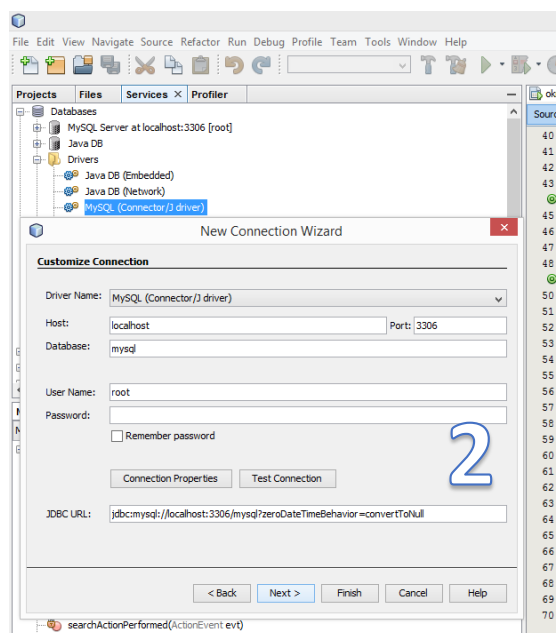
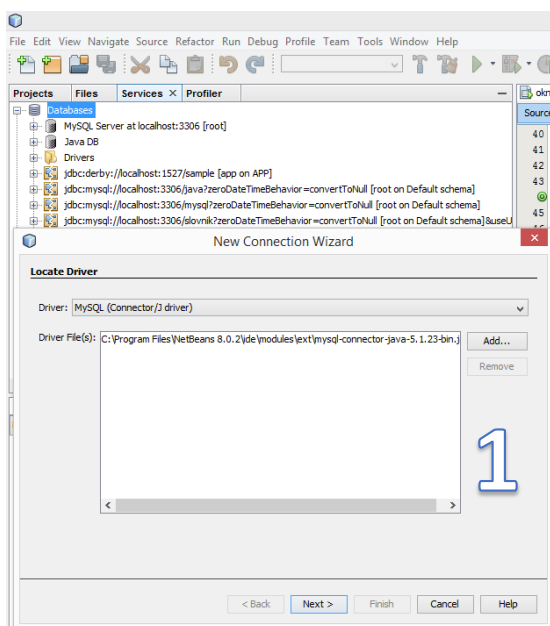


Jednoduchá databázová aplikace v Javě

Připojení MySQL prostřednictvím JDBC rozhraní



1. Přidání MySQL driveru pro JDBC

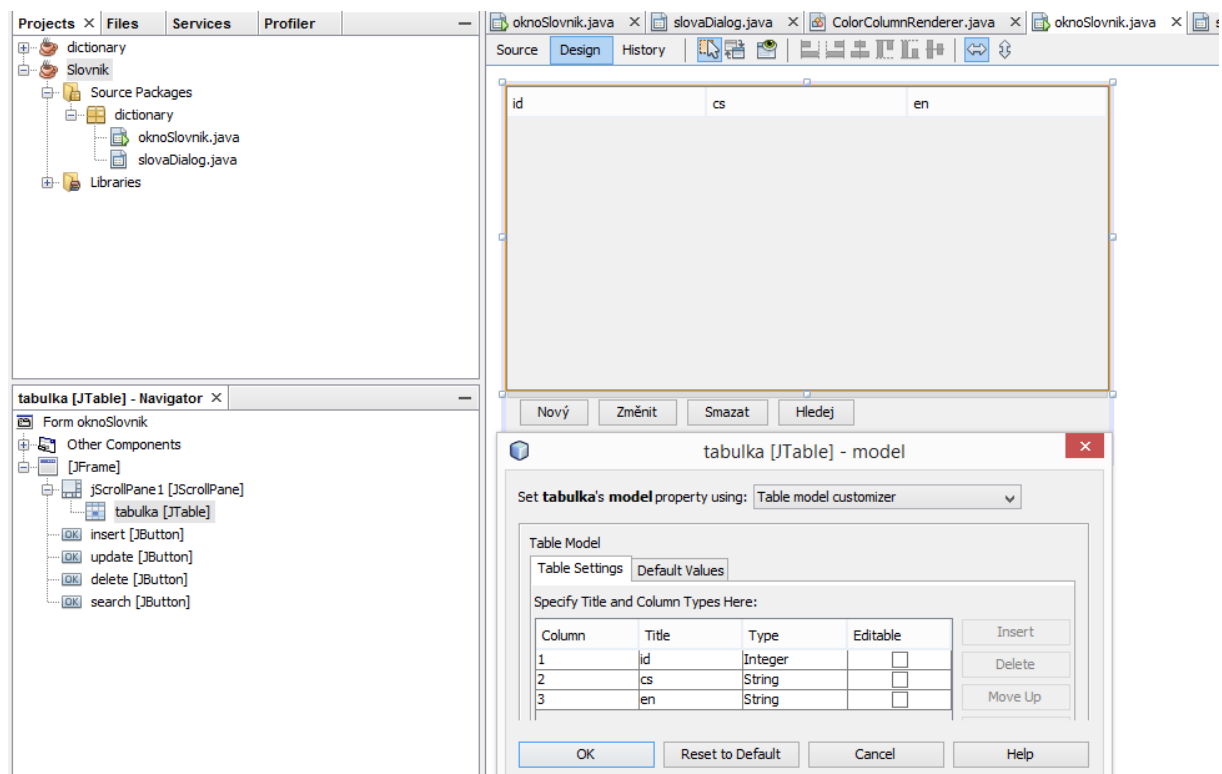
- Zvolit záložku Services a z místní nabídky nad položkou Databases otevřít dialogové okno pro výběr ovladače.
- Přidat driver pro databázový systém MySQL.

2. Nastavení spojení na databázový server MySQL

- Vyhledat mezi ovladači MySQL a zvolit z místní nabídky příkaz Connect Using...
- Vyplnit dialogové okno správnými údaji a otestovat připojení.

3. Vytvoření databáze na serveru MySQL a import dat ze souboru SQL

Vytvoření projektu okenní databázové aplikace



1. Vytvoření nového projektu

- V menu File | New Project... vytvořit pomocí průvodce nový projekt typu Java Application.
- Projekt bude označen Slovník, nebude požadováno vytvoření Main Class.
- V novém projektu bude vytvořen balíček dictionary.
- V tomto balíčku bude vytvořena nová třída JFrame Form nazvaná oknoSlovník.
- Po stisku pravého tlačítka myši nad názvem projektu (Slovník) a volbě příkazu Properties přidat v části Libraries knihovnu MySQL JDBC Driver.

2. Vytvoření uživatelského rozhraní aplikace

- Pro JFrame oknoSlovník nastavit prostřednictvím panelu Properties titulek (title) na „Česko-anglický slovník“.
- Vložit do okna aplikace komponentu JScrollPane (přes celou šířku okna a přibližně $\frac{3}{4}$ výšky okna) a na ni komponentu JTable. Tuto komponentu pojmenovat **tabulka**.
- Nastavit **model** komponenty tabulka podle výše uvedeného obrázku.
- Přidat privátní atribut (proměnnou) model do třídy oknoSlovník:

```
public class oknoSlovník extends javax.swing.JFrame {  
    private final DefaultTableModel model;  
    ...  
}
```

- Do této proměnné uložit v konstruktoru odkaz na aktuální model tabulky:

```
public oknoSlovník() {  
    initComponents();  
    model = (DefaultTableModel) tabulka.getModel();  
}
```

- Přidat do okna čtyři tlačítka: Nový (název insert), Změnit (update), Smazat (delete) a Hledej (search). Tlačítkům Změnit a Smazat nastavit vlastnost enabled na false.

Výpis záznamů z databáze

1. Vytvoření atributu (proměnné) spojení pro uložení připojení k databázi:

```
public class oknoSlovník extends javax.swing.JFrame {  
    private final DefaultTableModel model;  
    private Connection spojeni;
```

2. Vytvoření metody pro spojení s MySQL serverem prostřednictvím JDBC connector:

```
private boolean dbConnection() {  
    try {  
        spojeni = (Connection) DriverManager.getConnection  
("jdbc:mysql://localhost/slovník?useUnicode=true&characterEncoding=utf-8", "root", "");  
        return true;  
    } catch (SQLException ex) {  
        JOptionPane.showMessageDialog(this, "Nedošlo k připojení databáze",  
                                        "Chyba", JOptionPane.ERROR_MESSAGE);  
        return false;  
    }  
}
```

- Vyvolání této metody v konstruktoru:

```
public oknoSlovník() {  
    ...  
    if (!dbConnection()) {  
        System.exit(0);  
    }  
}
```

3. Metoda pro získání všech záznamů z databázové tabulky do dynamické sady:

```
private ResultSet getAllRecords() {  
    ResultSet vysledky = null;  
    try {  
        PreparedStatement dotaz = spojeni.prepareStatement("SELECT * FROM slovnícek");  
        vysledky = dotaz.executeQuery();  
    } catch (SQLException ex) {  
        JOptionPane.showMessageDialog(this, "Chyba při komunikaci s databází",  
                                        "Chyba", JOptionPane.ERROR_MESSAGE);  
    }  
    return vysledky;  
}
```

4. Metoda pro výpis dynamické sady do komponenty tabulka (JTable):

```
private void listData(ResultSet data) {  
    /* Odstranění všech řádků z tabulky */  
    for (int i = tabulka.getRowCount() - 1; i >= 0; i--) {  
        model.removeRow(i);  
    }  
    try {  
        /* Vložení řádků do tabulky a jejich naplnění daty ze získané dynamické sady */  
        while (data.next()) {  
            int id = data.getInt(1);  
            String cesky = data.getString("cs");  
            String anglicky = data.getString("en");  
            model.addRow(new Object[]{id, cesky, anglicky});  
        }  
        /* Zapnutí nebo vypnutí tlačítek Změnit a Smazat v závislosti na existenci záznamů  
        (řádků) v tabulce */  
        if (tabulka.getRowCount() > 0) {  
            tabulka.setRowSelectionInterval(0, 0); /* Označení prvního řádku tabulky */  
            update.setEnabled(true);  
            delete.setEnabled(true);  
        } else {  
            update.setEnabled(false);  
            delete.setEnabled(false);  
        }  
    }  
}
```

```

} catch (SQLException ex) {
    JOptionPane.showMessageDialog(this, "Chyba při komunikaci s databází",
                                   "Chyba", JOptionPane.ERROR_MESSAGE);
}
}

```

- Použití obou metod v konstruktoru:

```

public oknoSlovník() {
    ...
    listData(getAllRecords());
}

```

- Výsledkem bude vypsání záznamů z databáze do tabulky:



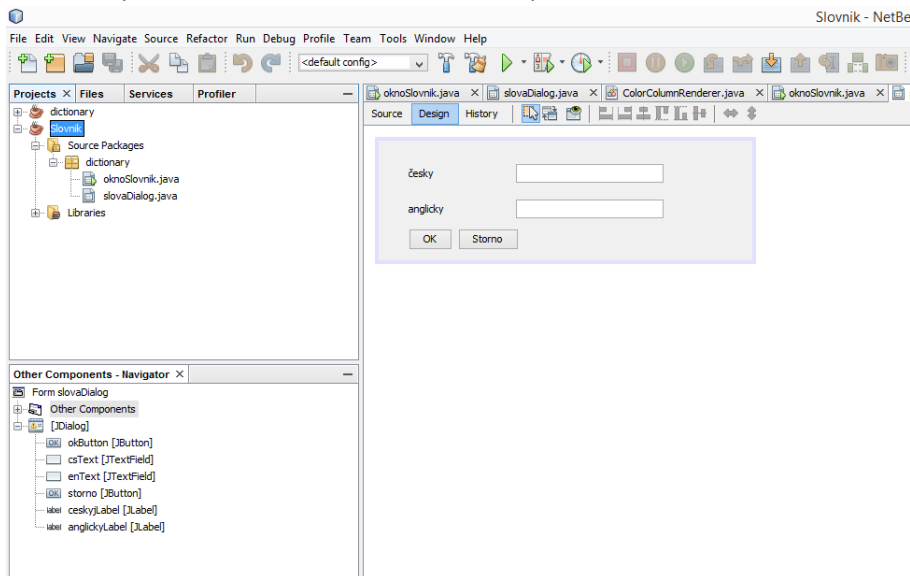
id	cs	en
1	počítač	computer
4	plamen	flame
6	ohně	fire
7	slovo	word
9	stůl	table
10	kniha	book
11	ulice	street
12	přidat	add
13	vymazat	delete
14	hledat	find
15	zeď	wall
16	cesta	road
17	řidič	driver

Nový Změnit Smazat Hledej

Vytvoření dialogového okna pro vytváření a změnu záznamů

1. Vytvoření modálního dialogového okna:

- Z místní nabídky nad označením balíčku vybrat New JDialog Form.
- Vyplnit v průvodci název dialogového okna – slovaDialog.
- Smazat z kódu dialogového okna metodu Main.
- Nastavit titulek okna na „Editor slovíček“.
- Vytvořit uživatelské rozhraní okna podle obrázku:



2. Doplnění kódu dialogového okna:

```
public class slovaDialog extends javax.swing.JDialog {
    private String actionButton = "Storno";

    public slovaDialog(java.awt.Frame parent, boolean modal, String[] slova) {
        super(parent, modal);
        initComponents();
        csText.setText(slova[0]);
        enText.setText(slova[1]);
    }

    public String showDialog() {
        /* Zviditelní dialogové okno */
        this.setVisible(true);
        /* Vrací název stisknutého tlačítka */
        return actionButton;
    }

    public String getAnglicky() {
        return enText.getText();
    }

    public String getCesky() {
        return csText.getText();
    }
}
```

3. Vytvoření ohlasových metod pro tlačítka OK a Storno:

```
private void okButtonActionPerformed(java.awt.event.ActionEvent evt) {
    actionButton = "OK";
    /* Uvolní objekt formuláře z paměti */
    this.dispose();
}

private void stornoActionPerformed(java.awt.event.ActionEvent evt) {
```

```

        actionPerformed = "Storno";
        this.dispose();
    }

```

4. Vložení nového záznamu do databáze – metoda v oknoSlovník:

```

private int insertRecord(String enWord, String csWord) {
    int numRows = 0;
    try {
        PreparedStatement dotaz =
            spojeni.prepareStatement("INSERT INTO slovnicek (cs, en) VALUES (?, ?)");
        dotaz.setString(1, csWord);
        dotaz.setString(2, enWord);
        numRows = dotaz.executeUpdate();
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(this, "Chyba při komunikaci s databází",
            "Chyba", JOptionPane.ERROR_MESSAGE);
    }
    return numRows;
}

```

5. Změna (aktualizace) záznamu v databázi – metoda v oknoSlovník:

```

private int updateRecord(int id, String enWord, String csWord) {
    int numRows = 0;
    try {
        PreparedStatement dotaz =
            spojeni.prepareStatement("UPDATE slovnicek SET cs=?, en=? WHERE id=?");
        dotaz.setString(1, csWord);
        dotaz.setString(2, enWord);
        dotaz.setInt(3, id);
        numRows = dotaz.executeUpdate();
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(this, "Chyba při komunikaci s databází",
            "Chyba", JOptionPane.ERROR_MESSAGE);
    }
    return numRows;
}

```

6. Vytvoření ohlasové metody pro tlačítko Nový:

```

private void insertActionPerformed(java.awt.event.ActionEvent evt) {
    String[] slova = {"", ""};
    slovaDialog slovaDialog = new slovaDialog(this, true, slova);
    /* Zobrazení dialogového okna metodou showDialog() */
    if (slovaDialog.showDialog().equalsIgnoreCase("OK")) {
        insertRecord(slovaDialog.getAnglicky(), slovaDialog.getCesky());
    }
    listData(this.getAllRecords());
}

```

7. Vytvoření ohlasové metody pro tlačítko Změnit:

```

private void updateActionPerformed(java.awt.event.ActionEvent evt) {
    int id = (int) tabulka.getModel().getValueAt(tabulka.getSelectedRow(), 0);
    String[] slova = {tabulka.getModel().getValueAt(tabulka.getSelectedRow(), 1).toString(),
        tabulka.getModel().getValueAt(tabulka.getSelectedRow(), 2).toString()};
    slovaDialog slovaDialog = new slovaDialog(this, true, slova);
    if (slovaDialog.showDialog().equalsIgnoreCase("OK")) {
        updateRecord(id, slovaDialog.getAnglicky(), slovaDialog.getCesky());
    }
    listData(this.getAllRecords());
}

```

Odstranění záznamů z databáze

1. Vytvoření metody pro vymazání záznamů z databáze:

```
private int deleteRecord(int id) {
    int numRows = 0;
    try {
        PreparedStatement dotaz = spojeni.prepareStatement("DELETE FROM slovnicek WHERE id=?");
        dotaz.setInt(1, id);
        numRows = dotaz.executeUpdate();
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(this, "Chyba při komunikaci s databází",
                                      "Chyba", JOptionPane.ERROR_MESSAGE);
    }
    return numRows;
}
```

2. Vytvoření ohlasové metody pro tlačítko Smazat:

```
private void deleteActionPerformed(java.awt.event.ActionEvent evt) {
    int id = (int) tabulka.getModel().getValueAt(tabulka.getSelectedRow(), 0);
    deleteRecord(id);
    listData(this.getAllRecords());
}
```

Vyhledání záznamu v databázi

1. Vytvoření metody pro vyhledání záznamu v databázi:

```
private ResultSet searchEn(String enWord) {
    ResultSet vysledky = null;
    try {
        PreparedStatement dotaz = spojeni.prepareStatement("SELECT * FROM slovnicek WHERE en=?");
        dotaz.setString(1, enWord);
        vysledky = dotaz.executeQuery();
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(this, "Chyba při komunikaci s databází",
                                      "Chyba", JOptionPane.ERROR_MESSAGE);
    }
    return vysledky;
}
```

2. Vytvoření ohlasové metody pro tlačítko Hledej:

```
private void searchActionPerformed(java.awt.event.ActionEvent evt) {
    String hledany = JOptionPane.showInputDialog(this, "Zadej anglické slovo");
    try {
        ResultSet data = searchEn(hledany);
        data.next();
        int id = data.getInt(1);
        String cesky = data.getString("cs");
        String anglicky = data.getString("en");
        JOptionPane.showMessageDialog(this, "Anglicky: \"" + anglicky + "\"\nČesky: \""
                                      + cesky + "\"", "Hledané slovo", JOptionPane.PLAIN_MESSAGE);
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(this, "Záznam nebyl nalezen",
                                      "Chyba", JOptionPane.ERROR_MESSAGE);
    }
}
```