

JDBC

AUDITORNE VJEŽBE

Sadržaj

Instalacija DTP *plugina*

Preuzimanje i instaliranje H2 baze podataka

Pokretanje administracijske konzole H2 baze podataka

Kreiranje baze podataka korištenjem Eclipse „Database Development” perspektive

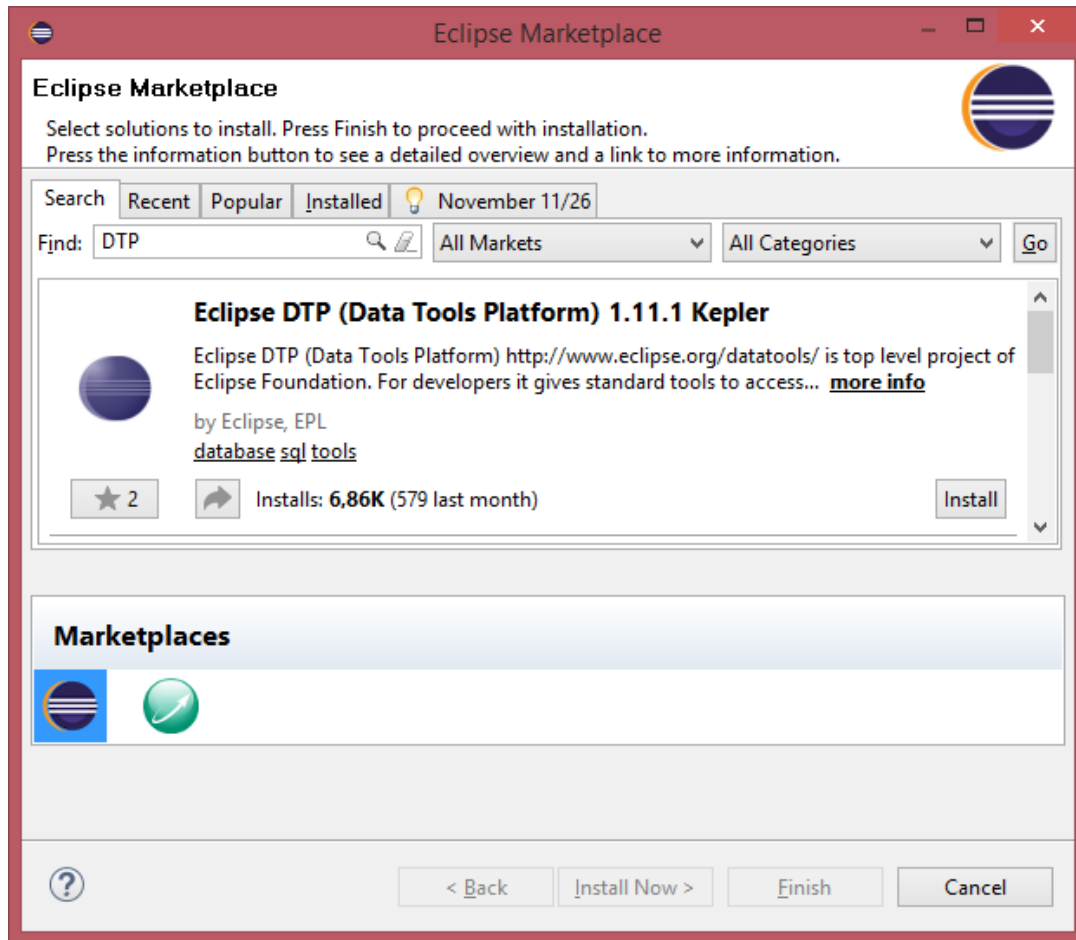
Izvršavanje upita kroz H2 administracijsku konzolu

Izvršavanje upita u Eclipse okruženju

Properties datoteke

H2 Maven konfiguracija

Instaliranje DTP *plugina*



- Moguće je koristiti Eclipse Marketplace
- Verzija 1.11.1 je kompatibilna i s Eclipse Luna inačicom
- Nakon instalacije omogućena je perspektiva „Database Development”

Preuzimanje i instaliranje H2 baze podataka

- Da bi se iz Eclipsea mogla ostvariti veza s H2 bazom podataka, prvo je potrebno preuzeti i pokrenuti instancu H2 baze
- Preporuča se preuzimanje H2 „Platform-Independent Zip” inačice baze:



[Translate](#)

Search:

[Home](#)

Downloads

Version 1.4.182 (2014-10-17), Beta

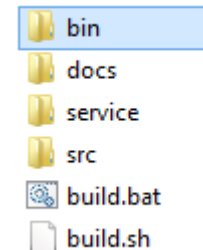
[Windows Installer](#) (SHA1 checksum: 94e393db37282af3fbc068b724ac62421a9b6aa5)

[Platform-Independent Zip](#) (SHA1 checksum: 2d9fc2147341e1942e06779bfeba75f4eca29c9c)

Version 1.3.176 (2014-04-05), Last Stable

[Windows Installer](#)

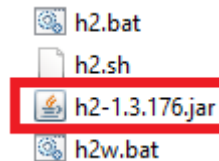
[Platform-Independent Zip](#)



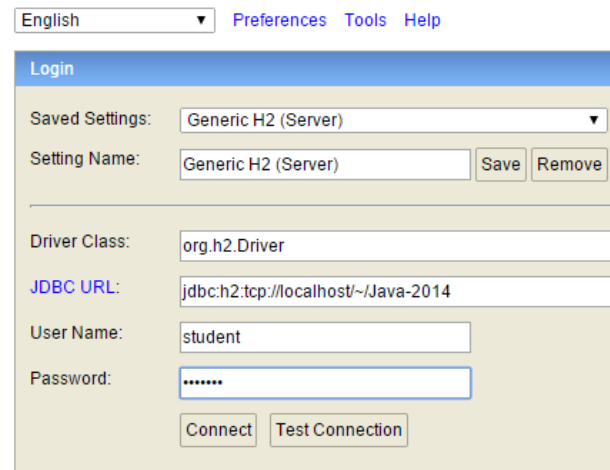
- Raspakiravanjem Zip arhive (7,4 MB) završava proces instalacije H2 baze

Pokretanje administracijske konzole H2 baze podataka

- Administracijska konzola H2 baze podataka pokreće se korištenjem npr. JAR datoteke unutar „bin” mape unutar instalacijske arhive:

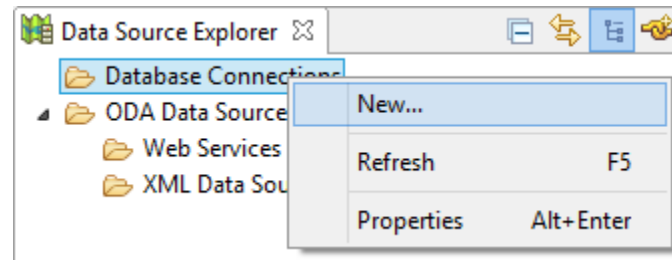


- Nakon toga se u Internet pregledniku prikazuje grafičko sučelje administracijske konzole:

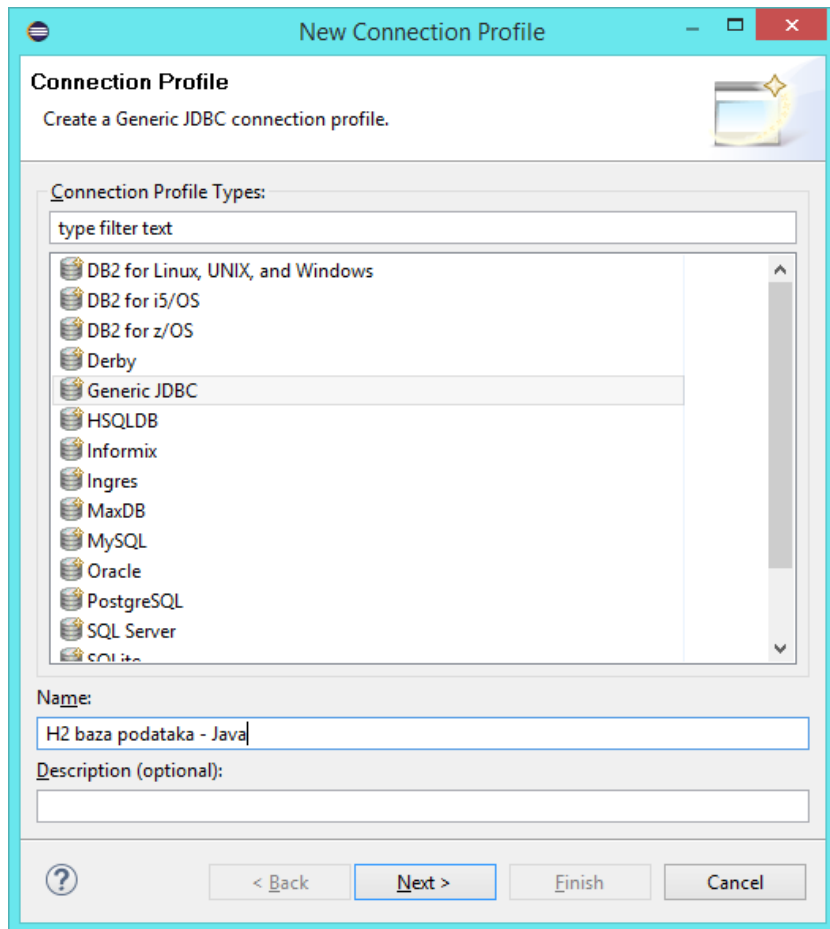
A screenshot of the H2 database administrative console's login page. At the top, there is a language dropdown menu set to 'English' and links for 'Preferences', 'Tools', and 'Help'. The main section is titled 'Login' and contains several input fields: 'Saved Settings' (a dropdown menu showing 'Generic H2 (Server)'), 'Setting Name' (a text input field with 'Generic H2 (Server)' and 'Save'/'Remove' buttons), 'Driver Class' (a text input field with 'org.h2.Driver'), 'JDBC URL' (a text input field with 'jdbc:h2:tcp://localhost/~Java-2014'), 'User Name' (a text input field with 'student'), and 'Password' (a masked text input field with '*****'). At the bottom, there are 'Connect' and 'Test Connection' buttons.

Kreiranje baze podataka korištenjem Eclipse „Database development” perspektive

- Nakon inicijalnog pokretanja H2 baze podataka što se manifestira prikazivanjem administracijske konzole nije još moguće spojiti se na bazu podataka, jer ona u tom trenutku još ne postoji
- Instanca baze podataka mora se kreirati korištenjem Eclipse „Database Development” perspektive
- Prije toga potrebno je definirati driver za H2 bazu podataka korištenjem opcije „New” unutar „Database Development” perspektive unutar „Data Source Explorer” *viewa*:



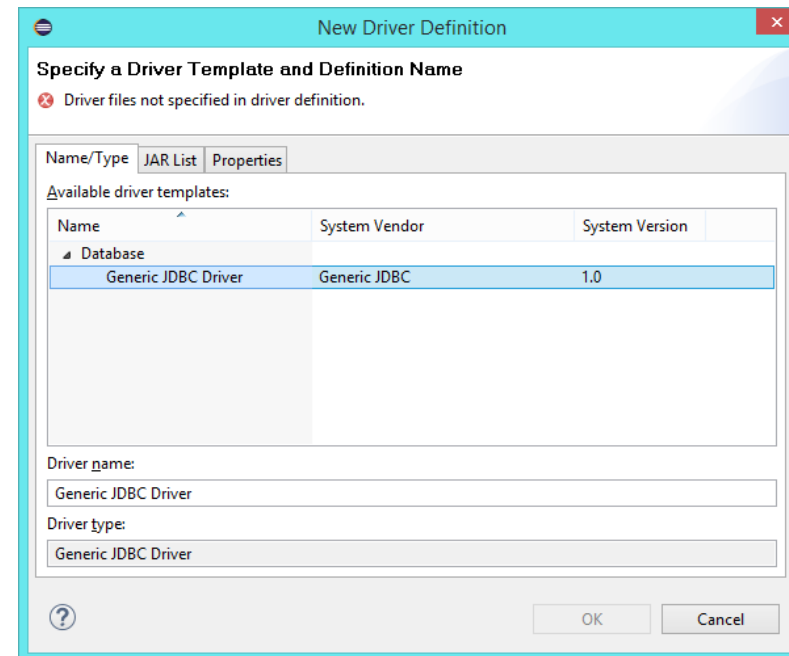
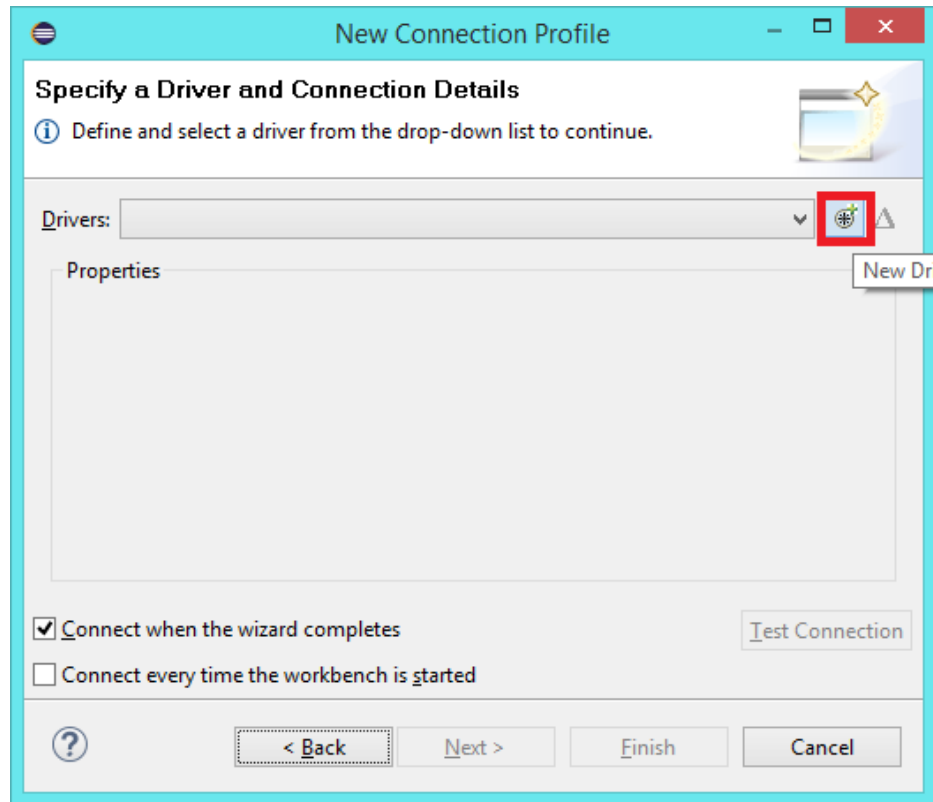
Kreiranje baze podataka korištenjem Eclipse „Database development” perspektive



- Na dijalogu koji se pojavljuje nakon odabira opcije „Add” potrebno je odabrati tip „Generic JDBC” i definirati prikladan naziv za bazu podataka

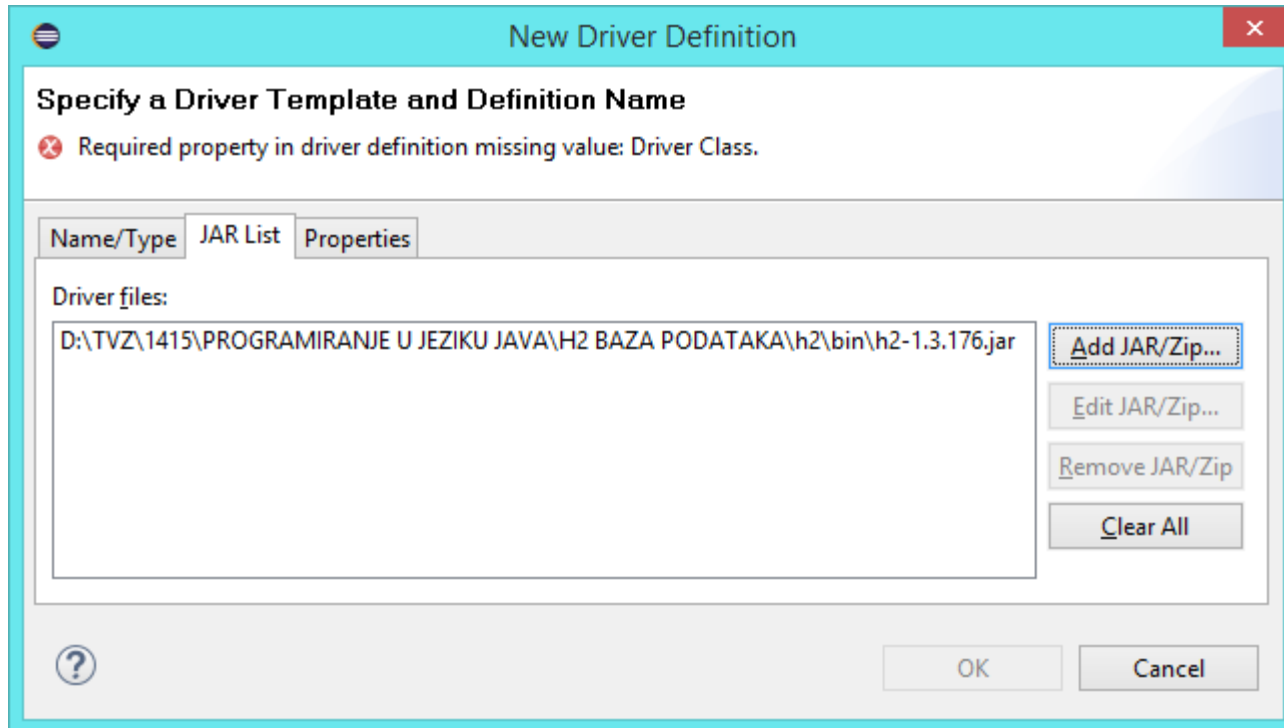
Kreiranje baze podataka korištenjem Eclipse „Database development” perspektive

- Na sljedećem ekranu potrebno je odabrati opciju za definiranje novog JDBC drivera i odabrati JAR koji definira vrstu baze podataka



Kreiranje baze podataka korištenjem Eclipse „Database development” perspektive

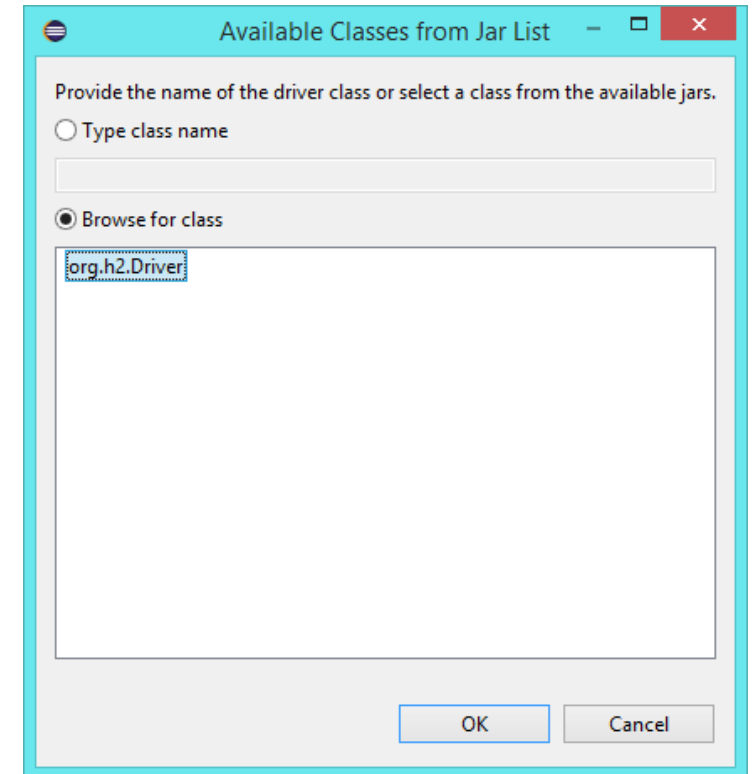
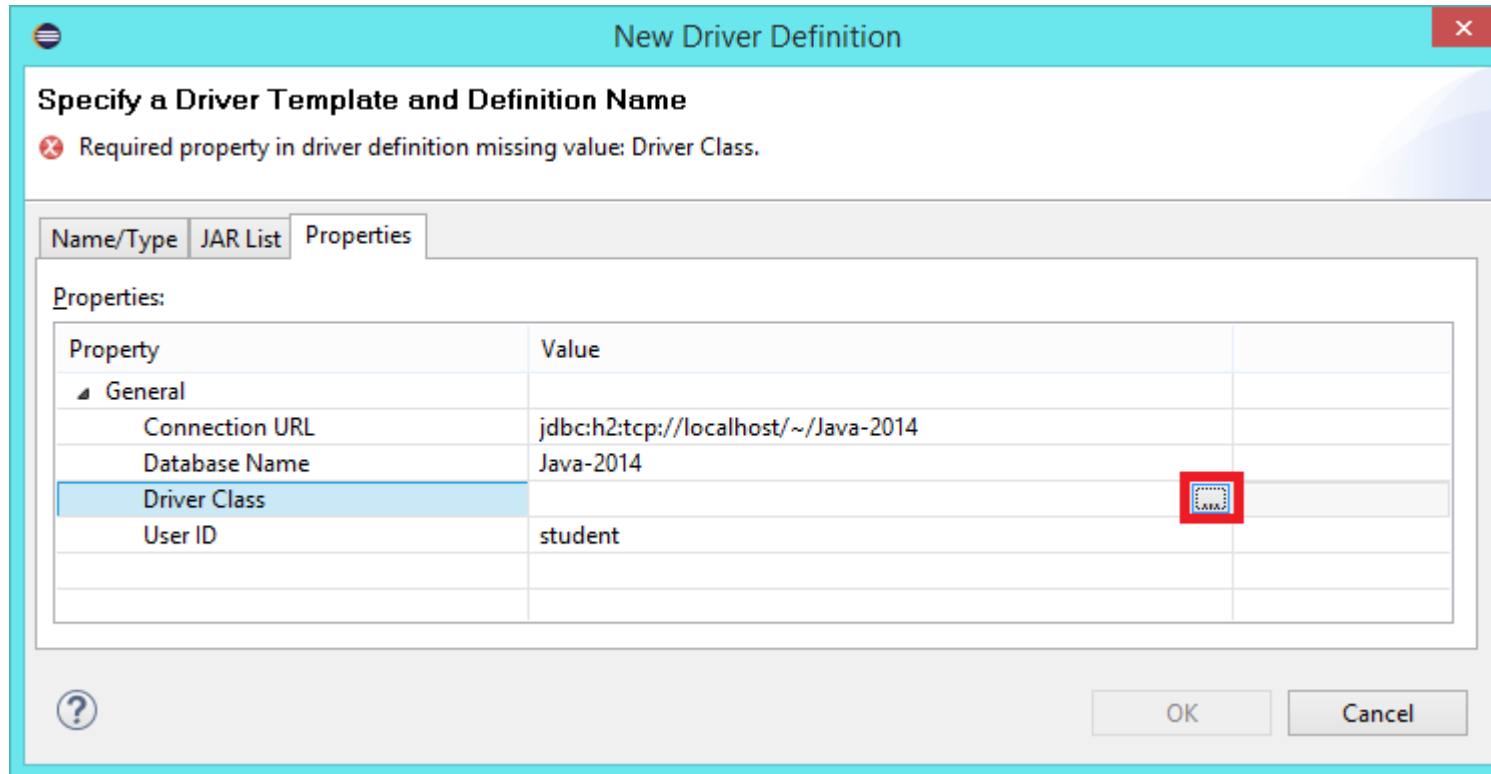
- Poruka o pogrešci „Driver files not specified in driver definition” zahtijeva da se na *tabu* „JAR List” definira lokacija datoteke koja sadrži *driver* za bazu podataka



- Nakon toga još preostaje definirati naziv klase koja predstavlja driver na tabu „Properties” kako bi se otklonila pogreška opisana porukom „Required property in driver definition missing value: Driver Class.”

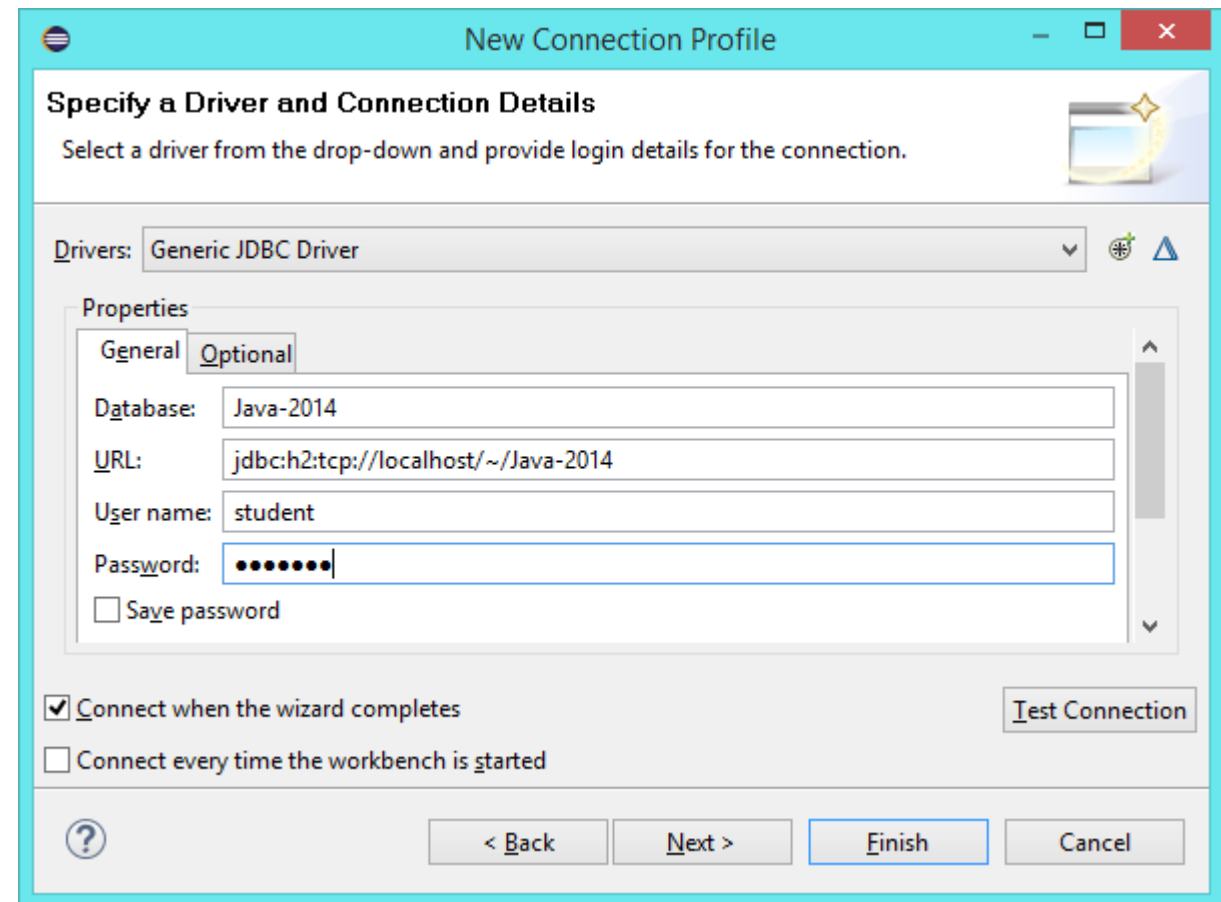
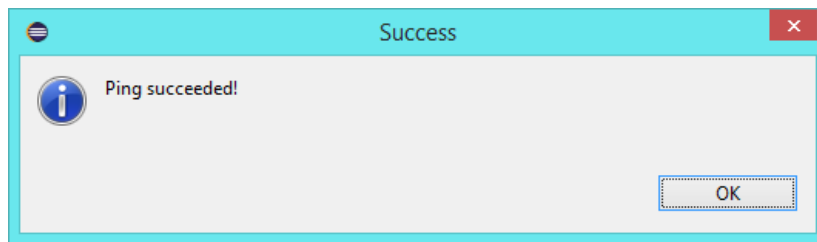
Kreiranje baze podataka korištenjem Eclipse „Database development” perspektive

- Na *tabu* „Properties” moguće je definirati i ostale parametre za H2 bazu podataka koji su inicijalno definirani na H2 administracijskoj konzoli:



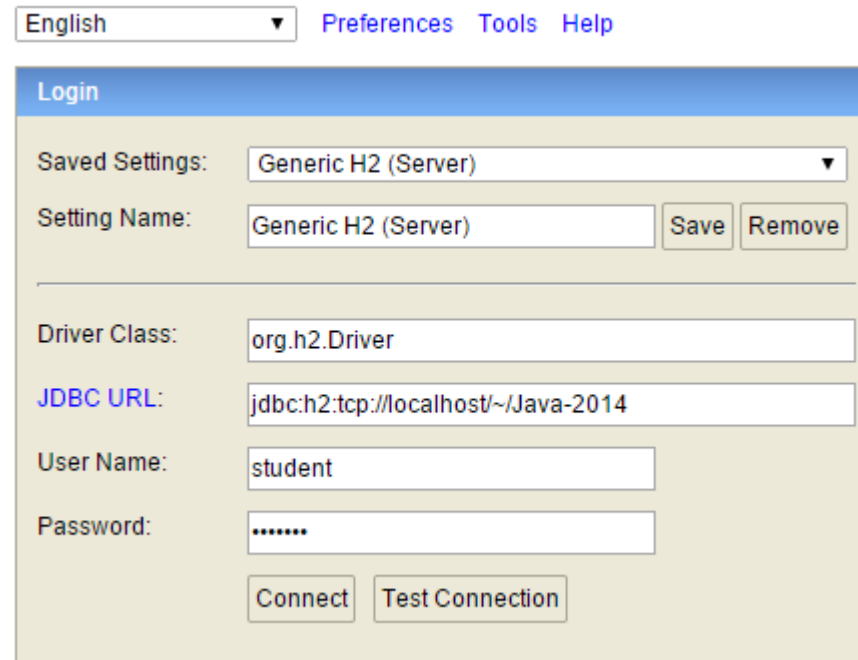
Kreiranje baze podataka korištenjem Eclipse „Database development” perspektive

- Na kraju još preostaje definiranje lozinke za bazu podataka i pritisnuti tipku „Test Connection”
- Ako je sve konfigurirano na ispravan način, prikazuje se poruka



Izvršavanje upita kroz H2 administracijsku konzolu

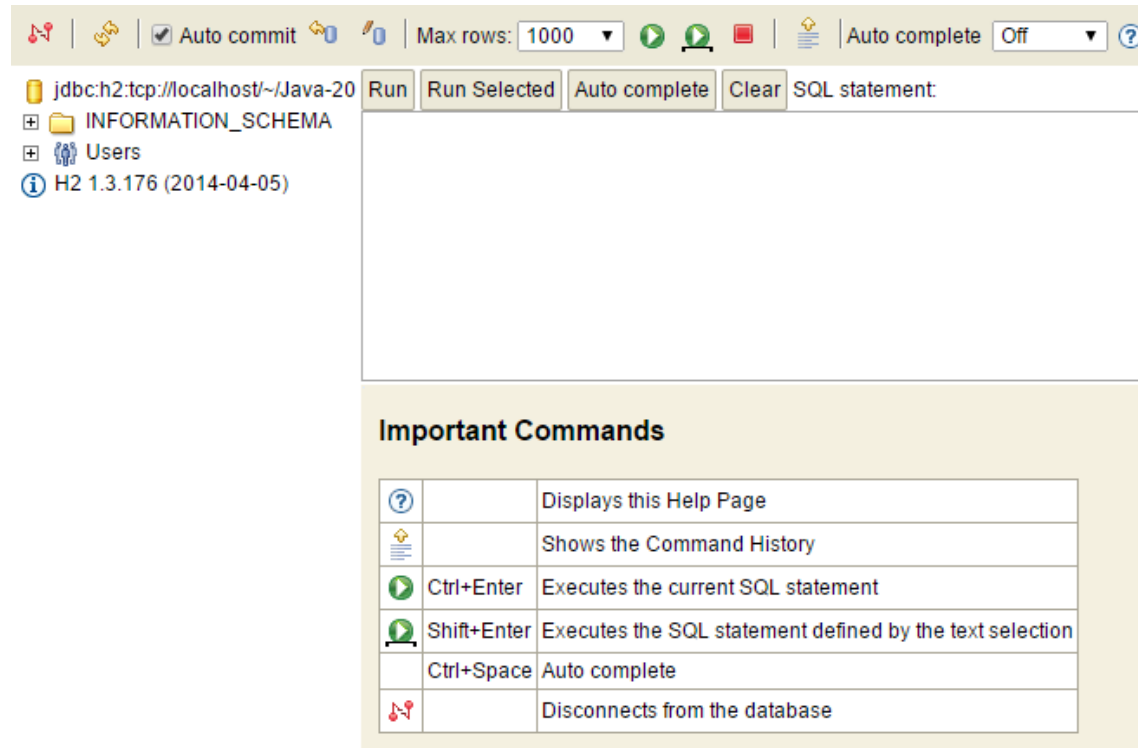
- Nakon kreiranja baze podataka kroz „Database Development” perspektivu, upisivanjem podataka s kojima je kreirana baza moguće je prijaviti se u administracijsku konzolu:



The screenshot shows the H2 Database Administration console interface. At the top, there is a language dropdown menu set to "English" and navigation links for "Preferences", "Tools", and "Help". Below this is a "Login" section with a blue header. It contains a "Saved Settings:" dropdown menu currently showing "Generic H2 (Server)". Below that is a "Setting Name:" field also containing "Generic H2 (Server)", with "Save" and "Remove" buttons to its right. A horizontal line separates this from the connection details. The "Driver Class:" field contains "org.h2.Driver". The "JDBC URL:" field contains "jdbc:h2:tcp://localhost/~Java-2014". The "User Name:" field contains "student". The "Password:" field is masked with "*****". At the bottom of the login section are "Connect" and "Test Connection" buttons.

Izvršavanje upita kroz H2 administracijsku konzolu

- Uspješnom prijavom prikazuje se grafičko sučelje koje omogućava uvid u strukturu baze podataka, kao i izvršavanje upita nad bazom:



Izvršavanje upita kroz H2 administracijsku konzolu

- Upisom ispravnih naredbi i pritiskom gumba „Run” moguće je izvršiti SQL naredbe nad bazom podataka i vidjeti rezultate:

RunRun SelectedAuto completeClear

SQL statement:

```
CREATE TABLE STUDENTI
(
  id INT NOT NULL GENERATED ALWAYS AS IDENTITY,
  jmbag VARCHAR(20) NOT NULL,
  ime VARCHAR(50) NOT NULL,
  prezime VARCHAR(50) NOT NULL,
  datum_rodjenja DATE NOT NULL,
  PRIMARY KEY (id)
);

INSERT INTO STUDENTI (jmbag, ime, prezime, datum_rodjenja) VALUES ('0036374849', 'Pero', 'Perić', '1994-01-01');
INSERT INTO STUDENTI (jmbag, ime, prezime, datum_rodjenja) VALUES ('0024568238', 'Ivo', 'Ivić', '1993-01-01');

SELECT * FROM STUDENTI;
```

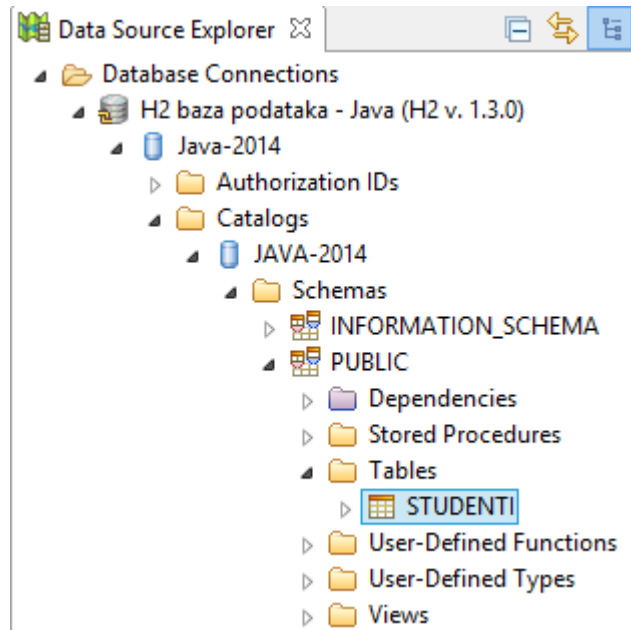
SELECT * FROM STUDENTI;

| ID | JMBAG | IME | PREZIME | DATUM_RODJENJA |
|----|------------|------|---------|----------------|
| 1 | 0036374849 | Pero | Perić | 1994-01-01 |
| 2 | 0024568238 | Ivo | Ivić | 1993-01-01 |

(2 rows, 3 ms)

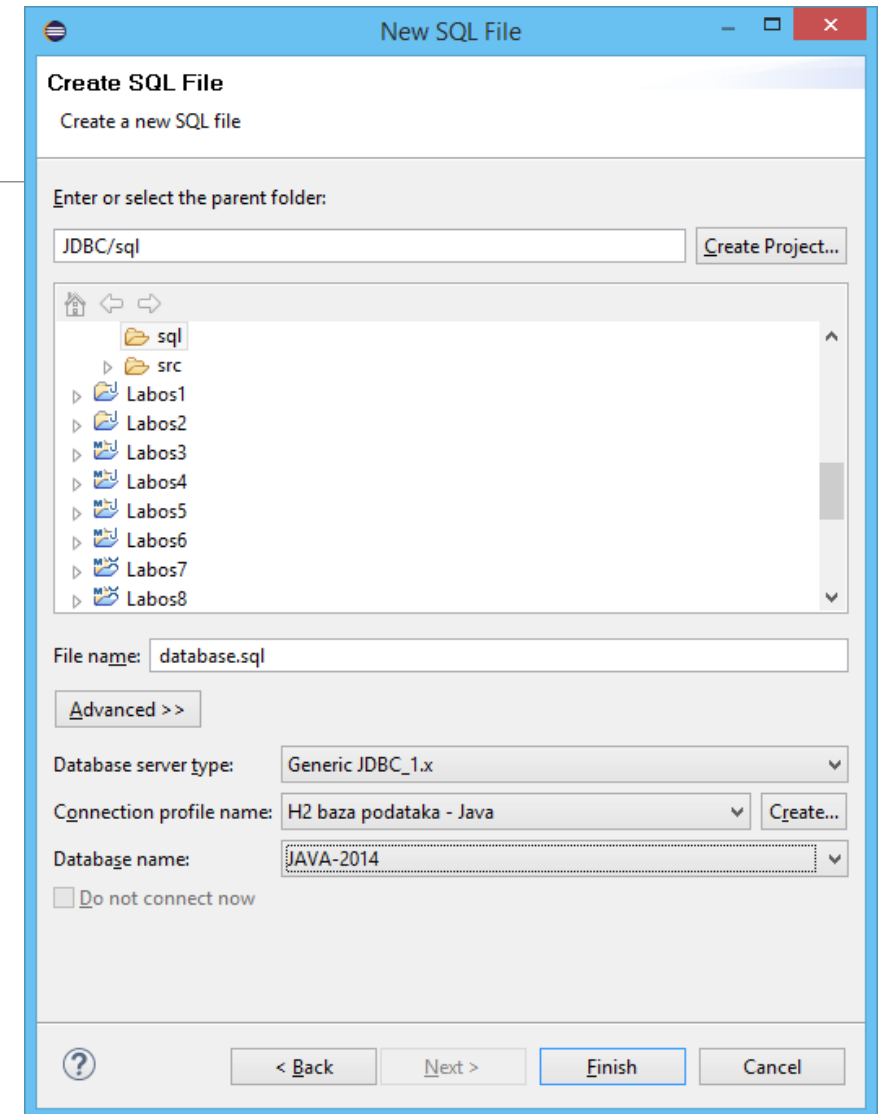
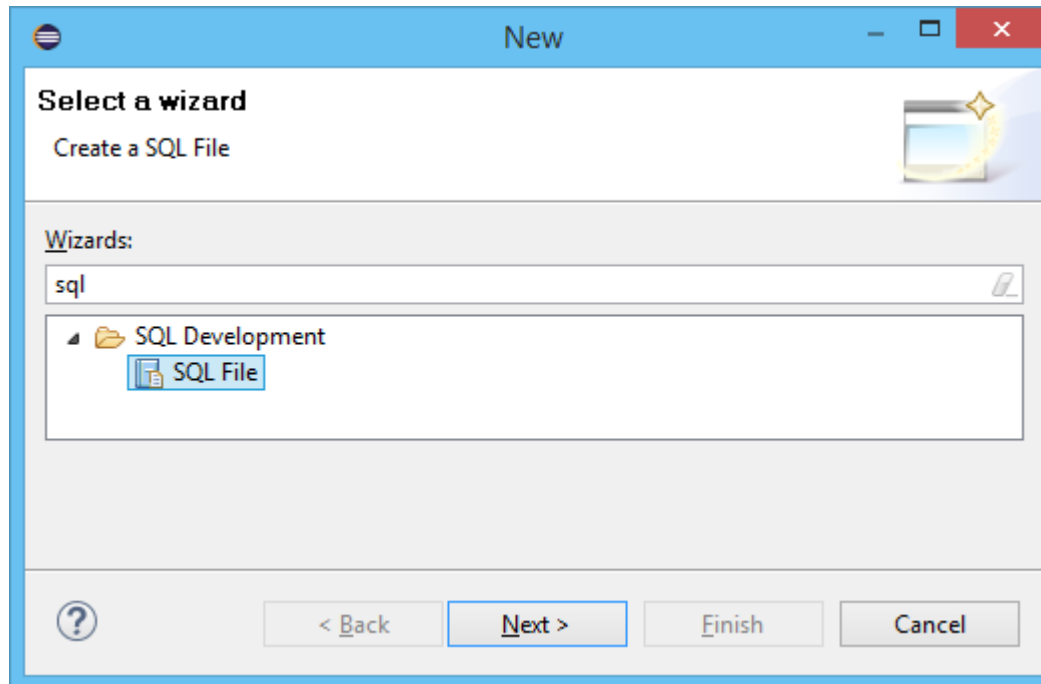
Izvršavanje upita u Eclipse okruženju

- Unutar Eclipsea je uz pomoć DTP *plugina* moguće također izvršavati upite nad bazom podataka
- Preduvjet je da je u „Database Development” perspektivi ostvarena veza s bazom podataka, npr.:



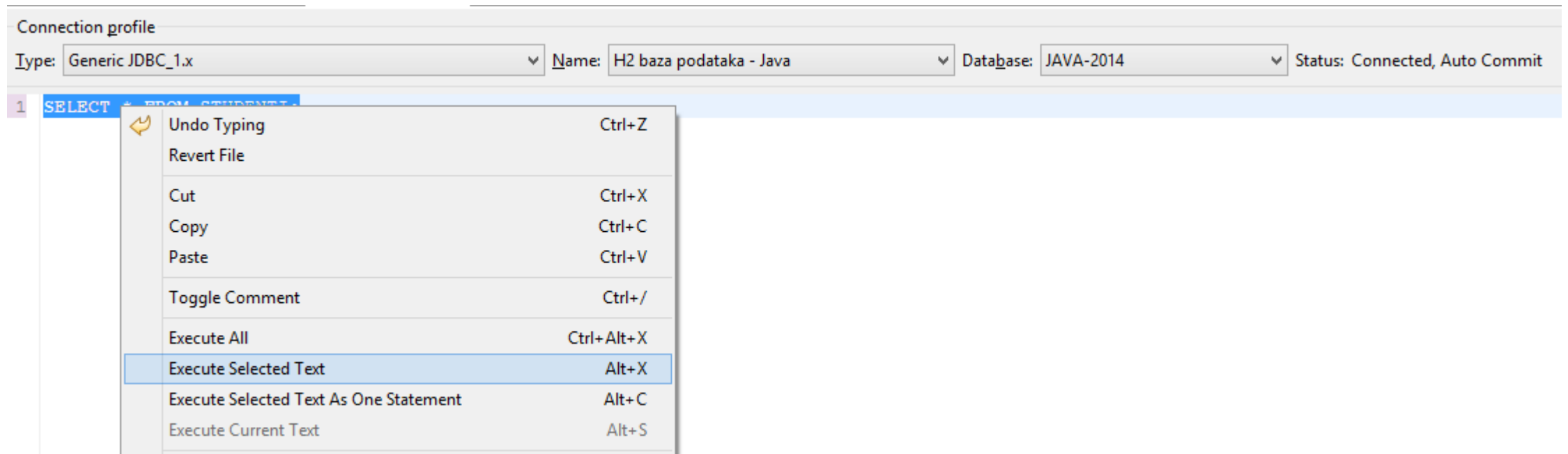
Izvršavanje upita u Eclipse okruženju

- Nakon toga je moguće kreirati SQL datoteku korištenjem opcije „File->New->Other->SQL File” koja izravno koristi tu vezu s bazom podataka:



Izvršavanje upita u Eclipse okruženju

- Nakon toga je upisivanjem SQL naredbi moguće odabrati jednu od opcija za izvršavanje upita:



Izvršavanje upita u Eclipse okruženju

- Rezultati izvršavanja upita su vidljivi unutar „Result1” *taba*:

| | | | | | | | | | |
|----------------------------|----------------|------------------|---------------|--------|---------|------------|--------|---------|----------------|
| Type query expression here | | | | Status | Result1 | | | | |
| Status | Operation | Date | Connectio... | | ID | JMBAG | IME | PREZIME | DATUM_RODJENJA |
| Started | SELECT * FR... | 4. pro 2014. ... | H2 baza po... | | 1 | 0036374849 | Pero | Perić | 1994-01-01 |
| Succes | SELECT * FR... | 4. pro 2014. ... | H2 baza po... | | 2 | 0024568238 | Željko | Ivić | 1993-01-01 |

Properties datoteke

- U Java programima je moguće koristiti tzv. „Properties” datoteke u kojima se nalaze ključni podaci koje nije najbolje držati unutar programskog koda (kao što su korisničko ime i lozinka za pristup bazi podataka)
- Na taj način se izbjegava „hardkodiranje” tih podataka u programskom kodu što zahtijeva ponovno prevođenje programa u slučaju njihove promjene
- Properties datoteke funkcioniraju po principu „ključ-vrijednost”, odnosno svaki jedinstveni ključ koji se koristi u Java aplikaciji ima svoju vrijednost koja se dohvaća prema tom ključu
- To znači da se programski kod ne mora mijenjati i ponovno prevoditi ako je potrebno promijeniti neku vrijednost, jer ključevi ostaju isti

Properties datoteke

- Na primjer, ako je potrebno definirati pristupne podatke za bazu, oni unutar Properties datoteke mogu izgledati ovako:

```
#Osnovni podaci za spajanje na bazu podataka  
bazaPodatakaUrl = jdbc:h2:tcp://localhost/~ /Java-2014
```

```
#Podaci za pristupanje bazi podataka  
korisnickoIme = student  
lozinka = student
```

Properties datoteke

- Primjer korištenja Properties datoteke:

```
private static final String DATABASE_FILE = "database.properties";

private static Connection connectToDatabase() throws SQLException, IOException {

    Properties svojstva = new Properties();

    svojstva.load(new FileReader(DATABASE_FILE));

    String urlBazePodataka = svojstva.getProperty("bazaPodatakaUrl");
    String korisnickoIme = svojstva.getProperty("korisnickoIme");
    String lozinka = svojstva.getProperty("lozinka");

    Connection veza = DriverManager.getConnection(urlBazePodataka,
                                                    korisnickoIme, lozinka);

    return veza;
}
```

H2 Maven konfiguracija

- Da bi se mogao koristiti H2 driver u Java aplikacijama, potrebno je iskoristiti sljedeću ovisnost unutar „pom.xml” datoteke:

```
<dependency>  
    <groupId>com.h2database</groupId>  
    <artifactId>h2</artifactId>  
    <version>1.3.176</version>  
</dependency>
```

Pitanja?
