

# Jegyzőkönyv

## Adatbázis rendszerek II.

### Féléves Feladat

#### Témakör leírás

A beadandóm egy kültéri levegő paramétereit mérő rendszerről szól, amely a levegő különböző szennyezőit méri például: CO, NO<sub>2</sub>, O<sub>3</sub>, NO, BENZOL, PM10, SO<sub>2</sub>. Az eszköznek rendelkeznie kell egy vezetékek nélküli összeköttetéssel, amelyen keresztül az adatokat képes továbbítani egy online adatbázisba.

Az élet elengedhetetlen feltétele a levegő, amelyet a Földet körülvevő gázok elegye alkot. Valaha az atmoszféra főleg szén-dioxidból állt. A növények, mint egy 600 millió éve a fotoszintézis során átalakították oxigénné. A levegő nitrogén tartalma pedig egyre csak nőtt volt olyan időszak, amikor elérte a 80%-ot is. A levegő alkotórészei az oxigén (20,93%), nitrogén (78,09%), argon (0,93%) és egyéb gázok. Ez az eszköz azért fontos mert a levegőben jelen vannak olyan szennyező anyagok, amelyek károsíthatják szervezetünket. A légszennyezés problémáival a 20. század 60-as éveiben kezdtek foglalkozni. Ebben az időszakban a nagyvárosok levegője olyan szennyezett volt, hogy világszerte több 1000 ember halálát okozta. A légszennyezettségért akkoriban főleg az ipar és a fűtés céljából elégetett tüzelőanyagok feleltek. A légszennyezés csökkentésének első lépése, hogy a régi elavult ipari technológiák helyett új környezetbarát technológiákat vezetnek be. A levegőben vannak úgynevezett nyomgázok: széndioxid, metán, vízgőz, nemesgázok. A levegő egyéb szennyező gázokat is tartalmaz: kén-dioxid, szén-monoxid és szálló por. Az ember és az élőlények számára nagyon fontos a páratartalom.

meresi adatok								
<u>meresi</u> <u>adatkulcs</u>	hőmerseklet	paratartal om	szallopor	ken- dioxid	szen- monoxid	nitrogen- monoxid	meres ideje	<u>id</u>

merőeszköz								
<u>eszközkulcs</u>	gyartas ideje	magassag	iranyitoszam	település	utca	hazszam	tipus	gyarto

karbantartas					
<u>karbantartás</u> <u>azonosito</u>	<u>Melyiket</u>	Datum	<u>hova</u>	<u>ki</u>	

telephely					
<u>telepkulcs</u>	telefonszam	iranyitoszam	település	utca	hazszam

szerelo				
<u>szerelokulcs</u>	nev	javitas hatarideje	szerviz dij	<u>telep</u>

## SQL kód:

```
Create table telephely (  
    telepkulcs number primary key NOT NULL,  
    telefonszam varchar(11),  
    irszam number,  
    varos varchar(30),  
    utca varchar(30),  
    hazszam number  
);
```

```
Create table meroeszkoz (  
    eszkozkulcs number primary key NOT NULL,  
    gyartas_ido number,  
    magassag number,  
    irszam number,  
    varos varchar(30),  
    utca varchar(30),  
    hazszam number,  
    tipus varchar(30),  
    gyarto varchar(30)  
);
```

```
Create table szerelo (  
    szerelokulcs number primary key NOT NULL,  
    nev varchar2(30),  
    javitas_hatarido date,
```

```
telepkulcs number,  
  
CONSTRAINT fk_telehely  
  
    FOREIGN KEY (telepkulcs)  
  
    REFERENCES telephely(telepkulcs)  
  
);
```

```
Create table meresiadatok(  
  
meresiadatkulcs number primary key NOT NULL,  
  
homerseklet number,  
  
meres_ideje date,  
  
paratartalom number,  
  
szallopor number,  
  
kendioxid number,  
  
szenmonoxid number,  
  
nitrogéndioxid number,  
  
eszkozkulcs number,  
  
CONSTRAINT fk_meroeszkoz  
  
    FOREIGN KEY (eszkozkulcs)  
  
    REFERENCES meroeszkoz(eszkozkulcs)  
  
);
```

```
Create table karbantartas(  
  
karbantartas_azonosito number primary key NOT NULL,  
  
datum date,  
  
eszkozkulcs number,  
  
telepkulcs number,  
  
szerelokulcs number,
```

```

CONSTRAINT fk_meroeszkoz_2

    FOREIGN KEY (eszkozkulcs)

    REFERENCES meroeszkoz(eszkozkulcs),

CONSTRAINT fk_telephely_2

    FOREIGN KEY (telepkulcs)

    REFERENCES telephely(telepkulcs),

CONSTRAINT fk_szerelo

    FOREIGN KEY (szerelokulcs)

    REFERENCES szerelo(szerelokulcs)

);

```

### **Táblázat feltöltéseinek mintapéldányai:**

```

insert into telephely values(1, 06204475215,3504,'Miskolc','Valamilyen utca',15);
insert into telephely values(2, 06209957546,3509,'Miskolc','Nem Létez? Utca',18);
insert into telephely values(3, 06301875546,4001,'Debrecen','adatbázis utca',6);
insert into meroeszkoz values(1,2017,5,3505,'Miskolc','Árpád utca',13,'y19','SPEC SENSORS');
insert into meroeszkoz values(2,2017,6,3514,'Miskolc','Mit írjak ide utca',4,'rz34','SPEC SENSORS');
insert into meroeszkoz values(3,2017,4,4440,'Tiszavasvári','Vásár utca',28,'y20','SPEC SENSORS');
insert into szerelo values(1,'Kiss Sándor',TO_DATE('2017/11/30', 'yyyy/mm/dd'),1);
insert into szerelo values(2,'Havas Jon',TO_DATE('2018/11/15', 'yyyy/mm/dd'),2);
insert into szerelo values(3,'Kovács Ferenc',TO_DATE('2017/11/29', 'yyyy/mm/dd'),3);
insert into meresiadatok values(1,9, TO_DATE('2017/03/11', 'yyyy/mm/dd'),40,34,2,400,32,1);
insert into meresiadatok values(2,12, TO_DATE('2017/03/12', 'yyyy/mm/dd'),20,28,2,359,31,2);
insert into meresiadatok values(3,6, TO_DATE('2017/03/03', 'yyyy/mm/dd'),30,24,2,394,26,3);
insert into meresiadatok values(4,7, TO_DATE('2017/03/05', 'yyyy/mm/dd'),31,26,2,399,27,1);
insert into karbantartas values(1, TO_DATE('2017/11/25', 'yyyy/mm/dd'),1,1,1);
insert into karbantartas values(2, TO_DATE('2017/11/27', 'yyyy/mm/dd'),2,2,2);
insert into karbantartas values(3, TO_DATE('2017/11/24', 'yyyy/mm/dd'),3,3,3);

```

## Tárolt eljárás adatok felvitelére

```
create or replace procedure ujEszkoz(p_eszkoszkulcs number, p_gyartas_ido number, p_magassag number,
p_irszam number, p_varos varchar, p_utca varchar, p_hazszam number, p_tipus varchar, p_gyarto varchar) is
exsisterror exception;
rows_found number;
begin
    select count(*)
    into rows_found
    from meroeszkoz
    where eszkoszkulcs = p_eszkoszkulcs;

    if rows_found != 0 then raise exsisterror;
    else
        insert into meroeszkoz values(p_eszkoszkulcs, p_gyartas_ido, p_magassag, p_irszam, p_varos,
        p_utca, p_hazszam, p_tipus, p_gyarto);
    end if;
exception
    when exsisterror then
        dbms_output.put_line('Ez a mer?eszköz már létezik!');
end;
```

## Tárolt eljárás adatok módosítására

```
create or replace procedure eszkozModositas(p_eszkoszkulcs number, p_ujgyartas_ido number,
p_ujmagassag number, p_ujirszam number, p_ujvaros varchar, p_ujutca varchar, p_ujhazszam number,
p_ujtipus varchar, p_ujgyarto varchar) is
notexsisterror exception;
rows_found number;
begin
    select count(*)
    into rows_found
    from meroeszkoz
    where eszkoszkulcs = p_eszkoszkulcs;

    if rows_found = 0 then raise notexsisterror;
    else
        update meroeszkoz set gyartas_ido = p_ujgyartas_ido, magassag = p_ujmagassag,
        irszam = p_ujirszam, varos = p_ujvaros, utca = p_ujutca, hazszam = p_ujhazszam,
        tipus = p_ujtipus, gyarto = p_ujgyarto where eszkoszkulcs = p_eszkoszkulcs;
    end if;
exception
    when notexsisterror then
        dbms_output.put_line('Ilyen azonosítójú mer?eszköz nem létezik!');
end;
```

## Tárolt eljárás adatok törlésére

```
create or replace procedure Eszkoztorles(p_eszkozkulcs number) is
notexisterror exception;
rows_found number;
begin
    select count(*)
    into rows_found
    from meroeszkoz
    where eszkozkulcs = p_eszkozkulcs;

    if rows_found = 0 then raise notexisterror;
    else
        delete from meroeszkoz where eszkozkulcs = p_eszkozkulcs;
    end if;
exception
    when notexisterror then
        dbms_output.put_line('Ilyen azonosítójú termék nem létezik!');
end;
```

## Tárolt függvény adott rekord mezőinek lekérdezésére

```
create or replace function eszkozAdatok(eszkozSzam in Meroeszkoz.eszkozkulcs%TYPE) return
meroeszkoz%ROWTYPE as f_result meroeszkoz%ROWTYPE;
begin
    select * into f_result from meroeszkoz WHERE eszkozkulcs = eszkozSzam;
    return f_result;
end;
```

## Tárolt függvény adott feltételű rekordok aggregált értékének lekérdezésére

```
create or replace function egyEszkozAtlagHomerseklet(eszkozSzam in int) return int as f_result int;
begin
    select avg(homerseklet) into f_result from meresiadatok WHERE eszkozkulcs = eszkozSzam;
    return(f_result);
end;
```

## Tárolt csomag készítése egy tábla funkcióinak összefogására

```
create or replace package EszkozCsomag is
    procedure ujEszkoz(p_eszkozkulcs number, p_gyartas_ido number, p_magassag number,
        p_irszam number, p_varos varchar, p_utca varchar, p_hazszam number, p_tipus varchar, p_gyarto varchar);
    procedure ujEszkoz2(p_eszkozkulcs number, p_gyartas_ido number, p_magassag number,
        p_irszam number, p_varos varchar, p_utca varchar, p_hazszam number, p_tipus varchar, p_gyarto varchar);
    procedure eszkozModositas(p_eszkozkulcs number, p_gyartas_ido number,
        p_magassag number, p_irszam number, p_varos varchar, p_utca varchar, p_hazszam number,
        p_tipus varchar, p_gyarto varchar);
    procedure Eszkoztorles(p_eszkozkulcs number);
    function eszkozAdatok(eszkozSzam in Meroeszkoz.eszkozkulcs%TYPE) return meroeszkoz%ROWTYPE;
end EszkozCsomag;

create or replace package body EszkozCsomag is
    procedure ujEszkoz(p_eszkozkulcs number, p_gyartas_ido number, p_magassag number,
        p_irszam number, p_varos varchar, p_utca varchar, p_hazszam number, p_tipus varchar, p_gyarto varchar) is
        exsisterror exception;
        rows_found number;
    begin
        select count(*)
        into rows_found
        from meroeszkoz
        where eszkozkulcs = p_eszkozkulcs;

        if rows_found != 0 then raise exsisterror;
        else
            insert into meroeszkoz values(p_eszkozkulcs, p_gyartas_ido, p_magassag, p_irszam, p_varos,
                p_utca, p_hazszam, p_tipus, p_gyarto);
        end if;
    exception
        when exsisterror then
            dbms_output.put_line('Ez a mer?eszköz már létezik!');
    end;

    procedure eszkozModositas(p_eszkozkulcs number, p_gyartas_ido number,
        p_magassag number, p_irszam number, p_varos varchar, p_utca varchar, p_hazszam number,
        p_tipus varchar, p_gyarto varchar) is
        notexsisterror exception;
        rows_found number;
    begin
        select count(*)
        into rows_found
        from meroeszkoz
        where eszkozkulcs = p_eszkozkulcs;

        if rows_found = 0 then raise notexsisterror;
        else
            update meroeszkoz set gyartas_ido = p_gyartas_ido, magassag = p_magassag,
                irszam = p_irszam, varos = p_varos, utca = p_utca, hazszam = p_hazszam,
                tipus = p_tipus, gyarto = p_gyarto where eszkozkulcs = p_eszkozkulcs;
        end if;
    exception
        when notexsisterror then
            dbms_output.put_line('Ilyen azonosítójú mer?eszköz nem létezik!');
    end;
```



```

procedure ujEszkoz2(p_gyartas_ido number, p_magassag number,
p_irszam number, p_varos varchar, p_utca varchar, p_hazszam number, p_tpus varchar, p_gyarto varchar) is
Begin
    insert into meroeszkoz values(null, p_gyartas_ido, p_magassag, p_irszam, p_varos,
    p_utca, p_hazszam, p_tpus, p_gyarto);
End;

procedure Eszkoztorles(p_eszkozkulcs number) is
notexisterror exception;
rows_found number;
begin
    select count(*)
    into rows_found
    from meroeszkoz
    where eszkozkulcs = p_eszkozkulcs;

    if rows_found = 0 then raise notexisterror;
    else
        delete from meroeszkoz where eszkozkulcs = p_eszkozkulcs;
    end if;
exception
    when notexisterror then
        dbms_output.put_line('Ilyen azonosítójú termék nem létezik!');
end;

function eszkozAdatok(eszkozSzam in Meroeszkoz.eszkozkulcs%TYPE) return
meroeszkoz%ROWTYPE as f_result meroeszkoz%ROWTYPE;
begin
    select * into f_result from meroeszkoz WHERE eszkozkulcs = eszkozSzam;
    return f_result;
end;
end package;

```

## Trigger készítése módosítási események naplózására

```

CREATE OR REPLACE TRIGGER MeresiAdatMod BEFORE UPDATE on meresiadatok for each row
Begin
    insert into naplo VALUES('Módosítás',
:new.meresiadatkulcs||'_'||old.meresiadatkulcs
||','||:new.homerseklet||'_'||old.homerseklet
||','||:new.meres_ideje||'_'||old.meres_ideje
||','||:new.paratartalom||'_'||old.paratartalom
||','||:new.szallop||'_'||old.szallop
||','||:new.kendioxid||'_'||old.kendioxid
||','||:new.szenmonoxid||'_'||old.szenmonoxid
||','||:new.nitrogéndioxid||'_'||old.nitrogéndioxid
||','||:new.eszkozkulcs||'_'||old.eszkozkulcs, CURRENT_TIMESTAMP);
End;

```

## Trigger készítése kulcs érték automatikus megadására

```
CREATE SEQUENCE Meres_seq;

CREATE OR REPLACE TRIGGER EszkoFelvitel
  BEFORE INSERT ON meroeszkoz
  FOR EACH ROW
BEGIN
  SELECT Meres_seq.nextval
  INTO :new.eszkosz kulcs
  FROM dual;
END;

create or replace procedure ujEszkoz2(p_gyartas_ido number, p_magassag number,
p_irszam number, p_varos varchar, p_utca varchar, p_hazszam number, p_tipus varchar, p_gyarto varchar) is
Begin
  insert into meroeszkoz values(null, p_gyartas_ido, p_magassag, p_irszam, p_varos,
    p_utca, p_hazszam, p_tipus, p_gyarto);
End;
```

## Trigger készítése a módosítások kontrollálására

```
create or replace trigger MeresiAdatokFigyelo before update on meresiadatok for each row
begin
  if (:new.homerseklet >= 50 OR :new.homerseklet <= -20) then
    :new.homerseklet := :old.homerseklet;
    dbms_output.put_line('A hőmérséklet módosítás nem lépheti át az 50 és -20 határt!');
  end if;
end;
```