**NÉV: Tóth Botond**

**NEPTUN: MQH41V**

Adatbázisok 1, 2. ZH, 2022.05.04 szerda, 8.15

=============================================

Az alábbi PL/SQL feladatokat kell megoldani. Csak olyan programot értékelek, ami lefut.

Részpontszám csak kivételes esetben szerezhető a feladatokra, ha az output apró hibát tartalmaz.

Minden feladat 10 pontot ér, a ponthatárok: 20,30,40,50 pont.

A procedúrákat és függvényeket olyan névvel és paraméterezéssel kell létrehozni, ahogyan

alább szerepel.

A programokat a megadott paraméterekkel meg is kell hívni (lásd Tesztelés) és a forráskódot

és a futtatás során kapott outputot kell emailben elküldeni a nikovits@inf.elte.hu email címre.

A futtatás során kapott OUTPUT nélkül nem értékelem a feladatot!!!

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. feladat

----------

Írjunk meg egy függvényt, amelyik a paraméterül kapott számnak visszaadja

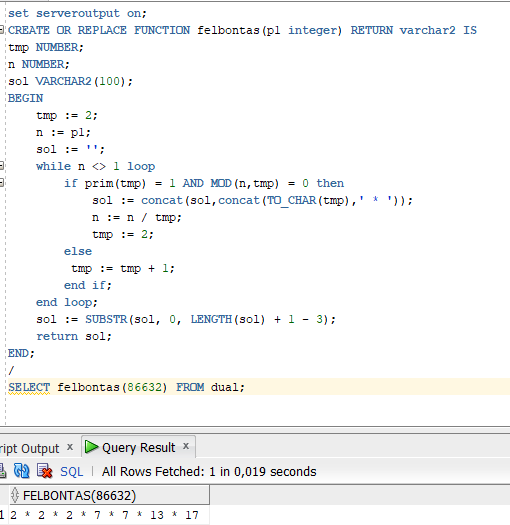
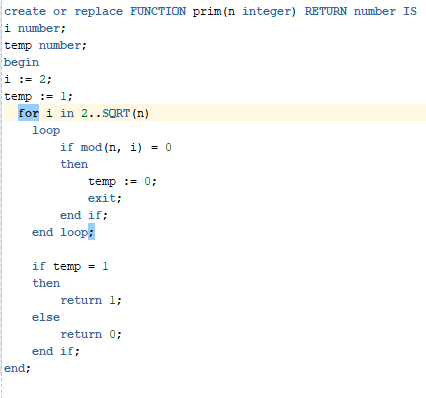
a prímtényezős felbontását egy karakterláncban.

Például: 120-ra visszaadja, hogy 2\*2\*2\*3\*5.

CREATE OR REPLACE FUNCTION felbontas(p1 integer) RETURN varchar2 IS

Tesztelés:

SELECT felbontas(86632) FROM dual;



2. feladat

----------

Írjunk meg egy procedúrát, amelyik megnöveli azon dolgozók fizetését, akik a paraméterként

megadott fizetési kategóriába esnek. A fizetést a dolgozó saját osztályának maximális

fizetésével kell megnövelni. A procedúra a módosítás után írja ki az összes dolgozó

átlagfizetését (az új fizetések átlagát).

A procedúra a futása végén adjon ki egy ROLLBACK utasítást, azért, hogy az adatok

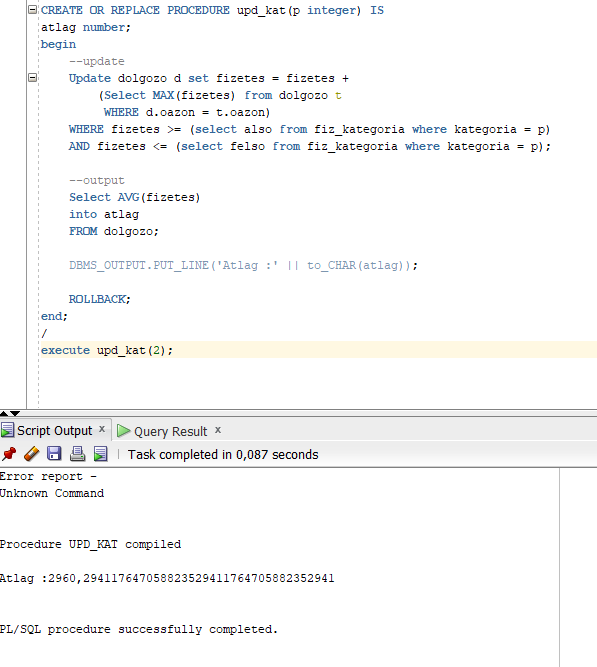
ne változzanak meg, és legközelebb futtatva is megfelelő eredményt írjon ki.

CREATE OR REPLACE PROCEDURE upd\_kat(p integer) IS

Tesztelés:

set serveroutput on

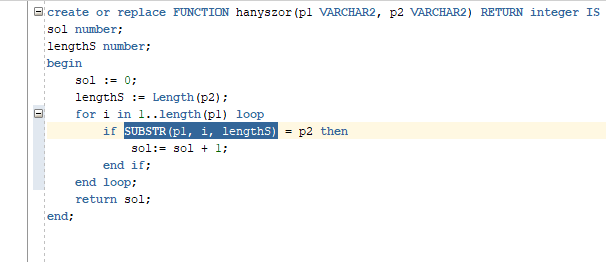
execute upd\_kat(2);



3. feladat

----------

Írjunk meg egy plsql procedúrát, amelyik kiírja azoknak a dolgozóknak a nevét, akiknek a nevében

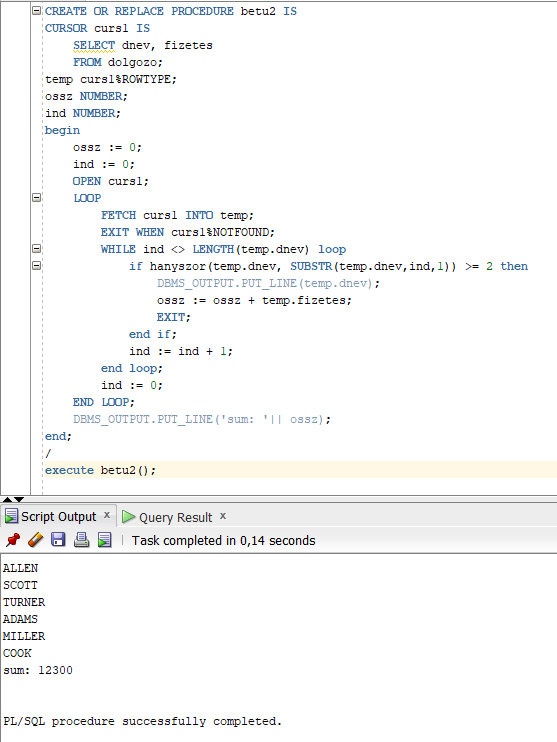
van két azonos betű (pl. TURNER nevében két R), valamint kiírja ezen dolgozók fizetéseinek összegét.

CREATE OR REPLACE PROCEDURE betu2 IS

Tesztelés:

set serveroutput on

execute betu2();



4. feladat

----------

Írjunk meg egy procedúrát, amelyik veszi az első n prímszámot (1. prímszám: 2, 2. prímszám: 3, stb.)

és beleteszi azokat egy asszociatív tömbbe. A procedúra a végén írja ki a tömb utolsó elemét és a

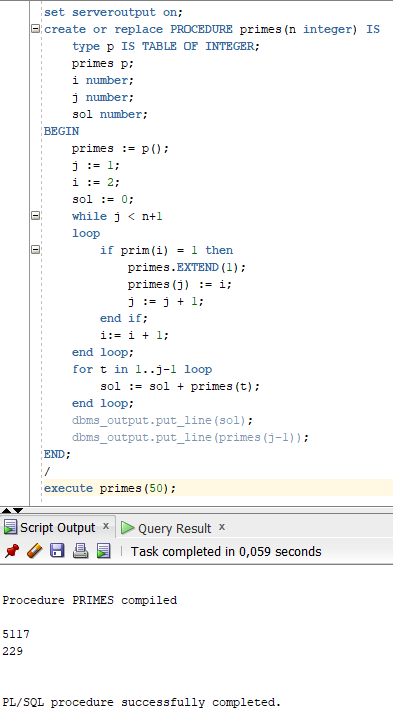
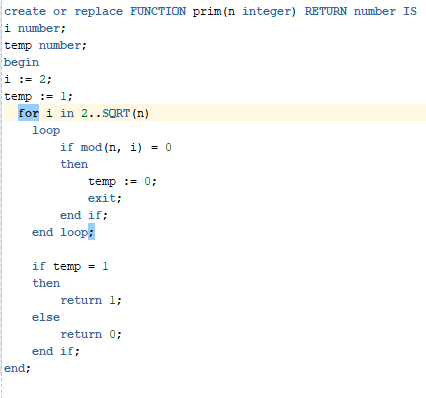
prímszámok összegét.

CREATE OR REPLACE PROCEDURE primes(n integer) IS

Tesztelés:

set serveroutput on

execute primes(50);



5. feladat

----------

Írjunk meg egy függvényt, amelyik egy karakteres típusú paraméterben egy dátumot

kap a következő formátumban: 'éééé.hh.nn' vagy 'nn.hh.éééé'.

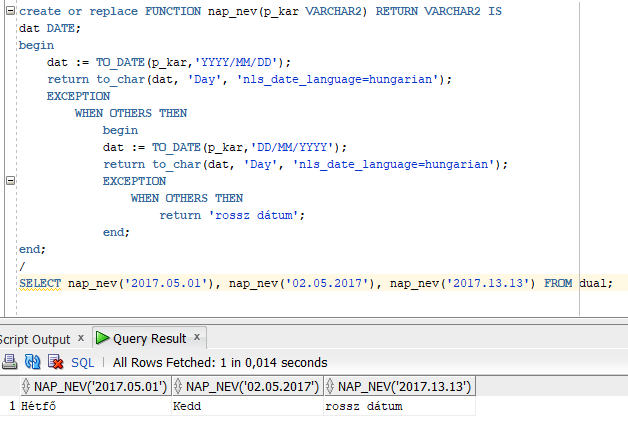
A függvény adja vissza a nap nevét, pl. 'kedd'.

Ha a megadott karakterlánc nem egy érvényes dátum, akkor adja vissza, hogy 'rossz dátum'.

CREATE OR REPLACE FUNCTION nap\_nev(p\_kar VARCHAR2) RETURN VARCHAR2 IS

Tesztelés:

SELECT nap\_nev('2017.05.01'), nap\_nev('02.05.2017'), nap\_nev('2017.13.13') FROM dual;

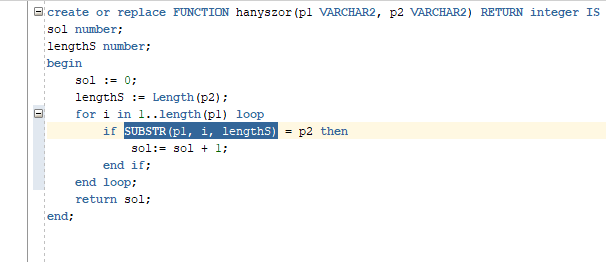


6. feladat

----------

Írjunk meg egy procedúrát, amelyik a NIKOVITS.VAGYONOK tábla alapján kiírja azoknak

a személyeknek a nevét és vagyonát, akikre igaz, hogy sem az ő nevük, sem egyetlen

felmenőjük neve sem tartalmazza a paraméterként megadott karakterláncot.

CREATE OR REPLACE PROCEDURE kar\_nincs(p varchar2) IS

Tesztelés:

set serveroutput on

execute kar\_nincs('B');

