ADATBÁZISOK 2

Számítógépes alkatrészek

|  |  |
| --- | --- |
| Készítette: Patócskai István,  Terezsák József  Leckekönyv: 16219205,16219230  Szak: Internet és elektronikus ügyvitel | Tanár: Dr. Simon János |

Tartalomjegyzék

[Projektleírás 3](#_Toc69821036)

[Adatbázis felépítése 4](#_Toc69821037)

[Az adatbázis relációs modell szemléltetése 5](#_Toc69821038)

[Tárolt eljárások 6](#_Toc69821039)

[Tárolt függvények 7](#_Toc69821040)

[Nézetek 8](#_Toc69821041)

[Triggerek 9](#_Toc69821042)

[Tranzakciók 10](#_Toc69821043)

[Felhasznált irodalom 11](#_Toc69821044)

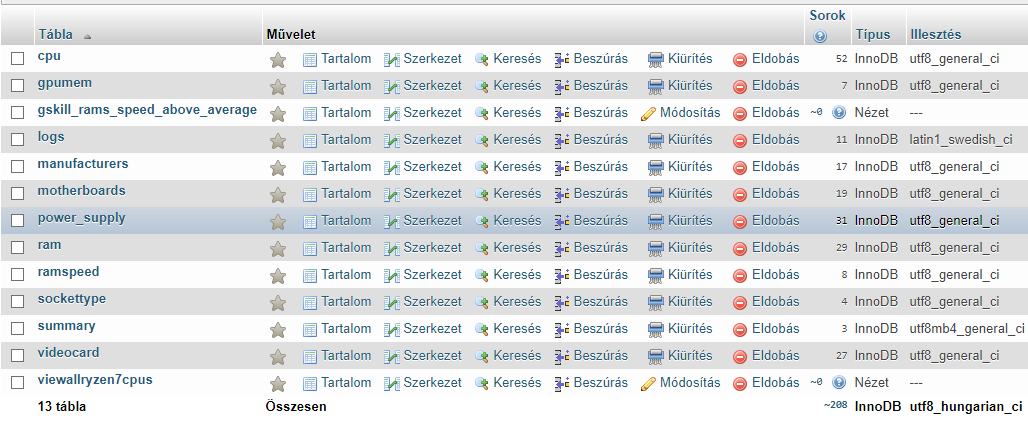
# Projektleírás

A projekt egy számítógépes alkatrészekkel foglalkozó adatbázist szemléltet. Az adatbázis számon vezeti a következőket: Processzor (CPU), Videó kártya Memória (GPUMEM), Naplók (LOGS), Készítők (MANUFACTURERS), Alaplapok (MOTHERBOARDS), Tápegység (POWER\_SUPPLY), Memória (RAM), Memória Sebessége (RAMSPEED), Foglalat Típus (SOCKETTYPE), Összefoglaló (SUMMARY), Videó kártya (VIDEOCARD).

# 

# Adatbázis felépítése

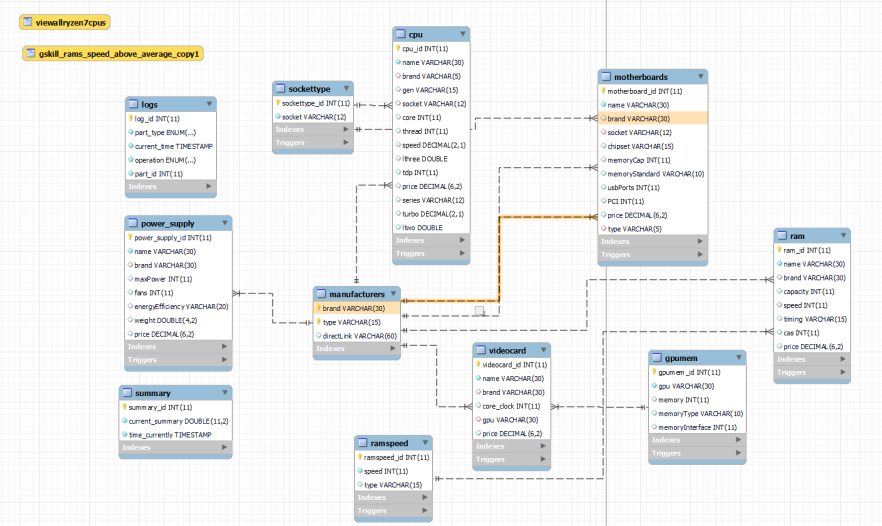
Az adatbázis a következő táblákból áll:



1. *ábra adatbázis táblázatai, és nézetei*

A „gskill\_rams\_speed\_above\_average” és a viewallryzen7cpus” két nézet, nem táblázatok (1. ábra).

# Az adatbázis relációs modell szemléltetése

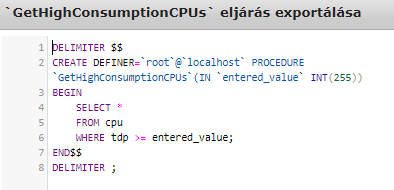


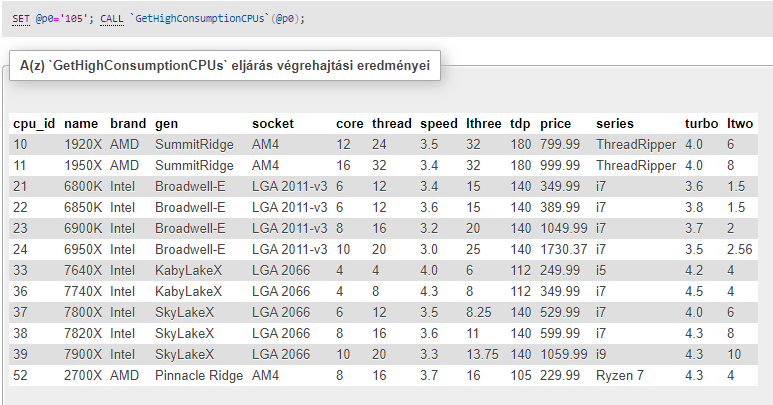
*2. ábra Az adatbázis relációs modellje*

A fenti ábrán (2. ábra) láthatjuk az adatbázis relációs felépítését, valamint azt is miként vannak összekötve a táblák egymással.

# Tárolt eljárások

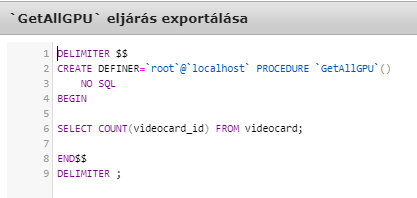
A „GetHighConsumptionCPUs” eljárás kilistázza azokat a CPU-kat az adatbázisban amelyeknek a TDP-jük (Thermal Design Power) a bevitt értékkel megegyezik vagy nagyobb tőle.

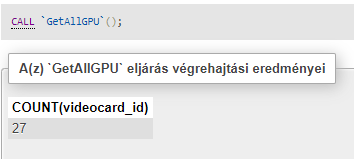


****

*3. ábra első tárolt eljárás*

A „GetAllGPU” eljárás megszámolja az összes elérhető Videó kártyát az adatbázisban, és visszaadja, hogy mennyi van belőle.

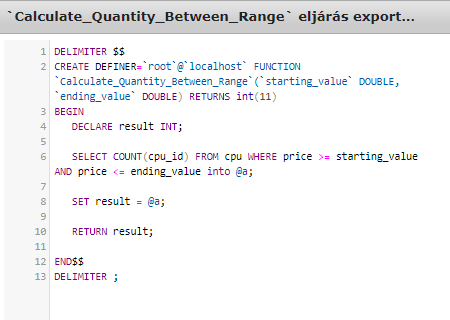
****

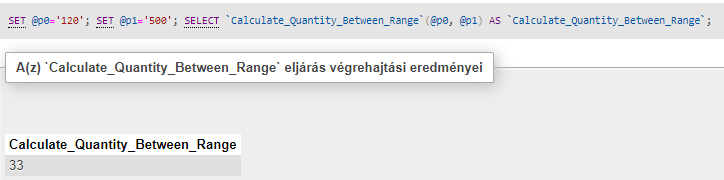
****

*4. ábra második tárolt eljárása*

# Tárolt függvények

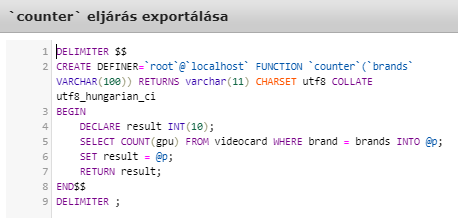
A „Calculate\_Quantity\_Between\_Range” függvény megszámolja hány olyan CPU elérhető az adatbázisban mely a megadott intervallumban helyezkedik el („starting\_value” és „ending\_value” között).

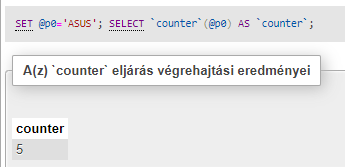
****

****

*5. ábra első tárolt függvény*

A „counter” függvény megszámolja hány olyan típusú videó kártya elérhető az adatbázisban mely megegyezik a bevitt adattal, pl „ASUS”.

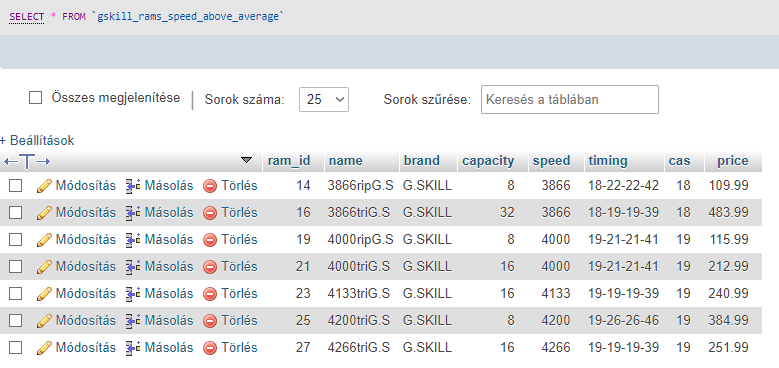
****

****

*6. ábra második tárolt függvény*

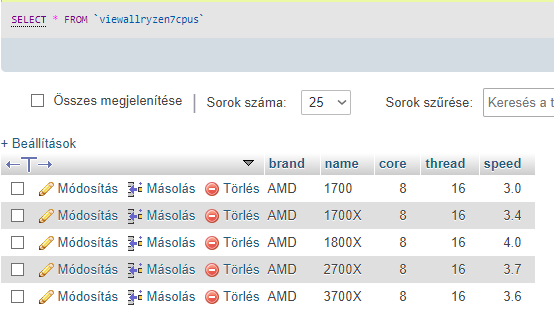
# Nézetek

gskill\_rams\_speed\_above\_average G.SKILL ramokat listázza ki átlag felett.

****

*7. ábra első nézet*

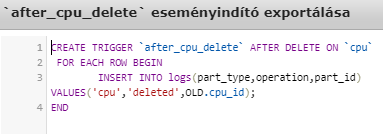
viewallryzen7cpus Az összes Ryzen 7 CPU-t kilistázza

****

*8. ábra második nézet*

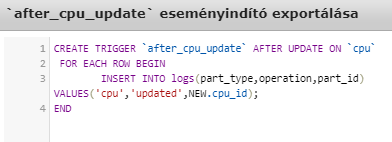
# Triggerek

az “after\_cpu\_delete” trigger beilleszti az eltávolított CPU-kat egy log táblázatba a következő paraméterek egyikével: „deleted”



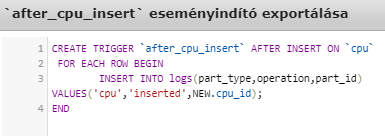
*9. ábra első trigger*

az “after\_cpu\_update” trigger beilleszti a frissített CPU-kat egy log táblázatba a következő paraméterek egyikével: „updated”

****

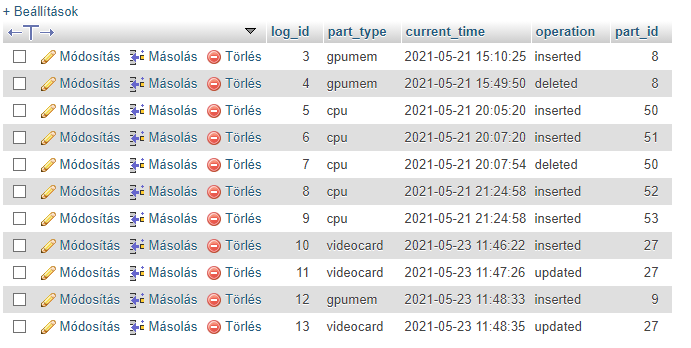
*10. ábra második trigger*

az “after\_cpu\_insert” trigger az új CPU-kat egy log táblázatba illeszti be a következő paraméterek egyikével: „updated”

****

*11. ábra harmadik trigger*

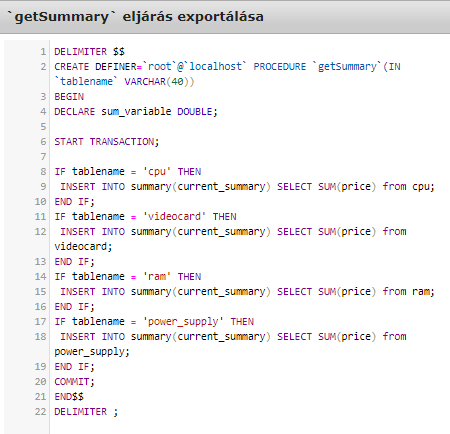
A triggerek által használt táblázat, a LOG tábla.

****

*12. ábra logs táblázat tartalma*

# Tranzakciók

A „getSummary” tranzakció egy paraméter alapján belehelyezi a summary táblába, megadott táblában levő áraknak az összegét tranzakció használatával.

****

*13. ábra első tranzakció*

A „countAllPartsSummary” kiolvassa a ram,videókártya,cpu táblákból a sorok számát, majd ezeket mind összeadja és a summary tábla 1. id-val rendelkező sorába helyezi bele, a sor módositásával és tranzakció használatával.

****

*14. ábra második tranzakció*

# Felhasznált irodalom

[1] Dr. Šimon Janoš – Priručnik za vežbe iz baze podataka II, Visoka tehnička škola strukovnih studija u Subotici, 2014.

[2] W3Schools – MYSQL : <https://www.w3schools.com/mysql/>

[3] Youtube - <https://www.youtube.com>

[4] SQLSCHAK - https://www.sqlshack.com