

JDBC

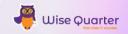
Java Database Connectivity (Java Veritabanı Bağlantısı)

1. Ders

Ayşe Sezen

Full-Stack Automation Engineer





BU DERS NELER ÖĞRENECEĞİZ?

- 1) JDBC Nedir?
- 2) JDBC Kavramları
- 3) JDBC'e Giriş



JDBC NEDIR?



Java oldukça standartlaşmıştır, Ancak SQL'in farklı bir çok versiyonu bulunmaktadır. JDBC, Java ortamında SQL veritabanına erişim için bir araçtır.

JDBC API'si, ilgili veritabanının JDBC sürücüleri yardımıyla

- Veritabanı ile bağlantı kurar.
- SQL ifadelerini gönderir.
- Sonuçları işler.

fyin⊗ **@** /wisequarter



Wise Quarter

JDBC'DE TANIMLI SINIFLAR (CLASSES)

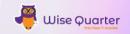
- → Driver
 - getConnection
- → Bağlantı
 - createStatement
- → SQL ifadesi çalıştırma
 - execute, executeQuery, executeBatch, executeUpdate
- → Sonuçları alma
 - next, getString, getInt, getDate, getMetaData
- → ResultSetMetadata
 - getColumnCount, getColumnName, getColumnType







- 1. Kullanılacak Veritabanı için doğru sürücüyü ekle,
- 2. Veritabanı ile iletişimi başlat,
- 3. SQL ifadeleri oluştur ve çalıştır (Select,insert/update/delete)
- 4. Gelen sonuçları işle, kaydet vb.,
- 5. Veritabanı bağlantısını bitir.



1. DRIVER (SÜRÜCÜ) EKLEME

- → Driver'ı eklemek için aşağıdaki metot kullanılır.
 - Class.forName("driver class")
- → Driver, seçilen veritabanına göre değişmektedir.
 - oracle.jdbc.driver.OracleDriver
 - mysql.jdbc.Driver
 - org.postgresql.Driver
- → ÖRNEK:

Class.forName("oracle.jdbc.driver.OracleDriver");

2. VERİTABANI BAĞLANTISI OLUŞTURMA

- → Veritabanı ile bağlantı oluşturabilmek için getConnection() metodu kullanılır.
- → Bu metot, bir Connection nesnesi döndürür.
- DriverManager.getConnection(url, username, password)

ÖRNEK 1:

"jdbc:mysql://localhost:3306/sys?serverTimezone=UTC", "root", "1234"

ÖRNEK 2:

"jdbc:oracle:thin:@localhost:1521/ORCLCDB.localdomain","murat","1234"

ÖRNEK 3:

Connection con =

DriverManager.getConnection("jdbc:mysql://localhost:3306/sys?serverTimezone=UTC", "root", "1234");



www.wisequarter.com

3. SQL İFADESİ OLUŞTURMA VE ÇALIŞTIRMA

- A) Bir SQL ifade oluşturmak için createStatement() metodu kullanılır.
 - Statement st = con.createStatement();
 - st, SQL komutlarını yazabilmemize imkan tanıyan bir Statement nesnesidir.
- B) SQL ifadelerini çalıştırmak için ise aşağıdaki Metotlar kullanılabilir.
 - executeQuery() => SELECT ifadeleri için
 - executeUpdate() => INSERT, UPDATE, DELETE ifadeleri

SELECT ifadesinin sonuç kümesi (satırlar/sütunlar) bir **ResultSet** nesnesi olarak döner.

ÖRNEK:

- ResultSet rs = st.executeQuery("SELECT * FROM personel");



4. SONUÇLARI İŞLE

- ResultSet olarak alınan SQL sonuçlarının her bir satırına erişmek için bir döngü kullanılabilir.
 - Döngü içerisinde dolaşmak için rs.next() metodu kullanılmalıdır :

```
ResultSet rs = st.executeQuery("Select personel_isim FROM personel");
  while(rs.next()) {
    System.out.println("Personel Adi:" + rs.getString("personel_isim"));
    }
}
```

→ Bir sütundaki değerleri almak için :

```
String personelld = rs.getString("personel_id");
Double maas = rs.getDouble("maas");
```



5. BAĞLANTIYI BİTİR

- → ResultSet nesnesi kapat.
 - rs.close();
- → Statement nesnesini kapat.
 - st.close();
- → Bağlantıyı kapat.
 - con.close();

ÖRNEK



```
import java.sql.*;
                       // JDBC metotlari icin kutuphane
public class JdbcQuery01 {
public static void main(String[] args) throws ClassNotFoundException, SQLException {
  //1) Veritabani icin uygun Driver'ı ekle
  Class.forName("oracle.jdbc.driver.OracleDriver");
  //2) Veritabani baglantisi olustur
  Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521/ORCLCDB.localdomain", "meryem", "1234");
  //3) SQL komutlari icin bir Statement nesnesi olustur.
  Statement st = con.createStatement();
  //4)Sorgu ifadesini calistir. (Personel tablosundaki id=101 olan personelin ismini sorgula)
  ResultSet rs1 = st.executeQuery("select isim from personel where id=101");
  //5)Sonuclari isle
  while(rs1.next()) {
     System.out.println("Personel Adi: " + rs1.getString("isim"));
  //6) Olusturulan nesneleri bellekten kaldir.
  st.close():
  con.close();
  rs1.close();
```