

# JDBC Resultset Tipleri



# Scrollable ResultSet

- ResultSet üzerinde **ileri**, **geri** veya **absolute** pozisyon belirterek hareket edilebilir
- JDBC sürücüsünün **desteklemesi** gerekir
- **Statement oluşturulurken** scrollable ResultSet istendiği belirtilmelidir

```
Statement stmt = connection.createStatement(
    ResultSet.TYPE_SCROLL_INSENSITIVE,
    ResultSet.CONCUR_READ_ONLY);
```

```
ResultSet rs = stmt.executeQuery("select * from T_PET");
```

TYPE\_FORWARD\_ONLY (default)  
 TYPE\_SCROLL\_INSENSITIVE  
 TYPE\_SCROLL\_SENSITIVE (DB'deki değişiklikleri anında yansıtır)

# Scrollable ResultSet

```
rs.next();
rs.previous();
rs.first();
rs.absolute(1);
rs.absolute(10);
rs.relative(1);
rs.relative(-1);
rs.beforeFirst();
rs.last();
rs.absolute(-1);
rs.absolute(-2);
rs.afterLast();
rs.isBeforeFirst();
rs.isAfterLast();
rs.isFirst();
rs.isLast();
```

Hepsi boolean sonuç döner

# Updatable ResultSet

- Mevcut **ResultSet** üzerinden ilgili sütunlar doğrudan **güncellenebilir**, yeni kayıt **eklenebilir** veya mevcut bir kayıt **silinebilir**
- JDBC sürücüsünün **desteklemesi** gerekir
- **Statement** oluşturulurken **updatable ResultSet** istendiği belirtilmelidir

```
Statement stmt = connection.createStatement(
    ResultSet.TYPE_SCROLL_INSENSITIVE,
    ResultSet.CONCUR_UPDATABLE);
```

```
ResultSet rs = stmt.executeQuery("select * from T_PET");
```

CONCUR\_READ\_ONLY  
CONCUR\_UPDATABLE

# Updatable ResultSet

```
rs.first();
rs.updateString("NAME", "111");
rs.updateRow();

rs.last();
rs.updateDate("BIRTH_DATE", new java.sql.Date(123));
rs.updateRow();

rs.moveToInsertRow();
rs.updateLong("ID", 101);
rs.updateString("NAME", "My Pet");
rs.updateInt("VERSION", 0);
rs.updateDate("BIRTH_DATE", new java.sql.Date(123));
rs.insertRow();
rs.moveToCurrentRow();

rs.absolute(10);
rs.deleteRow();
```

# Batch Insert & Update Kabiliyeti

- Tek bir network iletişiminde **birden fazla** kayıt üzerinde **ekleme**, **silme** veya **güncelleme** yapmaya imkan tanır
- JDBC sürücüsünün **desteklemesi** gerekir
- **PreparedStatement** üzerinden gerçekleştirilir

# Batch Insert & Update Kabiliyeti

```
connection.setAutoCommit(false);
```

```
PreparedStatement pstmt = connection.prepareStatement(  
    "insert into T_PET(ID,VERSION,NAME,BIRTH_DATE) values (?, ?, ?, ?)");
```

```
Object[][] data = new Object[][]{  
    {110,0,"Bobo",new java.sql.Date(123)},  
    {111,0,"Coco",new java.sql.Date(456)},  
    {112,0,"Dodo",new java.sql.Date(789)}  
};
```

```
for (Object[] row : data) {  
    pstmt.setObject(1, row[0]);  
    pstmt.setObject(2, row[1]);  
    pstmt.setObject(3, row[2]);  
    pstmt.setObject(4, row[3]);
```

```
    pstmt.addBatch();  
}
```

```
int[] updateCounts = pstmt.executeBatch();
```

```
connection.commit();
```

Batch işlem sonucu üzerinde  
işlem yapılan satır sayısı döner

# İletişim



[www.harezmi.com.tr](http://www.harezmi.com.tr)

[www.java-egitimleri.com](http://www.java-egitimleri.com)



[info@harezmi.com.tr](mailto:info@harezmi.com.tr)

[info@java-egitimleri.com](mailto:info@java-egitimleri.com)



[@HarezmiBilisim](https://twitter.com/HarezmiBilisim)

[@JavaEgitimleri](https://twitter.com/JavaEgitimleri)