

Ejecución de sentencias de descripción de datos (I)

- ▶ Cuando desarrollamos una aplicación JDBC conocemos la estructura de las tablas, los datos que estamos manejando y las relaciones que hay.
- ▶ Si desconocemos la estructura de la base de datos, puede obtenerse a través de los **metaobjetos**.
- ▶ El objeto **DatabaseMetaData** proporciona información sobre la base de datos a partir de los métodos que contiene.
- ▶ Podemos conectar con la BD MySQL de nombre ejemplo y mostrar información sobre el producto, el driver, la URL, el nombre de usuario y las tablas y vistas del esquema actual.

Ejecución de sentencias de descripción de datos (II)

```
public class DatosBase {  
    public static void main(String[] args) {  
        try {  
            Connection conexion =  
DriverManager.getConnection("jdbc:mysql://localhost/ejemplo", "ejemplo", "ejemplo");  
  
            DatabaseMetaData dbmd=conexion.getMetaData();  
  
            ResultSet result = null;  
            String nombre= dbmd.getDatabaseProductName();  
            String driver= dbmd.getDriverName();  
            String url= dbmd.getURL();  
            String usuario= dbmd.getUserName();  
  
            System.out.println("INFORMACIÓN BASE DATOS");  
            System.out.println("Nombre: "+nombre);  
            System.out.println("Driver: "+driver);  
            System.out.println("URL: "+url);  
            System.out.println("Usuario: "+usuario);  
        }  
    }  
}
```

Ejecución de sentencias de descripción de datos (III)

```
// información sobre tablas y vistas
result= dbmd.getTables(null, "ejemplo", null, null);
while (result.next()){
    String catalogo= result.getString(1);
    String esquema= result.getString(2);
    String tabla= result.getString(3);
    String tipo= result.getString(4);
    System.out.println(tipo + " - Catálogo: "+ catalogo+
        ", Esquema: "+esquema+", Nombre: "+tabla);
}
conexion.close();
}
catch ( ClassNotFoundException | SQLException cn ) {cn.printStackTrace();}
}
```

Ejecución de sentencias de descripción de datos (IV)

- ▶ El método `getTables()` devuelve información sobre las tablas y vistas. Sus parámetros son:
 - ▶ Primero: catálogo de la base de datos. null indica todos los catálogos.
 - ▶ Segundo: esquema de la base de datos. Obtiene las tablas del esquema indicado. null indica el esquema actual (en algunos SGBD significa todos, p.e. en Oracle)
 - ▶ Tercero: patrón para seleccionar las tablas. Se puede utilizar `_` o `%`. Ej. `"de%"`, correspondería a todas las tablas que empiecen por de.
 - ▶ Cuarto: es un array de String en el que se indica qué tipos de tabla se desean:
 - ▶ `TABLE`, para tablas
 - ▶ `VIEW`, para vistas
 - ▶ `null`, todos los tipos

Ejecución de sentencias de descripción de datos (V)

- ▶ Con `result.getString()`, podríamos utilizar las siguientes constantes:
 - ▶ `TABLE_CAT`, nombre del catálogo al que pertenece.
 - ▶ `TABLE_SCHEM`, nombre del esquema.
 - ▶ `TABLE_NAME`, nombre de la tabla o vista.
 - ▶ `TABLE_TYPE`, el tipo TABLE o VIEW.
 - ▶ `REMARKS`, comentarios.

```
String catalogo= result.getString("TABLE_CAT");
```

```
String esquema= result.getString("TABLE_SCHEM");
```

```
String tabla= result.getString("TABLE_NAME");
```

```
String tipo= result.getString("TABLE_TYPE");
```

Ejecución de sentencias de descripción de datos (VI)

- ▶ Otros métodos importantes del objeto DatabaseMetaData son:
 - ▶ `getColumns(catálogo, esquema, nombre_tabla, nombre_columna)` devuelve información sobre las columnas filtradas. Se puede utilizar `_` y `%`. Si se pasa null en los cuatro parámetros se devuelve información sobre todas las columnas y tablas del esquema actual.

```
ResultSet columnas=null;
columnas= dbmd.getColumns(null, null, null, null);
while (columnas.next()){
    String name= columnas.getString("TABLE_NAME");//getString(3);
    String nombreCol= columnas.getString("COLUMN_NAME");//getString(4);
    String tipoCol= columnas.getString("TYPE_NAME");//getString(6);
    String tamCol= columnas.getString("COLUMN_SIZE");//getString(7);
    String nula= columnas.getString("IS_NULLABLE");//getString(18);
    System.out.println("Tabla: "+name+" Columna: "+nombreCol+", Tipo: "+
tipoCol +
", Tamaño: "+tamCol+", ¿Es nula?: "+ nula);
}
▶ 52
```

Ejecución de sentencias de descripción de datos (VII)

- ▶ Otros métodos importantes del objeto DatabaseMetaData son:
 - ▶ `getPrimaryKeys(catálogo, esquema, nombre_tabla)` devuelve la lista de columnas que forman la clave primaria.
 - ▶ `getExportedKeys(catálogo, esquema, nombre_tabla)` devuelve la lista de las claves ajenas que utilizan la clave primaria de esta tabla.

```
ResultSet pk= dbmd.getPrimaryKeys(null, "ejemplo", "departamentos");
```

```
StringBuilder claves= new StringBuilder();
```

```
String sep="";
```

```
while (pk.next()){
```

```
    claves.append(sep);
```

```
    claves.append(pk.getString(4));
```

```
    sep=" + ";
```

```
}
```

```
System.out.println("Clave primaria: "+ claves.toString());
```

```
ResultSet fk= dbmd.getExportedKeys(null, "ejemplo", "departamentos");
```

```
while (fk.next()){
```

```
    String fk_name=fk.getString("FKCOLUMN_NAME");
```

```
    String fk_tablename= fk.getString("FKTABLE_NAME");
```

```
    System.out.println("Tabla FK: "+fk_tablename+" Clave ajena: "+  
        fk_name);
```

Ejecución de sentencias de descripción de datos (VIII)

► **ResultSetMetaData**

- Se trata de una interfaz que permite obtener metadatos a partir de un *ResultSet*. Significa que podemos obtener información sobre los tipos y propiedades de las columnas, como el número de columnas devuelto. Su principales métodos son:
 - `getColumnCount()`. Devuelve el número de columnas devueltas por una consulta.
 - `columnName(indice de la columna)`. Devuelve el nombre de la columna.
 - `getColumnTypeName(indice)`. Devuelve el nombre del tipo de dato que contiene la columna específico del sistema de bases de datos.
 - `isNullable(indice)`. Devuelve 0 si la columna puede contener valores nulos.
 - `getColumnDisplaySize(indice)`. Devuelve el máximo ancho en caracteres de la columna.



Ejecución de sentencias de descripción de datos (IX)

```
Statement sentencia= conexion.createStatement();
ResultSet rs= sentencia.executeQuery("Select * from departamentos");
ResultSetMetaData rsmd= rs.getMetaData();
int nCol= rsmd.getColumnCount();
System.out.printf("%-20s%-15s%-8s%-5s\n", "NOMBRE", "TIPO", "NULA",
"ANCHO");

    for (int i=1; i<= nCol; i++){
        System.out.printf("%-20s%-15s%-8s%-5s\n",
            rsmd.getColumnName(i),
            rsmd.getColumnTypeName(i),
            rsmd.isNullable(i),
            rsmd.getColumnDisplaySize(i));
    }
```