***HAYVANAT BAHÇESİ***

create database ZOO\_DB

use ZOO\_DB

Create table tür(

tür\_id int primary key identity(1,1) NOT NULL,

latin\_ism VARCHAR (50) NOT NULL,

ortak\_isim VARCHAR (50) NOT NULL,

miktar int NOT NULL,

)

Create table hayvan(

hayvan\_id int primary key identity(1,1) NOT NULL,

hayvan\_isim VARCHAR(25) NOT NULL,

dogum\_tarih DATE NOT NULL,

tür\_id int foreign key references tür(tür\_id) NOT NULL,

)

Create table bakýcý(

bakýcý\_id int primary key identity(1,1) NOT NULL,

bakýcýismi VARCHAR (50) NOT NULL,

bakýcýadrs VARCHAR (50) NOT NULL,

bakýcýsehir VARCHAR (50) NOT NULL,

bakýcýdevlet varchar (20) NOT NULL,

)

Create table bölüm(

bölüm\_id int primary key identity(1,1) NOT NULL,

bölüm\_isim VARCHAR(50) NOT NULL,

bölüm\_yer VARCHAR(50) NOT NULL,

bölüm\_taným VARCHAR(75),

)

CREATE TABLE hayvan\_ev (

nakil\_id int primary key identity(1,1) NOT NULL,

nakil\_cins int NOT NULL,

giris\_trh DATE NOT NULL,

cýkýs\_trh DATE NOT NULL,

hayvan\_id int foreign key references hayvan(hayvan\_id) NOT NULL,

bölüm\_id int foreign key references bölüm(bölüm\_id) NOT NULL

)

CREATE TABLE habitat (

habitat\_id int primary key identity(1,1) NOT NULL,

habitat\_bölge VARCHAR(50) NOT NULL,

habitat\_gecici int NOT NULL

)

CREATE TABLE habitat\_bölüm (

habitat\_id int foreign key references habitat(habitat\_id) NOT NULL,

bölüm\_id int foreign key references bölüm(bölüm\_id) NOT NULL

)

CREATE TABLE species\_habitat (

tür\_id int foreign key references tür(tür\_id) NOT NULL,

habitat\_id int foreign key references habitat(habitat\_id) NOT NULL

)

CREATE TABLE tür\_bakýcý (

bakýcý\_id int foreign key references bakýcý(bakýcý\_id) NOT NULL,

tür\_id int foreign key references tür(tür\_id) NOT NULL

)

CREATE TABLE yük (

yük\_id int primary key identity(1,1) NOT NULL,

yük\_trh DATE NOT NULL,

cins int NOT NULL,

bakýcý\_id int foreign key references bakýcý(bakýcý\_id) NOT NULL,

hayvan\_id int foreign key references hayvan(hayvan\_id) NOT NULL

)

CREATE TABLE besin(

besin\_id int primary key identity(1,1) NOT NULL,

besin\_isim VARCHAR(50) NOT NULL,

besin\_zinciri int NOT NULL,

bakýcý\_id int foreign key references bakýcý(bakýcý\_id) NOT NULL

)

CREATE TABLE besin\_liste (

besinliste\_id int primary key identity(1,1) NOT NULL,

zaman\_aralýk int NOT NULL,

besin\_kaynak VARCHAR(75) NOT NULL,

besin\_miktar VARCHAR(75) NOT NULL,

besin\_id int foreign key references besin(besin\_id) NOT NULL,

hayvan\_id int foreign key references hayvan(hayvan\_id) NOT NULL

)

CREATE TABLE problem (

prob\_id int primary key identity(1,1) NOT NULL,

prob\_tip VARCHAR(50) NOT NULL

)

CREATE TABLE hayvan\_cözüm (

cözüm\_id int primary key identity(1,1) NOT NULL,

cözüm\_tip VARCHAR (50) NOT NULL

)

CREATE TABLE hayvan\_problem (

hayvanProb\_id int primary key identity(1,1) NOT NULL,

hayvanProb\_date DATE NOT NULL,

hayvan\_id int foreign key references hayvan(hayvan\_id) NOT NULL,

prob\_id int foreign key references problem(prob\_id) NOT NULL,

cözüm\_id int foreign key references hayvan\_cözüm(cözüm\_id) NOT NULL

)

CREATE TABLE habitat\_cözüm (

habcöz\_id int primary key identity(1,1) NOT NULL,

habcöz\_tip VARCHAR(50) NOT NULL

)

CREATE TABLE bölüm\_problem (

blmProb\_id int primary key identity(1,1) NOT NULL,

blmProb\_date DATE NOT NULL,

prob\_id int foreign key references problem(prob\_id) NOT NULL,

bölüm\_id int foreign key references bölüm(bölüm\_id),

habcöz\_id int foreign key references habitat\_cözüm(habcöz\_id) NOT NULL

)

CREATE TABLE anagörev (

ana\_id int primary key identity(1,1) NOT NULL,

görevad VARCHAR(50) NOT NULL,

anadesc VARCHAR(50) NOT NULL

)

CREATE TABLE isci(

isci\_id int primary key identity(1,1),

isci\_fName VARCHAR(25),

isci\_lName VARCHAR(25)

)

CREATE TABLE calýsanisci (

calýsanisci\_id int primary key identity(1,1) NOT NULL,

isci\_id int foreign key references isci(isci\_id) NOT NULL,

bölüm\_id int foreign key references bölüm(bölüm\_id) NOT NULL,

ana\_id int foreign key references anagörev(ana\_id) NOT NULL,

gün\_maint VARCHAR (20) NOT NULL

)

INSERT INTO tür VALUES ('Pan Troglodytes', 'Common Chimpanzee', 12);

INSERT INTO tür VALUES ('Panthera Tigris Tigris', 'Bengal Tiger', 6);

INSERT INTO tür VALUES ('Panthera Leo', 'Lion', 9);

INSERT INTO tür VALUES ('Ursus Maritimus', 'Polar Bear', 7);

INSERT INTO tür VALUES ('Loxodonta Africana', 'African Bush Elephant', 3);

select \* from tür

INSERT INTO hayvan VALUES('Chimpy', '01/15/2001', 1);

INSERT INTO hayvan VALUES( 'Frisky Tiger', '03/25/2002', 2);

INSERT INTO hayvan VALUES( 'Lame Lion', '11/25/2004',3);

INSERT INTO hayvan VALUES( 'Polorizer', '03/13/2010', 4);

INSERT INTO hayvan VALUES('Egor the Elephant','10/02/2004', 5);

select \* from hayvan

INSERT INTO bakýcý VALUES( 'WallyWorld', '4972 Richison Drive','Milwaukee', 'MT');

INSERT INTO bakýcý VALUES('Animal Food and Stuff', '4780 Berkley Street', 'Philadelphia', 'PA');

INSERT INTO bakýcý VALUES('Random Supplier Person', '4827 Wilkinson Court', 'Kuzey Napoli', 'FL');

INSERT INTO bakýcý VALUES( 'Perez Inc.', '4561 Frank Avenue', 'Springfield', 'MA');

INSERT INTO bakýcý VALUES('Joyce Rides', '523 Boring Lane', 'San Francisco', 'CA');

select \* from bakýcý

INSERT INTO bölüm VALUES ('Regenstein Wolf Woods', 'Orman', 'aðaçlýk orman');

INSERT INTO bölüm VALUES ('The Swamp', 'Bataklýk', 'Düþük arazi bataklýk');

INSERT INTO bölüm VALUES ('Australia House', 'tarla', 'Düz Arazi');

INSERT INTO bölüm VALUES ('Hoofed Animals', 'tarla', 'Düz Arazi');

INSERT INTO bölüm VALUES ('Salt Creek Wilderness', 'Tuzlu Su ile aðaçlýk', 'Bir tuzlu su birikintisi ile bazý ahþap alaný');

select \* from bölüm

INSERT INTO hayvan\_ev Values(8,'2005-12-12','2007-08-02', 3, 3 );

INSERT INTO hayvan\_ev Values( 72,'2000-06-17','2012-12-21', 4, 4);

INSERT INTO hayvan\_ev Values(99,'2007-05-22','2008-10-18', 5, 5);

select \* from hayvan\_ev

INSERT INTO habitat VALUES ('su', '88');

INSERT INTO habitat VALUES ('sulak', '65');

INSERT INTO habitat VALUES ('Bataklýk', '60');

INSERT INTO habitat VALUES ('Tropical', '90');

INSERT INTO habitat VALUES ('Çöl', '99');

select \* from habitat

INSERT INTO habitat\_bölüm VALUES(1,1);

INSERT INTO habitat\_bölüm VALUES(2,2);

INSERT INTO habitat\_bölüm VALUES(3,3);

INSERT INTO habitat\_bölüm VALUES(4,4);

INSERT INTO habitat\_bölüm VALUES(5,5);

select \* from habitat\_bölüm

INSERT INTO species\_habitat VALUES (1, 1);

INSERT INTO species\_habitat VALUES (2, 2);

INSERT INTO species\_habitat VALUES (3,3);

INSERT INTO species\_habitat VALUES (4, 4);

INSERT INTO species\_habitat VALUES (5,5);

select \* from species\_habitat

INSERT INTO tür\_bakýcý VALUES(1, 1);

INSERT INTO tür\_bakýcý VALUES(2, 2);

INSERT INTO tür\_bakýcý VALUES(3, 3);

INSERT INTO tür\_bakýcý VALUES(4, 4);

INSERT INTO tür\_bakýcý VALUES(5, 5);

select \* from tür\_bakýcý

INSERT INTO yük VALUES('2010-04-29', 1, 1, 3);

INSERT INTO yük VALUES('2010-04-14', 2, 2, 4);

INSERT INTO yük VALUES('2010-05-13', 3, 3, 5);

INSERT INTO yük VALUES('2010-05-28', 4, 4, 6);

INSERT INTO yük VALUES('2010-06-12', 5, 5, 7);

select \* from yük

INSERT INTO besin VALUES ('yulaf', '8000', '1');

INSERT INTO besin VALUES ('yer fýstýðý', '6000', '2');

INSERT INTO besin VALUES ('muz', '6000', '3');

INSERT INTO besin VALUES ('havuçlar', '3000', '4');

INSERT INTO besin VALUES ('ringa', '750', '5');

select \* from besin

INSERT INTO besin\_liste VALUES ('3' , 'Depo 2', '3 pounds', '12', '3');

INSERT INTO besin\_liste VALUES ('5' , 'Depo 5', '10', '13', '4');

INSERT INTO besin\_liste VALUES ('1' , 'Depo 2', '15', '14', '5');

INSERT INTO besin\_liste VALUES ('4' , 'Depo 3', '2 pounds', '15', '6');

INSERT INTO besin\_liste VALUES ('1' , 'Depo 1', '25', '16', '7');

select \* from besin\_liste

INSERT INTO problem VALUES ('Týbbi');

INSERT INTO problem VALUES ('Bakým');

INSERT INTO problem VALUES ('hayvan Davranýþý');

INSERT INTO problem VALUES ('Týbbi');

INSERT INTO problem VALUES ('Týbbi');

select \* from problem

INSERT INTO hayvan\_cözüm VALUES('Bir hafta Antibiyotikler');

INSERT INTO hayvan\_cözüm VALUES('izolasyon');

INSERT INTO hayvan\_cözüm VALUES('1 Ay Antibiyotikler');

INSERT INTO hayvan\_cözüm VALUES('izolasyon');

INSERT INTO hayvan\_cözüm VALUES('Bir hafta Antibiyotikler');

select \* from hayvan\_cözüm

INSERT INTO habitat\_cözüm VALUES('kuru temizlik');

INSERT INTO habitat\_cözüm VALUES('izolasyon');

INSERT INTO habitat\_cözüm VALUES('kapatma');

INSERT INTO habitat\_cözüm VALUES('döþeme');

INSERT INTO habitat\_cözüm VALUES('kaplama');

select \* from habitat\_cözüm

INSERT INTO hayvan\_problem VALUES('12/05/2010', 3, 1, 1);

INSERT INTO hayvan\_problem VALUES('10/10/2009', 4, 2, 2);

INSERT INTO hayvan\_problem VALUES('12/05/2010', 5, 3, 3);

INSERT INTO hayvan\_problem VALUES('6/15/2008', 6, 4, 4);

INSERT INTO hayvan\_problem VALUES('8/01/2009', 7, 5, 5);

select \* from hayvan\_problem

INSERT INTO bölüm\_problem VALUES('1/05/2010', 1, 1, 1);

INSERT INTO bölüm\_problem VALUES('10/01/2009', 2, 2, 2);

INSERT INTO bölüm\_problem VALUES('2/05/2010', 3, 3, 3);

INSERT INTO bölüm\_problem VALUES('6/01/2008',4, 4,4);

INSERT INTO bölüm\_problem VALUES('8/01/2009', 5, 5, 5);

select \* from bölüm\_problem

INSERT INTO anagörev VALUES ('atýk Pickup', 'Iskarta hayvan atýklarý');

INSERT INTO anagörev VALUES ('su Dolum', 'Hayvan containters doldurun');

INSERT INTO anagörev VALUES ('Refah Kontrol', 'tüm hayvanlar kontrol');

INSERT INTO anagörev VALUES ('Aydýnlatma', 'Ampuller deðiþtirin');

INSERT INTO anagörev VALUES ('dezenfekte', 'tüm sergiler kontrol');

select \* from anagörev

INSERT INTO isci VALUES( 'Ahmet ', 'Dursun');

INSERT INTO isci VALUES('Bekir', 'kurt');

INSERT INTO isci VALUES('Hakan', 'Atalay');

INSERT INTO isci VALUES('Mehmet', 'konmaz');

INSERT INTO isci VALUES('Hasan', 'Sabbah');

select \* from isci

INSERT INTO calýsanisci VALUES (1, 1, 1, '10/10/2010');

INSERT INTO calýsanisci VALUES (2, 2, 2 ,'1/05/2009');

INSERT INTO calýsanisci VALUES ( 3, 3, 3,'12/25/2009');

INSERT INTO calýsanisci VALUES (4, 4, 4,'04/09/2010');

INSERT INTO calýsanisci VALUES (5, 5, 5,'05/26/2010');

select \* from calýsanisci

use ZOO\_DB

--1--Problemi 'Týbbi olanlarýn tipi'

SELECT problem.prob\_tip FROM problem WHERE prob\_tip = 'Týbbi'

--2--Hangi hayvana hangi bakýcý bakýyor?

select hayvan.hayvan\_isim,bakýcý.bakýcýismi from hayvan,bakýcý,yük where hayvan.hayvan\_id=yük.hayvan\_id and bakýcý.bakýcý\_id=yük.bakýcý\_id

--3--rooney'in baktýðý hayvanýn besin zaman aralýðý nedir?

select bakýcý.bakýcýismi,hayvan.hayvan\_isim,besin\_liste.zaman\_aralýk from bakýcý,hayvan,besin,besin\_liste where

besin.bakýcý\_id=bakýcý.bakýcý\_id and besin.besin\_id=besin\_liste.besin\_id and hayvan.hayvan\_id=besin\_liste.hayvan\_id

and bakýcý.bakýcýismi='Rooney'

--4--2010-03-13 doðumlu taranihinden küçük ve isim sýrasýna göre olan hayvanlarý listele?

select hayvan\_isim,hayvan\_id,dogum\_tarih from hayvan where hayvan.dogum\_tarih<'2010-03-13' order by hayvan\_isim ASC

--5--Ýstanbul'da ikamet eden ve adýnýn içerisinde a harfi içeren bakýcýlarý listelemek için gerekli SQL ifadesini yazýnýz.

select \* from bakýcý where bakýcýismi like '%a%' and bakýcýsehir like'%istanbul%'

--6 hayvanýn problem tipi ne?

create procedure sinan

as

begin

select hayvan.hayvan\_isim, prob\_tip from problem,hayvan\_problem,hayvan where problem.prob\_id=hayvan\_problem.prob\_id and

hayvan.hayvan\_id=hayvan\_problem.hayvan\_id;

end

exec sinan

--roney herbir hayvan türünden kaç tane hayvana bakýyor

select bakýcý.bakýcýismi,count(tür.miktar)from hayvan,tür,tür\_bakýcý,bakýcý where hayvan.tür\_id=tür.tür\_id

and bakýcý.bakýcý\_id=tür\_bakýcý.bakýcý\_id and tür\_bakýcý.tür\_id=tür.tür\_id and bakýcý.bakýcýismi='Rooney' group by

bakýcý.bakýcýismi

***GALERİ***

create database galeri

use galeri

create table arac (

aracno int primary key identity(1,1),

model varchar(50),

plaka varchar(20) ,

fiyat int

)

insert into arac(model,plaka,fiyat,marka) values(2004,'60tt6060',16000,'Fiat Marea')

insert into arac(model,plaka,fiyat,marka) values(2000,'60tt6061',14000,'Renault Megane')

insert into arac(model,plaka,fiyat,marka) values(2007,'60tt6062',28000,'Ford Focus')

insert into arac(model,plaka,fiyat,marka) values(2005,'60tt6063',26000,'Volkswagen Golf')

insert into arac(model,plaka,fiyat,marka) values(1998,'60tt6064',9000,'Opel Astra')

create table musteri(

mno int primary key identity(1,1),

madi varchar(20),

msoyadi varchar(20),

madres varchar (50),

mtelefon varchar(11)

)

create table satýs(

satno int primary key identity(1,1),

mno int foreign key references musteri(mno),

aracno int foreign key references arac(aracno) ,

sat\_tarih date,

sfiyat int

)

insert into satýs(sat\_tarih,sfiyat) values('04/05/2010',17000)

insert into satýs(sat\_tarih,sfiyat) values('01/06/2010',11500)

insert into satýs(sat\_tarih,sfiyat) values('06/15/2010',27000)

insert into satýs(sat\_tarih,sfiyat) values('02/07/2010',17500)

create table alim(

alimno int primary key identity(1,1),

mno int foreign key references musteri(mno),

aracno int foreign key references arac(aracno),

alim\_tarih date,

afiyat int

)

alter table arac add marka varchar(50)

insert into musteri(madi,msoyadi,madres,mtelefon) values('Turgut','Özseven','Turhal Tokat','0531654849')

insert into musteri(madi,msoyadi,madres,mtelefon) values('Mustafa','Çaðlayan','Meran Konya','025164854')

insert into musteri(madi,msoyadi,madres,mtelefon) values('Ahmet','Kara','Zile Tokat','014846531')

insert into musteri(madi,msoyadi,madres,mtelefon) values('Murat','Beyaz','Turmahal Tokat','5464123198')

insert into musteri(madi,msoyadi,madres,mtelefon) values('Bülent','Ayar','Turhal Tokat','897984564')

insert into alim(alim\_tarih,afiyat) values('08/02/2010',15000)

insert into alim(alim\_tarih,afiyat) values('12/04/2010',15500)

insert into alim(alim\_tarih,afiyat) values('05/04/2010',9500)

insert into alim(alim\_tarih,afiyat) values('05/05/2010',14000)

insert into alim(alim\_tarih,afiyat) values('02/08/2010',26000)

select \* from arac where model<2004

select \* from arac where model>2000 and model<2009

Select \* from satýs where sfiyat>10000 and sat\_tarih>'01.05.2010'

Select \* from musteri where madi LIKE ('%y%') and madres like ('%turhan%')

Select aracno, model, plaka, fiyat from arac where marka='Opel' or marka='Fiat' and model>2000

select \* from musteri order by madi desc

select \* from arac order by model desc ,fiyat asc

select satýs.aracno,satýs.sat\_tarih,satýs.sfiyat as 'satýs fiyati' from satýs

select \* from arac where fiyat>1200 order by fiyat desc

select \* from satýs where sfiyat>10000 or mno=4

SELECT \* FROM satýs WHERE sat\_tarih>='2010-04-01'

SELECT alim\_tarih,afiyat,aracno FROM alim WHERE alim\_tarih>='2010-02-01' AND alim\_tarih>='2010-06-01' AND afiyat>=10000 ORDER BY alim\_tarih,afiyat DESC

SELECT \* FROM satýs WHERE sfiyat>=10000 AND sfiyat<=20000 AND mno=4 OR mno=2

SELECT DISTINCT musteri.mno,aracno,sat\_tarih,satno,sfiyat FROM satýs,musteri WHERE musteri.madres LIKE 'Turhal/Tokat' AND musteri.mno=satýs.mno

SELECT \* FROM musteri WHERE madres LIKE '%tokat%' AND mtelefon LIKE '%6%'

SELECT madres,mtelefon FROM musteri WHERE madi LIKE 'a%'

SELECT \* FROM arac WHERE marka LIKE '%Opel%' OR marka LIKE '%Fiat%' ORDER BY marka

SELECT \* FROM arac WHERE model LIKE '%1998%' OR model LIKE '%2005%' OR model LIKE '%2007%'

***OKUL SORGU***

use okul

select \* from ogrenci

insert into ogrenci values('Hasan','Celik',1,23,'Bilecik')

--group by

select ogrenci.oadres,count(ogrenci.oadres) as 'sayisi' from ogrenci group by ogrenci.oadres

select \* from puan

select puan.dno,min(puan.vize) as'ENK',max(puan.vize) as 'enb' from puan group by puan.dno

--en küçük vizeye sahip olunan dersten daha büyük vizeye sahip dersler

select puan.dno,puan.vize,min(puan.vize) 'Enk' from puan group by puan.dno,puan.vize

--herbir bölümden herbir dersi almayan kaç ogrenci bulunur

select bolum.badi,ders.dadi,count(ogrenci.ono) as 'dersi alamayan sayisi' from bolum,ogrenci,ders,puan where ogrenci.bno=bolum.bno and bolum.bno=ders.bno

and ogrenci.ono not in (select puan.ono from puan) and puan.dno=ders.dno group by bolum.badi,ders.dadi

---inner join

select ogrenci.oadi,ogrenci.osoyadi,bolum.badi from ogrenci,bolum where ogrenci.bno=bolum.bno

select ogrenci.oadi,ogrenci.osoyadi,bolum.badi from ogrenci inner join bolum on ogrenci.bno=bolum.bno

--iç içe sorgu

select ogrenci.oadi,ogrenci.osoyadi,bolum.badi from ogrenci,bolum where ogrenci.bno in (select bno from bolum)

--bir yarýþma sitesi için tasarlanan database aþþaðýda.bun database göre kodlarla tasarým yapýnýz ve sorularý cevaplayýnýz.

create database yarisma

use yarisma

Create Table il

(

ilpno int Primary Key,

iladi varchar(30),

)

Create Table uye

(

uno int Primary Key identity (1,1),

uadi varchar(30)NOT NULL,

usoyadi varchar(30)NOT NULL,

ubtrh datetime Null,

ilpno int foreign key references il(ilpno),

usifre varchar(30)NOT NULL DEFAULT('123abc'),

girsay int Default(0),

sonpuan int

)

Create Table sorutur

(

stno int Primary Key identity (1,1),

stadi varchar(30)NOT NULL

)

Create Table soru

(

sno int Primary Key identity (1,1),

sbaslik varchar(30)NOT NULL,

c1 varchar(2),

c2 varchar(2),

c3 varchar(2),

c4 varchar(2),

c5 varchar(2),

stno int foreign key references sorutur(stno),

spuan int NOT NULL,

uno int foreign key references uye(uno)

)

Create Table Ýstatistik

(

uno int foreign key references uye(uno),

aldtrh date,

toplampuan int

)

insert into il values(1,'Adana')

insert into il values(2,'Adýyaman')

insert into il values(3,'Afyon')

insert into il values(4,'Aðrý')

insert into il values(5,'Amasya')

insert into il values(6,'Ankara')

insert into il values(7,'Antalya')

insert into uye values('Sinan','Yazici',02/01/2017,1,'123abc',20,40)

select \* from uye

--herbir üye her bir soru türünden kaç soru göndermiþtir

select uye.uadi,count(sorutur.stno) from soru,sorutur,uye where sorutur.stno=soru.stno and uye.uno=soru.uno group by sorutur.stadi

--gönderdiði sorularýn puan ort en yüksek olan ancak hiç biliþim konusunda soru göndermeyen üyelerin adýný soyadýný

--hangi ilden yarýsmaya katýldýðýný

--ve bu güne kadar kaç kez yarýstýðýný ve aldýðý toplam puaný sql yaz

select uye.uadi,uye.usoyadi,il.iladi,uye.girsay,Ýstatistik.toplampuan,avg(soru.spuan),count(uye.uno) from uye,il,Ýstatistik,soru ,sorutur

where uye.uno=soru.uno and uye.ilpno=il.ilpno and uye.uno=istatistik.uno and soru.stno=sorutur.stno and uye.uno in

(select soru.uno from soru) group by uye.uadi,uye.usoyadi,il.iladi,uye.girsay,Ýstatistik.toplampuan

--1-hiç soru göndermeyen ancak yarýþmaya katýlan üyelerin bugüne kadar aldýklarý ortalama puanlarý listeleyiniz.

--2-yarýþmaya herbir ilden herbir soru türünden kaçar tane soru gönderilmiþtir.

--3--yarýþamada bu güne kadar en yüksek puano almýþ üye yarýþmaya hangi soru türünden kaç soru göndermiþtir.

--4--yarýþmada aldýðý son puan bugüne kadar almýþ olduðu toplam puanlarý ortalamasýndan daha büyük olan üyelerin hangileri en yüksek

--puaný sahip soruyu yanýþ cevaplamýþtýr.

***WEB YARİSMA***

create database Webyarisma

use webyarisma

create table il

(

ilpno int primary key check (ilpno >=1 and ilpno <=81),

iladi varchar(50) not null,

)

create table uye

(

uno int primary key identity(1,1),

uadi varchar(50) not null,

usoyadi varchar(50) not null,

utrh datetime,

usifre varchar(8),

ugirissayisi int,

ilpno int foreign key references il(ilpno)

)

create table sorutur

(

stno int primary key identity(1,1),

stadi varchar(100) not null

)

create table soru

(

sno int primary key identity(1,1),

sbaslik varchar(255) not null,

c1 varchar(100) not null,

c2 varchar(100) not null,

c3 varchar(100) not null,

c4 varchar(100) not null,

dc varchar(5),

stno int foreign key references sorutur(stno),

spuan int,

uno int foreign key references uye(uno)

)

create table istatistik

(

uno int foreign key references uye(uno),

istrh datetime,

toplampuan int

)

---3

select sorutur.stadi,count(sorutur.stno) as 'soru sayisi' from sorutur,soru where sorutur.stno=soru.stno group by sorutur.stadi

--2

select il.iladi,sorutur.stno,count(sorutur.stno) as 'adet' from sorutur,il,uye where uno>100 group by il.ilpno

***OKUL SORGU 2***

use okul

select ogrenci.oadi,ogrenci.osoyadi,(select bolum.badi from bolum where bolum.bno=ogrenci.bno) as ' Bolum Adi ' from ogrenci

use yarisma

select max(sorutur.stno) as'max soru sayisi' from(select sorutur.stadi),count(soru.sno) as stno from soru,sorutur where soru.stno=sorutur.stno

group by sorutur.stadi) sorutur

--Views(Görünümler):

--gerçekte var olmayan sanal tablolardýr.select ifadesi ile sanal tablolar üzerinde gerçek tablolar üzerinde iþlem yapabilmek için kullanýlýr.

--temel amacý güvenliktir.Neseneler sql üzerinde create ile oluþturulur alter ile güncellenir drop ile silinir.

--oluþturma : create view v\_adi(sütün isimleri) as

use okul

create view liste1(oadi,osoyadi,badi)

as

select ogrenci.oadi,ogrenci.osoyadi,bolum.badi from ogrenci,bolum where ogrenci.bno=bolum.bno

select \* from liste1

------------------------------------

create view liste2

with encryption

as

select \* from liste1

-----TSQ----------

declare @s1 int,@s2 int,@tpl int

set @s1=10

set @s2=20

set @tpl=@s1+@s2

print('Toplam: '+CAST(@tpl as Varchar(10)))

select @@Error

select @@Servername

select @@version

select @@language

select @@SPID

select @@Max\_Connections

select @@RowCount

select @@Connections

declare @ogr Table(ono int,

oadi varchar(20),osoyadi varchar(20),bno int)

insert into @ogr values(1,'Elif','Cetin',3)

select \* from @ogr

--öðrenci tablosuna yeni kayýt eklerken

--Eklenen kayýtý bilgilerini kullanýcýya

--gösteren output

--insert output

select \* from ogrenci

declare @ogr Table(

ono int,

oadi varchar(20),osoyadi varchar(20),bno int,oyas int,oadres varchar(50)

)

insert into ogrenci

output inserted.ono,inserted.oadi,inserted.osoyadi,inserted.oyas,inserted.oadres into @ogr

Values('Elif','Gül',1,20,'Bilecik')

select \* from ogr

-- okul database üzerinde hem vize hemde final puanlarý

--yanlýþ girilen bir öðrencinin notlarýnýn güncellenmesi esnansýnda önceki ve sonraki

--puan bilgilerini kullanýcýya gösteren output

use okul

declare @puan Table(ono int,dno int,ev int,ef int,yv int,yf int)

update puan set vize=20,final=40

output deleted.ono,deleted.dno,deleted.vize,deleted.final,inserted.vize,inserted.final into @puan

where ono=1

select ono,dno,ev as'Eski vize',ef as 'Eski final0',yv as'yeni vize',yf as 'yeni final' from @puan

--veritabaný dersini almayan öðrencileri veritabanýndan silerken silinen kayýtlarý kullanýcya gösteren output

declare @ders table( ono int,

oadi varchar(20),osoyadi varchar(20),bno int,oyas int,oadres varchar(50))

declare @almayan table(dno int,dadi varchar(20),bno int)

delete from ders,ogrenci

output deleted.ono,deleted.oadi,deleted.osoyadi,deleted.bno,deleted.oyas,deleted.oadres into @ders, @almayan

where ders.dadi dadi Not in dadi='Veritabaný'

select \* from @ders,@almayan

use okul

select \* from puan

***KARIŞIK***

create database uygulama1

use uygulama1

create table OGRENCILER (

ogrno int primary key identity(100,1),

ograd varchar(5),

ogrsoyad varchar(15),

veliadsoyad varchar(20),

adres varchar(50),

sinif char(3) constraint cp\_sýnýf check(sinif like'[10-12][A-z]'),

dogumtar date,

alan varchar(20) constraint cp\_alan check(alan in('fen','TM','SOS')),

)

create table URUNLER (

urunný int primary key identity(1,1),

urunad varchar(15),

uruntaným varchar(50),

maliyetfiyat money,

karoran real constraint ck\_karoran check(karoran>=0.05 and karoran<=0.25),

kdv real constraint ck\_kdv check(kdv>=0.03 and kdv<=0.25),

kolisekli varchar(10) constraint cp\_kolisekli check(kolisekli in('adet','kg','deste','litre','seri'))

)

create table ARABALAR (

aracno varchar(4)primary key constraint cp\_aracno check(aracno like'[ORF][0-9][0-9][0-9]' ),

marka varchar(7) constraint cp\_marka check(marka in('opel','renault','fors')),

model varchar(10),

trafikyil date constraint cp\_trafikyil check(trafikyil like'[2][0][0][0-9]'),

renk varchar(10),

yakit varchar(10) constraint cp\_yakit check(yakit in('BENZÝN','DIZEL','LPG','KARMA')),

kilo int,

)

--1.OGRENCILER tablosuna 3 adet tam dolu kayýt giriþi yapýnýz. (Ayrý insert’ler ile ve tek insert ile)

alter table OGRENCILER add kangurup char(3) constraint cp\_kangurup check(kangurup in('A+','A-','B+','B-','AB-','AB+','O+','O-'))

--2.OGRENCILER tablosundaki “ograd” alanýnýn uzunluðunu 15 karakterlik hale getiriniz.

alter table OGRENCILER alter column ograd varchar(15)

--3. OGRENCILER tablosuna “dogumyer” alanýný ekleyiniz.

alter table OGRENCILER add dogumyer varchar(15)

--4. URUNLER tablosundaki “uruntaným” alanýnýn uzunluðunu 40 karakterlik hale getiriniz.

alter table URUNLER alter column uruntaným varchar(40)

--5. URUNLER tablosundaki “kolisekli” alanýný siliniz.

alter table URUNLER drop column kolisekli

--6. URUNLER tablosuna “renk” ve “kilo” alanlarýný ekleyiniz. (kilo>0 olmalý)

alter table URUNLER add renk varchar(10),kilo int

--7. URUNLER tablosundaki “kilo” alanýný siliniz.

alter table URUNLER drop column kilo

--8. ARABALAR tablosundaki “aracno” alanýna “H” seçeneði de girilebilir hale getiriniz.

alter table ARABALAR alter column aracno varchar(4)primary key constraint cp\_aracno check(aracno like'[ORHF][0-9][0-9][0-9]' )

--9. ARABALAR tablosundaki “marka” alanýna “HYUNDAI” seçeneðini de ekleyiniz.

alter table ARABALAR alter column marka varchar(7) constraint cp\_marka check(marka in('opel','renault','fors','HYUNDAI'))

--10. ARABALAR tablosundaki kilo alanýný siliniz.