

Dokumentace k projektu z databázových systémů

Vypracovala:

Jméno a příjmení: Adéla Kopecká

Třída: C4B

Školní rok: 2022/2023

Střední průmyslová škola
elektrotechnická, Praha 2, Ječná 30

1. Úvod a analýza zadání
2. Relační a Logické schéma
3. Normalizace a Integrita databáze
4. Procedury, views, trigger a indexy
5. Uživatelské účty
6. Transakce
7. Místa pro zlepšení

Obsah dokumentace:

Úvod

Způsob zadání je faktura a tématem jsou stavebniny. Proto jsem si jako zadání zvolila vzorovou fakturu firmy vipstone, která zařizuje kamenné obklady a dlažbu které se dají využít při přestavbě či rekonstrukci domu/bytu. Z níž jsem došla k závěru tvorby databáze která eviduje objednávky, obsah a náležitosti jejich faktury. Při tvorbě jsem vycházela z myšlenky že jde o menší/větší podnik.

Cílem databáze je sestavit lehce zjednodušenou verzi ukázkové faktury. Rozsah databáze je, jak jsem již zmínila mírně osekáný například u entity Zamestnanec, není provedena perfektně (Zamestnanec má svou pracovní pozici plat. atd) ale nejde o věc na kterou by se primárně měla databáze soustředit.

Co se funkčních požadavků týká bude nutné vytvořit určité indexy na kodama zboží, dále view který bude obsahovat všechna data faktury, proceduru pro tvorbu objednávky s fakturou na kterou se bude vázat trigger který odečte zboží ze skladu po tvorbě objednávky.

odkaz na web kde byla faktura nalezena:

<https://www.vipstone.cz/vzorova-faktura/>

Cesta k dokladu:

src/vipstone faktura vzorová

Relační a logické schéma (E-R model)

Logické a relační schéma se nachází ve složce img, obě dvě jsou vyexportovány v svg - html z důvodu lepší čitelnosti

Obě dvě schémata vychází z faktury vipstone

Logické:

img\Logické_schéma

Relační:

img\Relační_schéma

Normalizace a Integrita databáze

a)Integrita

1) Entitní integrita

Každá Entita obsahuje číselný atribut *ID[nazev entity]* který slouží jako primární klíč ve všech entitách je pro tento atribut nastavený auto increment, je také not null. *Entitní integrita byla splněna.*

2) Doménová integrita

Jak nám říká doménová integrita tak atributy musí být omezeny rozsahem, datovým typem not null. Tak také bylo učiněno a všechny entity mají atributy omezené. *Doménová integrita byla dostatečně splněna.*

př: entita výrobce má atributy: IDvyrobce..... integer, not null

Nazev.....varchar2(30), not null

Zkratka....varchar2(10)

E-mail.....varchar(30), not null

3) Referenční integrita

Byla splněna jelikož, všechny entity které mají kardinalitu tak na straně podřízené entity obsahují atribut prezentující cizí klíč.

Výpis cizích klíčů(FK) ve schématu podle vztahů (vycházím z Relačního schématu):

Vyrobce x Zbozi(1:N) - FK je na straně Zbozi =Vyrobce_ID_vyrobce

Zbozi x Objednavka/Zbozi(1:N) - FK je na straně Objednavka/Zbozi =Zbozi_ID_zbozi

Objednavka/Zbozi x Objednavka(1:N)- FK je na straně Objednavka/Zbozi Objednavka_ID_Objednavka

Prodejna x Objednavka(1:N) - FK je na straně Objednavka =Prodejna_ID_prodejna

Faktura x Objednavka(1:N) - FK je na straně Objednavka =Faktura_ID_Faktura

Zakaznik x Faktura(1:N) - FK je na straně Faktury=Zakaznik_ID_zakaznik

Zamestnanec x Faktura(1:N) - FK je na straně Faktury= Zamestnanec_ID_zamestnanec

Prodejna x Zamestnanec (1:N) - FK je na straně Zamestnanec= Prodejna_ID_prodejna

Objednavka x Faktura (1:1) - FK je na straně Objednavk= Faktura_ID_Faktura

a)Normalizace

1) 1. NF

Byla provedena dostatečně, za předpokladu že se jedná o malý až malý střední podnik.

2) 2. NF

Je dle mého názoru splněna, protože relační schéma neobsahuje složené klíče a ani žádné kandidátské klíče.

3) 3. NF

Dle mého názoru splněna, protože všechny atributy entit závisí na primárních klíčích entit.

Procedury, views, trigger a indexy

a) Procedury

Databáze obsahuje 2 procedury

stored procedure se nachází:

sql\Tvorba_db\Tvorba_procedur

1. Vypis_polozek_objednavky_podle_cisla

popis: vypíše podle objednávky čísla všechny položky

důvod: když by databáze sloužila reálnému webu tak by zákazníka určitě zajímalo co si objednal

2. Vypocet_cenny_objednavky

popis: Vypočítat celkovou cenu objednávky

důvod: aktivně se využívá v triggeru *Prepocitej_cenu_after_inser*

b) Views

Databáze obsahuje celkem 4 views.

SQL views se nachází:

sql\Tvorba_db\Tvorba_views

1. Pocet_zamestnancu_na_Prodejn

popis: pocet zamestnancu na prodejnách

důvod: Ciste informativni pro vedeni

2. Zasoby_na_sklade

popis: Zbozi ktere je na sklade a kdo je dodavatel a kontakt na dodavatele

důvod: Pokud by se uvazovalo nad api k databázi tak by toto view mohli vyuzit skladnici.

3. Polozky_Objednavky

popis: Ukazuje k jaké objednávce objednaný produkt patří

důvod: Kdyz se bali objednávky tak aby věděli do jaké objednávky co zabalit resp. vydat na výdeji

4. Vypis_Faktur_a_Zakazniku

popis: Vypisuje Zákazníka, číslo jeho objednávky , datum vystavení a splatnosti u faktury, částku , zda byla již zaplacená a kdo ji vystavil.

důvod: Při výdeji se může hodit k ověření zda bylo již zaplacené či nikoliv

c) Trigger

Databáze obsahuje 2 triggery

Triggery se nachází:

sql\Tvorba_db\Tvorba_triggeru

1. Prepocitej_cenu_after_inser

důvod: nikomu se celkovou cenu nechce počítat ručně

popis: při každé tvorbě nové instance aktualizuje cenu objednávky

2. Vytvor_Objednavku_k_nove_fakture

důvod: aby vždy objednávce náležela faktura a obráceně

popis: při tvorbě nové faktury vytvoří automaticky novou objednávku a propojí je spolu

d) Indexy

Databáze obsahuje celkem 4 Indexy které nejsou nad pk a dále indexy nad pk entit.

Indexy se nachází:

sql\Tvorba_db\Tvorba_indexu

1. Jmeno_a_prijmeni - nad entitou Zakaznik
2. Cislo_objednavky - nad Objednavkou
3. Cely_nazev - nad entitou Zbozi
4. Kod- nad entitou Zbozi

Uživatelské účty

databáze celkem obsahuje 3 uživatelské účty

Cesta k sql:

sql\Tvorba_db\Tvorba_loginu_a_useru

- 1) Vydejce
-Má oprávnění, dbreader protože jen vydává objednávky
- 2) Prodejce
-Má oprávnění, dbreader a dbwritera
- 3) Majitel
-Má oprávnění dbownera

Transakce

V databázi se nachází celkově 1 transakce a to v triggeru

Vytvor_Objednavku_k_nove_fakture, protože každé ke každé objednávce se váže 1 faktura

Místa pro zlepšení

První místo kde by se mohla databáze zlepšit je že by se na vztah Faktury s Objednavkou který je 1:1 nahlíželo tak že Faktuře náleží objednávka, tím pádem by FK byl na straně Faktury a lépe by se s entitou objednávky pracovalo.

Dále by se dala provést lepší normalizace např. u entity Prodejna kde by se adresa mohla rozdělit na více atributů. U zboží by se dal využít supertype a subtype a subtype by představovali obklady a chemikálie. Jako posledně by se u entity Faktura dal provést arcs respektive u nové entity, tak že by se vytvořila entita platba která by měla arcs na dvě entity platba hotově a platba kartou.

