Komentarze

Bazy danych: DDL i ETL

Spis treści

[Refleksje 3](#_Toc39764383)

[Brak widoku 3](#_Toc39764384)

[COPY-paste skryptu 3](#_Toc39764385)

[brak więzów fk + on delete/on update 4](#_Toc39764386)

[Na marginesie – ERD 4](#_Toc39764387)

[Wiadomość do 160798 4](#_Toc39764388)

[Wiadomość do 160802 4](#_Toc39764389)

[( ͡° ͜ʖ ͡°) 5](#_Toc39764390)

[Użycie Accessa + Excela 5](#_Toc39764391)

[It’s bullshit. Oh, hi mark (1) 5](#_Toc39764392)

[Wiadomość do 160763 5](#_Toc39764393)

[Wiadomość do 160800 5](#_Toc39764394)

[Jak zdobyć bonusowe punkt (1) 6](#_Toc39764395)

[Jak zdobyć bonusowe punkt (2) 6](#_Toc39764396)

[Float vs NUMERIC 6](#_Toc39764397)

[Good one 6](#_Toc39764398)

[One comment 7](#_Toc39764399)

[A propos niektórych ddl-ek 7](#_Toc39764400)

[Wiadomość do 160767 8](#_Toc39764401)

[Wait what? 8](#_Toc39764402)

[Wiadomość do 160823 8](#_Toc39764403)

[Wiadomość do 157760 8](#_Toc39764404)

# Refleksje

Tu macie zbiorcze refleksje

## Brak widoku

Widok to obiekt bazodanowy. Sam SQL który służy jako definicja widoku nie wystarcza. Tj:

SELECT

m.title,

r.userId,

r.rating

FROM

movies m

JOIN rating r ON r.movieID=m.movieID

ORDER BY

r.title ASC;

To nie wystarcza. Potrzebowałem:

CREATE VIEW MyView AS

SELECT

m.title,

r.userId,

r.rating

FROM

movies m

JOIN rating r ON r.movieID=m.movieID

ORDER BY

r.title ASC;

## COPY-paste skryptu

Bezpośredni copy-paste skryptu powoduje że jest on nieprzenośny. Na przykład:

CREATE DATABASE [Filmy]

CONTAINMENT = NONE

ON PRIMARY

( NAME = N'Filmy', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\MSSQL\DATA\Filmy.mdf' , SIZE = 8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )

LOG ON

( NAME = N'Filmy\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.MSSQLSERVER\MSSQL\DATA\Filmy\_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )

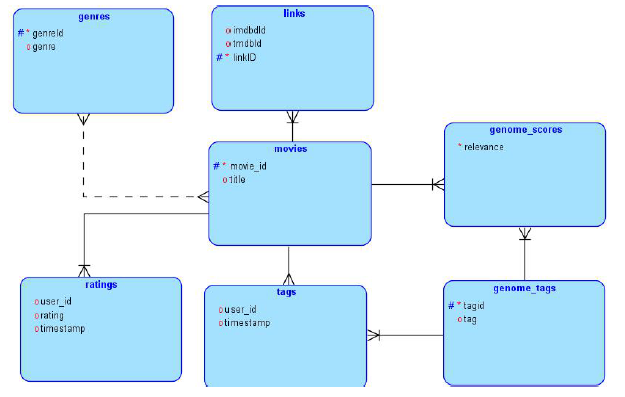
To eksponuje „wrażliwe miejsca” + wymusza sztywną konfiguracje bazy danych… zresztą dziennik jest tu ustawiony na maxsize 2 TB LOL.

## brak więzów fk + on delete/on update

Taka sytuacja ma miejsce w gatunkach. Usunięcie gatunku powinno powodować usunięcie filmów podanego gatunku (Albo nullowanie FK). W przeciwnym razie BD może tylko rosnąć / wymusza usuwanie w pewnej kolejności: najpierw filmy, potem gatunki

## Na marginesie – ERD

Nie obcinam za to pkt-ów, ale diagram ERD/UML Class przy modelowaniu BD muszą odwzorowywać 1-w-1 strukturę tabel. Jeśli korzystamy z czysto relacyjnych baz danych (bez obiektowej domieszki) to nie można osiągnąć związku wiele-do-wielu bez przechodniej tabeli. Taki diagram jest zły:



Pomimo że obiektywnie rzecz biorąc oddaje koncept BD.

## Wiadomość do 160798

Zabrakło Ci widoku i sensownych constraintów dla FK. Za użycie SSIS dorzucam mnóstwo bonusowych punktów, bo to przykład użycia dobrego narzędzia do pracy.

## Wiadomość do 160802

Bardzo mi się to dobrze czytało. Do tej pory najciekawsze ze sprawozdań – dorzucam bonusowe punkty…. Pomimo że nie jestem fanem wykorzystania Accessa :P z drugiej strony MSSQL Server ma mostek Accessowy więc … why not?

Plus o takie constrainty FK mi chodziło.

## ( ͡° ͜ʖ ͡°)

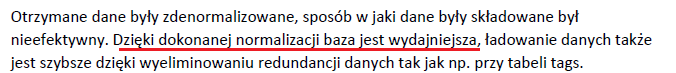


## Użycie Accessa + Excela

Doceniam koncept, doceniam.

## It’s bullshit. Oh, hi mark (1)

**CYTAT:**



**KONIEC CYTATU**



**Nie.**

Na odwrót – normalizacja BD **pogarsza** wydajność odczytów (a nawet mógłby wykazać w pewnych warunkach, że zapisu). (Jak ktoś to zapamięta to może zalśnić na liście numer 5.) Normalizacja ma na celu wymuszenie, by aktualizacja (lub zapis) danych był zależny od pojedynczej lokalizacji na dysku. Tym samym, jak dysk natrafi na tę lokalizację, to może przestać pracować.

Skutkiem ubocznym zdenormalizowanych baz danych jest to, że programiści muszą aktualizować logicznie tożsamą informację w wielu tabelach lub w wielu wierszach.

## Wiadomość do 160763

Nie widzę widoku. Byłby max – o takie DDL mi chodziło (DDL\_stworzony przeze\_mnie.sql).

## Wiadomość do 160800

Nie widzę widoku. Nie podoba mi się nazewnictwo tabel (Relation\_18 WTF) i żadnych ON DELETE / UPDATE.

## Jak zdobyć bonusowe punkt (1)

**nazwy tagów**: także powinny wspierać Unicode (postępowo, żeby użytkownicy oprocz własnego alfabetu, mogli wpisywać w nich emoji),

UTF8 jako złoty standard

## Jak zdobyć bonusowe punkt (2)

Np. użyć Pythona

## Float vs NUMERIC

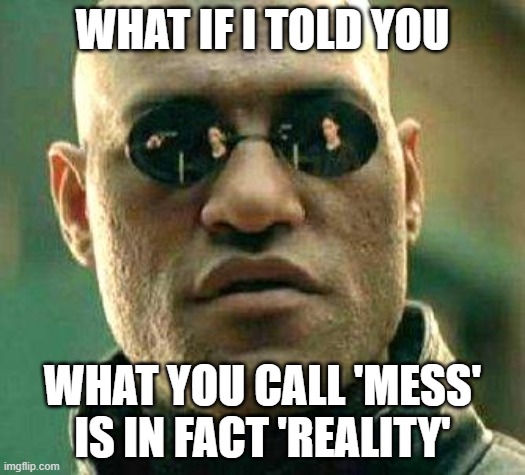
Numeric > Float.

Nie obcinam punktów za to.

Fun fact: typ MONEY broń boże nie wolno używać do przechowywania danych dot. rachunkowości tj. na podstawie kolumn typu MONEY nie wolno wystawiać rachunku klientowi lub pokazywać zarządowi ile zarobili. Bo to FLOAT.

## Good one





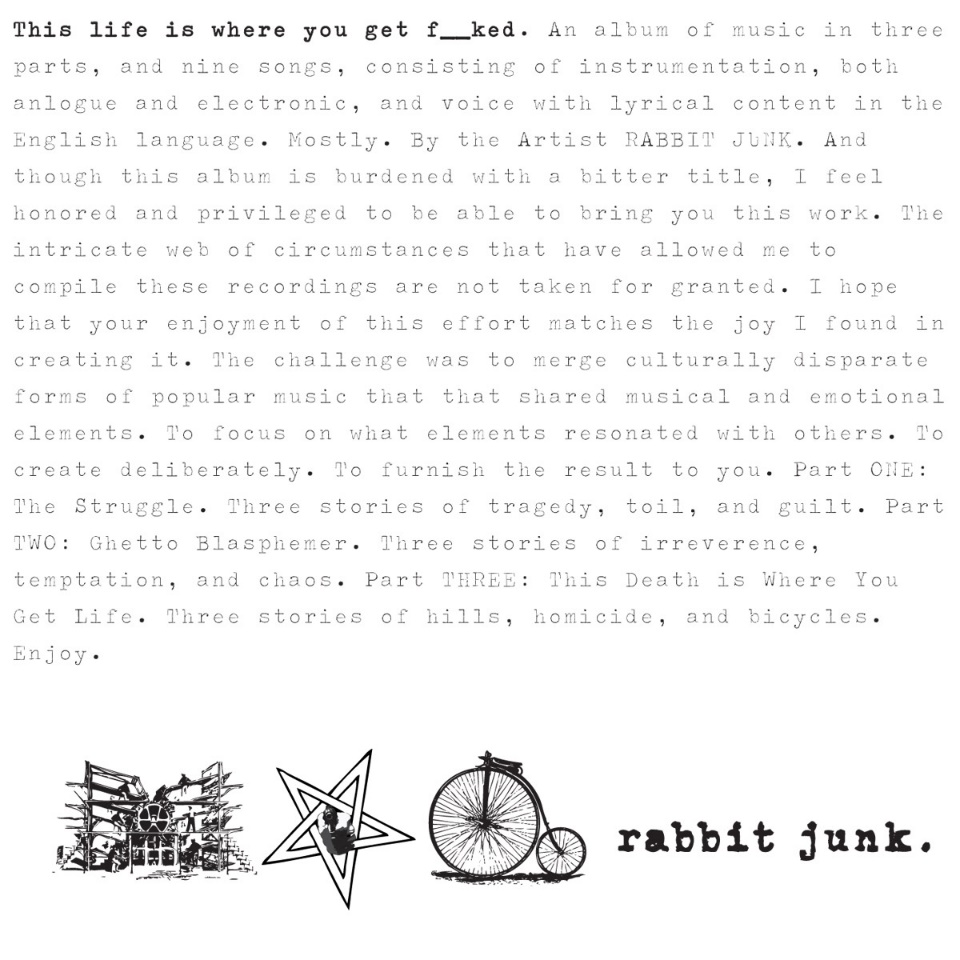
## One comment

Już któraś osoba zrobiła:

user: 'sa',

password: 'admin123',

**Nie robić tak. Nie dawać passów. W przeciwnym wypadku czeka Was los jaki zgrabnie ujął zespół Rabbit Junk:**



## A propos niektórych ddl-ek

Niektóre z DDL-ek muszą działać w ramach pojedynczego połączenia – w ramach bazy w obrębie której tworzony jest obiekt. Widoki są obiektami objętymi tego typu restrykcjami.

Długo czekałem, aż u kogoś pojawi się ten fragment:

EXECUTE('CREATE VIEW [dbo].[users\_ratings]

AS

SELECT

m.title,

r.userId,

r.rating

FROM

ratings r

INNER JOIN movies m ON m.movieId = r.movieId');

Który jest częstym i sprytnym ominięciem tworzenia widoków w ramach jednego wsadu DDL.

Można inaczej: oddzielać wsady za pomocą słowa kluczowego GO.

// Jakieś DDL-ki

GO

CREATE VIEW [dbo].[users\_ratings]

AS

SELECT

m.title,

r.userId,

r.rating

FROM

ratings r

INNER JOIN movies m ON m.movieId = r.movieId;

GO

// Jakieś DDL-ki

GO

## Wiadomość do 160767

Ogólnie pro, ale nie ma diagramu i jakiegoś PDF-a zbierającego info.

## Wait what?

timestamp varchar(200) o\_O

Lepiej:

Timestamp BIGINT OK but MEH

Timestamp TIMESTAMP ̿̿ ̿̿ ̿'̿'\̵͇̿̿\з= ( ▀ ͜͞ʖ▀) =ε/̵͇̿̿/’̿’̿ ̿ ̿̿ ̿̿ ̿̿

Timestamp DATETIME OK

## Wiadomość do 160823

Brak FK, widoków.

## Wiadomość do 157760

Bardzo mi się podobało to sprawozdanie… Jedyny mankament jaki mi się praktycznie zdarzył to chyba wysłałeś poprawkę która mi zamieszała co komu oceniłem :P