над подацима (и читање и писање) се одвијају над подацима који су у страницама бафера за странице
 - да би се податак прочитао, одговарајућа страница мора да буде у баферу
 - да би се податак променио, одговарајућа страница мора да буде у баферу
- Бафер за странице једног простора за табеле има улогу радне меморије за рад са подацима који су у том простору за табеле



Бафер за странице (3)

- Добро конфигурисање простора за табеле и бафера за странице може да буде од пресудног утицаја на перформансе
- Често се прави више простора за табеле истих карактеристика само да би се раздвојили њихови бафери страница
 - Чак и када цела база података физички стаје на један диск
 - Табеле и индекси се обично раздвајају



Бафер за странице (4)

- У идеалном случају је цела база података у меморији
 - данас је то често оствариво у пракси
 - бар за пословне трансакционе базе података
- Ако то није могуће, тежи се да у меморији буду бар најчешће коришћени елементи
 - најважнији индекси (или бар првих неколико нивоа индекса)
 - мање табеле (шифарници)
 - зато се такви елементи стављају у посебне просторе за табеле, да би могли да се посебно баферишу



Прављење бафера за странице

```
CREATE BUFFERPOOL [ IMMEDIATE | DEFERRED ]
[ DATABASE PARTITION GROUP ... ]
[ SIZE <broj stranica> [ AUTOMATIC ] ]
[ PAGESIZE <n> K ]
...
```



Прављење бафера за странице (2)

- При прављењу бафера за странице одређују се бар:
 - име бафера
 - величина страница (подразумевана вел. зависи од конфигурације система)
 - величина бафера (подразумевано 1000 страница)
 - опција *AUTOMATIC* омогућава аутоматско подешавање (оптимизацију) величине



Други важни концепти

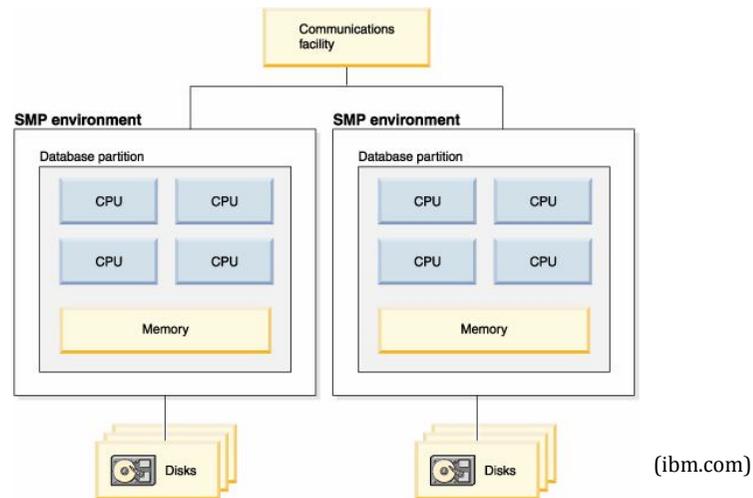
- Физичка организација података почива на још много важних концепата
 - партиционисане базе података
 - партиционисане табеле
 - колонске табеле
 - компресија података
 - материјализовани погледи
 - катанци или алтернативне технике за изоловање података
 - различити други концепти, често специфични за имплементације



Партиционисана база података

- База података може да се подели на партиције
 - партиција је *geo* БП који располаже сопственим структурама података
 - подаци, индекси, дневници трансакција, конфигурација
 - база података и даље функционише као целина
- Основна намена партиционисања БП је подизање перформанси
 - свака партиција може да ради на посебном чвору (серверу)
 - више чворова сарађује у раду једне БП
- Ради обезбеђивања скалабилности и флексибилности уводе се *логичке партиције*
 - више логичких партиција може да ради на истом серверу
 - могу да се према потреби промовишу у физичке

Партиционисана база података



Партиционисане табеле

- Садржај табеле може да се подели на више физичких целина (партиција) (обично на основу вредности кључа)
 - На пример, сви подаци о студијама би могли да се партиционису по години уписа студента
- Табела се и даље логички употребљава као неподељена целина
- Свака партиција се физички понаша као посебна табела
 - партиције могу да иду у различите просторе за табеле
 - могу одвојено да се бришу, архивирају, рестаурирају...
- Тиме се омогућава
 - лакше архивирање старих података,
 - различито управљање баферисањем нових и старих података
 - боља паралелизација читања и писања и друго



Колонске табеле

- Колонске табеле (тј. табеле организоване по колонама) су физичка репрезентација у којој се подаци не чувају “ред-по-ред” него “колона-по-колона”
- Намењене превасходно аналитичким базама података
 - и у другим случајевима, ако се на табели *изузетно ретко* врши рестрикција и *скоро никада* ажурирање
- Колонска организација се често користи и код нерелационих база података



Компресија података

- Савремени СУБП обично омогућавају да се подаци компримују, како би заузимали мање места
- То обично не утиче на начин употребе података
- Савремен харвер омогућава да се компримовање и декомпримовање таквих података врши „у ходу“, без видљивог утицаја на перформансе
- Слично је и са енкриптовањем



Материјализовани погледи

- Материјализовани погледи (енгл. *materialized query table*) су табеле чија структура и садржај су дефинисани датим упитом
- Врсте:
 - системски одржаване
 - кориснички одржаване
 - табеле у сенци (*shadow table*)
 - одржавају се репликацијом
 - федеративне
 - одржавају се алатима за федеративне базе података



Материјализовани погледи (2)

- Основна намена им је убрзавање сложених (обично аналитичких) упита
 - над подацима који се релативно ретко мењају
 - који се релативно лако ажурирају у случају промене података:
 - агрегирани упити
 - спајања и агрегације над више табела
 - упити над често коришћеним скупом података

Литература за тему



- Гордана Павловић-Лажетић, **Увод у релационе базе података**, 2. изг. Математички факултет, 1999.
 - доступно онлајн: <http://poincare.matf.bg.ac.rs/~gordana/urbp-2016.htm>
- Документација за DB2 11.5:
 - онлајн:
 - https://www.ibm.com/support/knowledgecenter/SSEPGG_11.5.0
 - ПДФ:
 - <https://www.ibm.com/support/pages/node/627743>