

JDBC (Java Database Connectivity)

JDBC je Java API za konektovanje i izvršavanje upita nad bazom. Za konektovanje na bazu koristi se JDBC drajver.

Koraci pri radu sa bazom podataka su sledeći:

1. Registrovanje drajvera (urađeno u DB.java)

```
Class.forName("com.mysql.jdbc.Driver");
```

2. Kreiranje konekcije (urađeno u DB.java, mi samo treba da uzmemo jednu od konekcija iz niza svih konekcija kada želimo da radimo sa bazom)

```
DriverManager.getConnection("jdbc:mysql://localhost:3306/naziv_baze","username","password");
```

```
DB.getInstance().getConnection();
```

3. Kreiranje Statement-a iz konekcije. Statement objekat služi za izvršavanje upita nad bazom. *PreparedStatement

```
Statement statement = connection.createStatement();
```

4. Izvršavanje upita

Metode od interesa prilikom rada sa Statement objektom su sledeće:

public ResultSet executeQuery(String sql): za izvršavanje SELECT upita.

Povratna vrednost je objekat ResultSet-a (vraćaju se svi redovi koji odgovaraju select-u).

Metode od interesa prilikom rada sa ResultSet-om su sledeće:

public boolean next(): pomeranje kursora na sledeći red u odnosu na tekući

public boolean previous(): pomeranje kursora na prethodni red u odnosu na tekući

public boolean first(): pomeranje kursora na prvi red

public boolean last(): pomeranje kursora na poslednji red

public boolean absolute(int row): pomeranje kursora na određeni red

public boolean relative(int row): pomeranje kursora relativno za određeni broj mesta napred ili nazad u odnosu na tekući red

public int getInt(int columnIndex): dohvatanje kolone iz reda kao int. Koloni se pristupa preko rednog broja

public int getInt(String columnName): dohvatanje kolone iz reda kao int. Koloni se pristupa preko naziva kolone

public String getString(int columnIndex): dohvatanje kolone iz reda kao String. Koloni se pristupa preko rednog broja

public String getString(String columnName): dohvatanje kolone iz reda kao String. Koloni se pristupa preko naziva kolone

public int executeUpdate(String sql): za izvršavanje UPDATE, INSERT i DELETE upita. Povratna vrednost je broj redova na koje upit utiče.

5. Zatvaranje otvorenih resursa

```
resultSet.close();
```

```
statement.close();
```

Konekciju ne zatvaramo, već je vraćamo u niz svih konekcija da bi mogla da se ponovo koristi.

```
DB.getInstance.putConnection(connection);
```

U slučaju da svaki put otvaramo novu konekciju možemo i da je zatvorimo.