



Тема №7

Робота з базами даних



Поняття про JDBC

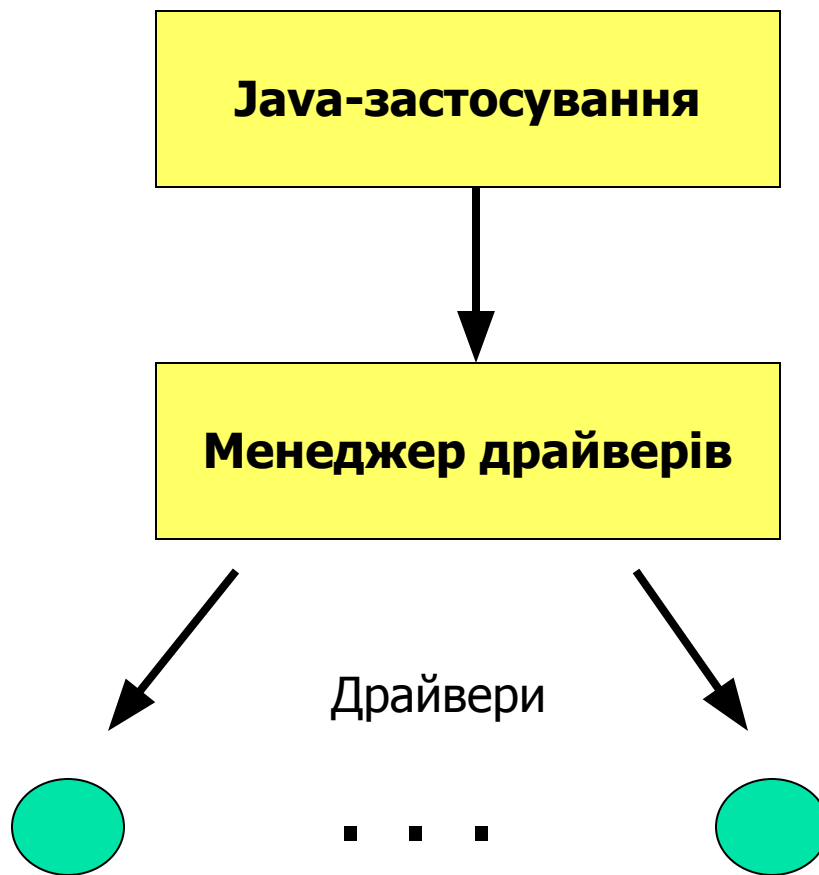
- Можливості, пов'язані з доступом з Java-програм до баз даних.
- Такі засоби утворюють інтерфейс JDBC (Java DataBase Connectivity).
- Базовий пакет - **java.sql**.
- Додаткові можливості - **javax.sql**.



JDBC: продовження

- За задумом - універсальні засоби, придатні для роботи з будь-якою БД.
- Для взаємодії з конкретною БД - ***драйвер БД***, який розробляється постачальником.

Базова схема





Клас DriverManager

- Забезпечує реєстрацію драйверів БД, з'єднання з конкретною БД та передачу драйверові SQL-запити.
- Використання **DriverManager** не дуже рекомендується для розвинених корпоративних застосувань.
Рекомендується - на основі реалізації інтерфейса **DataSource**.



Про драйвери БД

- Є різні типи драйверів.
- Один з них (мабуть, історично перший) - т.зв. (міст) **JDBC-ODBC**.
- Найбільш перспективними вважаються драйвери, написані повністю на Java і здатні безпосередньо контактувати з сервером БД.

Приклад роботи з базою даних



```
public static void main(String args []) throws Exception {  
String dbName="jdbc:odbc:test";  
System.out.println("Hello, folks");  
String query="SELECT * from student";  
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");  
Connection conn = DriverManager.getConnection(dbName);  
Statement statement = conn.createStatement();  
ResultSet rs=statement.executeQuery(query);  
while (rs.next()) {  
System.out.println(rs.getString("name")+  
    "+rs.getString("Rating"));}  
rs.close();  
}
```



Більш рекомендована схема

```
try { отримати з'єднання  
    try { операції}  
    finally {conn.close();}  
}  
catch (SQLException se) {  
    повідомлення про помилку  
}
```




Оновлення БД

- Якщо дається запит на оновлення бази даних, замість методу `executeQuery` слід використовувати метод **`executeUpdate`** (або **`execute`**).



Підготовлені запити (Prepared Statements)

- підвищення ефективності;
- можна задавати параметри запиту.

Prepared Statements:

приклад

```
BufferedReader br=new BufferedReader(new InputStreamReader
(System.in));
System.out.println("Enter city");
String town=br.readLine();
String query="SELECT * from city where town=?";
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection conn = DriverManager.getConnection(dbName);
PreparedStatement statement =
conn.prepareStatement(query);
statement.setString(1,town);
ResultSet rs=statement.executeQuery();
```



Деякі інші можливості

- Робота з *метаданими*.
- *Транзакції*.