//=================================================================

//                     TEMEL KOMUTLAR

//=================================================================

//-----------------------------------------------------------------

// 1) Varolan veritabanlarini listelemek icin.

// //-----------------------------------------------------------------

// show dbs;

//-----------------------------------------------------------------

// 2) Veritabani tanimlamak veya icine girmek icin

//-----------------------------------------------------------------

use("kisiler");

//-----------------------------------------------------------------

// 3) Icinde bulundugumuz (aktif) Veritabanini gormek icin

// //----------------------------------------------------------------

// db;

//-----------------------------------------------------------------

// 4) Aktif olan veritabanini silmek icin

//-----------------------------------------------------------------

// db.dropDatabase();

//-----------------------------------------------------------------

// 5) Aktif olan veritabaninda bir collection olusturmak icin

// //-----------------------------------------------------------------

// db.createCollection("ogrenciler");

//-----------------------------------------------------------------

// 6) Bir collection'ı silmek icin

//-----------------------------------------------------------------

// db.ogrenciler.drop();

//-----------------------------------------------------------------

// 7) collection'a bir dokuman eklemek icin

//-----------------------------------------------------------------

// db.ogrenciler.insertOne({"ad": "Murat", "soyad": "Cakir", "yas": 41});

//-----------------------------------------------------------------

// 8) Collection'daki tum dokumanlari listelemek (SORGU-QUERY)icin.

//-----------------------------------------------------------------

// db.ogrenciler.find();

//-----------------------------------------------------------------

// 9) Collection'a birden fazla dokuman eklemek icin

//-----------------------------------------------------------------

// db.ogrenciler.insertMany([

//  {"ad": "Ahmet", "soyad": "Can", "yas": 21},

//  {"ad": "Ali", "soyad": "Can", "yas": 51},

//  {"ad": "Veli", "soyad": "Gel", "yas": 22}] );

// db.ogrenciler.find();

//-----------------------------------------------------------------

// 10) Bir değişken (Tarih)  eklmeek için

// //-----------------------------------------------------------------

// var tarih = new Date();

// db.ogrenciler.insertOne({"suanki\_tarih": tarih});

// use ("kisiler")

// db.ogrenciler.find();

*var* tarih1 = new ISODate ();

*var* ay = tarih1.getMonth();

db.ogrenciler.insertOne({"ad": "Can", "Soyad":"Kan","yas":20,"dogumAyi":ay})

db.ogrenciler.find();

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

////=================================================================

////                      SORGULAR (QUERIES)

////=================================================================

////-----------------------------------------------------------------

//// 1) "urunler" adinda bir veritabani ve icerisinde "fiyatlar"

//// adinda bir collection tanimlayalim.

////-----------------------------------------------------------------

use("urunler");

// db.createCollection("fiyatlar");

////-----------------------------------------------------------------

//// 2) asagidaki dokumanlari collection'a kaydedelim ve listeleyelim.

// ////-----------------------------------------------------------------

// db.fiyatlar.insertMany([

// {"urunAdi":"TV", "fiyat": 5000, "vergi": 10 },

// {"urunAdi":"Telefon", "fiyat": 3500, "vergi": 20 },

// {"urunAdi":"Utu", "fiyat": 1500, "vergi": 5 },

// {"urunAdi":"Buzdolabi", "fiyat": 4000, "vergi": 10 } ] );

// db.fiyatlar.find();

////-----------------------------------------------------------------

//// 3) ilk üc dokumani sorgulayalim --- limit() metodu

////-----------------------------------------------------------------

db.fiyatlar.find().limit(3);

////-----------------------------------------------------------------

//// 4) Sadece 2. ve 3. dokumanlari sorgulayalim

////-----------------------------------------------------------------

db.fiyatlar.find().skip(1).limit(2);

////-----------------------------------------------------------------

//// 5) Birinci dokumani sorgulayalim

// ////-----------------------------------------------------------------

// db.fiyatlar.find().limit(1);

db.fiyatlar.findOne();

////=================================================================

////       LOGICAL QUERY OPERATOR ($and ,$or, $not,$nor)

////

//// SYNTAX: { $and: [ {ifade1}, {ifade2} , ... , {ifadeN} ] }

////         { field: { $not: { <operator-ifade> } } }

////=================================================================

////-----------------------------------------------------------------

//// 5) Collection asagidaki dokumani ekleyelim.

// ////-----------------------------------------------------------------

// db.fiyatlar.insertOne({"urunAdi":"Koltuk", "fiyat": 5000, "vergi": 10});

db.fiyatlar.find();

////-----------------------------------------------------------------

//// 6) fiyat=5000 olan urunleri listeleyiniz.

////-----------------------------------------------------------------

// db.fiyatlar.find({ "fiyat": 5000 });

db.fiyatlar.find({"fiyat": {$eq: 5000}});

////-----------------------------------------------------------------

//// 7) fiyat=5000 olan ve urunAdi="TV" olan dokumanlari listeleyiniz.

////-----------------------------------------------------------------

db.fiyatlar.find( {$and: [{"fiyat":5000}, {"urunAdi":"TV"}]});