**RDI – Casus ODISEE**

Je gaat in deze casus voor RDI verder waar je met DDDQ gebleven bent. Je werkt dan ook verder met dezelfde groep aan jullie eigen ODISEE oplossing uit die casus. Die mag je uiteraard aanpassen indien je dat nodig of wenselijk acht. Leg dan wel even uit dat je dit doet en waarom. Een reden kan b.v. zijn: de feedback op jullie uitwerking bij DDDQ. Misschien hadden jullie Rules gemist of een suboptimale uitwerking gemaakt die ook in deze RDI casus zou kunnen doorsijpelen.

Naast het ontwerp en de conversie van data (DDDQ casus), wil ODISEE extra bevragingen en constraints toevoegen. Daarvoor zul je in deze casus code gaan schrijven. Dit zal vaak in de vorm van Stored Procedures of Triggers zijn, maar andere onderdelen die jullie al kennen mogen gebruikt worden. Denk daarbij aan Views en Functies. Let er daarbij wel goed op dat je die dan goed gebruikt en geen potentiële performance issues introduceert.

Je zult vaak meerdere opties voor queries moeten vergelijken. Denk eraan dat je voor deze keuze een realistische hoeveelheid data in jouw database moet hebben. Maak die dus aan en beschrijf hoe je bepaald hebt dat dit realistische data is.

Bij het beoordelen van deze casus zal naar alle beoordelingscriteria gekeken worden die in de module beschrijving staan. Daarbij wordt van een beoordelingsmodel met rubrics gebruik gemaakt, bekijk die vooraf om zelf al goed te kunnen bepalen hoe je een voldoende of meer gaat halen.

**Bevragingen**

Hieronder vind je een lijst met bevragingen die ODISEE door de database wil kunnen laten uitvoeren. Schrijf de code zodat de database het juiste resultaat voor deze bevraging teruggeeft.

Maak voor iedere query die je schrijft minimaal één alternatieve implementatie indien mogelijk en vergelijk de queryplannen. Kies op basis van het queryplan, performance en onderhoudbaarheid van de code de juiste query en verklaar jouw keuze.

1. Geef van een film, de hele reeks waar die bij hoort met volgnummer en in de juiste volgorde. Indien hij niet in een reeks zit, is de lijst gewoon één lang met volgnummer 1.

Dit moet één statement worden die van een variabel ID de reeks geeft zoals onderstaand voorbeeld:

DECLARE @MovieInReeks INT = 207989;

-- jouw statement hier levert onderstaand resultaat:

ITEM\_ID TITLE Volgnummer

207992 Matrix, The 1

207989 Matrix Reloaded, The 2

207991 Matrix Revolutions, The 3

1. Geef het statement dat per land het maandgebruik over 12 maanden geeft met daarbij het percentage dat die maand uitmaakt van het totaal van die 12 maanden.

Hier zijn twee interpretaties mogelijk: per land de afgelopen 12 maanden in rijen, hier onder de uitdraai voor b.v. ‘Netherlands’ uitgevoerd in april 2019.

Year Month ItemsPerMonth PercentageOfTotal

2018 4 60 4.44%

2018 5 60 4.44%

2018 6 60 4.44%

2018 7 56 4.14%

2018 8 132 9.76%

2018 9 155 11.46%

2018 10 141 10.43%

2018 11 138 10.21%

2018 12 148 10.95%

2019 1 137 10.13%

2019 2 117 8.65%

2019 3 148 10.95%

Een mooier alternatief is een statement dat per land de percentages en totalen geeft over een jaar in kolommen.

-- jouw statement hier levert b.v. onderstaand resultaat voor 2017

Countryname January Feburary March April May June July August September October November December TotalItems

Chile 9.76% 8.22% 10.06% 9.07% 9.26% 8.38% 8.91% 7.75% 7.28% 7.09% 7.53% 6.68% 3638

Greece 9.67% 7.69% 9.06% 8.49% 8.49% 7.92% 9.06% 8.49% 8.40% 7.78% 7.78% 7.17% 2120

Poland 9.93% 7.94% 9.41% 8.82% 8.82% 8.24% 8.24% 7.72% 7.72% 7.72% 7.72% 7.72% 1360

Netherlands 8.38% 7.26% 8.94% 8.38% 8.38% 7.82% 8.94% 8.38% 8.38% 8.38% 8.38% 8.38% 716

…

…

Maak ook deze tweede versie als je voor een 10 in aanmerking wilt komen.

1. Geef de 10 films met de hoogste mediaan van hun reviews. Wat zegt dit getal over een film?
2. In de oorspronkelijke data van IMDB staat in Imported\_Director\_Genre een veld Prob. Dit geeft de kans dat een film in deze database van dat genre is. Onderstaande query geeft de resultaten voor Sergio Leone.

SELECT \*

FROM myimdb.dbo.Imported\_Director\_Genre

WHERE did = 46046

ORDER BY Prob

Staan deze gegevens in de juiste volgorde? Maak een query die deze gegevens produceert vanuit jouw eigen database op volgorde van kans, met de grootste kans bovenaan.

1. Vermoedelijk zijn niet alle Film ID’s in gebruik. Geef het statement dat de langste aaneengesloten reeks geeft van Film ID’s in jouw database. Geef ook het statement dat de langste reeks geeft die NIET in de database aanwezig is. Geef van beide statements het start en eind ID van de reeks.

**Constraints**

Implementeer onderstaande constraints. Het kan zijn dat je een aantal al tijdens de casus DDDQ heb geïdentificeerd en/of zelfs al gemaakt. Lever in dat geval die code weer in voor deze casus. Indien je de constraints procedureel oplost, maak je **zowel een Stored Procedure als een After Trigger**. Zorg voor nette **error handling** in jouw code en schrijf een **complete testset** voor beide constraint implementaties.

Leg uit of de SP of Trigger jouw voorkeur heeft en uiteraard waarom. Als je bij een constraint vindt dat een **Instead Of Trigger** de betere variant is, maak je die **ook** en leg je uit waarom dit de beste oplossing is voor deze constraint.

* Een film of spel hoort altijd bij minimaal één genre.
* Bij een film met previous part, is de film later uitgebracht dan het previous part.
* Het kan natuurlijk gebeuren dat iemand meerdere abonnementsperiodes heeft. Dan kunnen de verschillende abonnementsperiodes van een persoon niet overlappen.
* De datum waarop een film wordt bekeken valt binnen de/een abonnementperiode.
* Voor een geldige film recensie zijn cijfers voor Plot en Acting verplicht en dient minimaal een van de rubrieken Cinematography en Music and Sound te worden beoordeeld. Als hieraan is voldaan, dan wordt de recensie geaccepteerd en wordt het totaalcijfer berekend.
* Genres voor films en spellen zijn verschillend, deze mogen niet bij het verkeerde media-item gebruikt worden. Hetzelfde geld voor Review aspecten.

**Transactie management en Concurrency**

In zowel jouw SP's als Triggers maak je correct gebruik van Transacties. Kies het juiste isolation level bij een implementatie en leg uit waarom dat het juiste level is. Leg hierbij uit welke problemen zich kunnen voordoen in het default isolation level, hoe locking in die situatie werkt en hoe die d.m.v. het isolation level aangepast wordt.

**Indexeren**

Geef aan waar jij denkt dat indexen nuttig kunnen zijn. Toon het verschil aan tussen de situatie met en zonder index en verklaar de verschillen in performance en queryplannen. Mogelijk veranderd je keuze van query als je gebruik maakt van indexen, geef dat dan aan.