

dr inż. Krzysztof Panczerz
Programowanie obiektowe
Obsługa baz danych w języku Java

Baza danych:

Serwer: localhost ▶ Baza danych: kredyt ▶ Tabela: kredyt1

Przeglądaj Struktura SQL Szukaj Dodaj Eksport Import Operacje Wyczyść Usuń

	Pole	Typ	Metoda porównywania napisów	Atrybuty	Null	Domyślnie	Dodatkowo	Działanie					
<input type="checkbox"/>	ID	int(11)			Nie	None	auto_increment						
<input type="checkbox"/>	Oszczednosci	varchar(20)	latin1_swedish_ci		Nie	None							
<input type="checkbox"/>	Majatek	varchar(20)	latin1_swedish_ci		Nie	None							
<input type="checkbox"/>	Dochod	int(11)			Nie	None							
<input type="checkbox"/>	Ryzyko	varchar(20)	latin1_swedish_ci		Nie	None							

```
import java.awt.BorderLayout;
import java.awt.Container;
import java.awt.FlowLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.ScrollPaneConstants;
import javax.swing.table.DefaultTableModel;
```

```
public class Program extends JFrame implements ActionListener
{
    DefaultTableModel tableModel;
    JTable table;
    Container content;
    JScrollPane scrollPane;
    JPanel przyciski;
    JButton wczytaj, dodaj, zapisz;

    public Program()
    {
        super("Baza danych");

        this.setSize(800,600);

        this.setLayout(new BorderLayout());

        this.tableModel=new DefaultTableModel();
        this.tableModel.addColumn("ID");
        this.tableModel.addColumn("Oszczednosci");
        this.tableModel.addColumn("Majatek");
        this.tableModel.addColumn("Dochod");
        this.tableModel.addColumn("Ryzyko");
        this.table=new JTable(tableModel);
```

```
content=this.getContentPane();
content.setLayout(new BorderLayout());

this.scrollPane=new JScrollPane(table);

this.scrollPane.setVerticalScrollBarPolicy(
    JScrollPaneConstants.VERTICAL_SCROLLBAR_AS_NEEDED);

this.content.add(this.scrollPane, BorderLayout.CENTER);

this.przyciski=new JPanel(new FlowLayout());

this.wczytaj=new JButton("Wczytaj");
this.wczytaj.setActionCommand("wczytaj");
this.wczytaj.addActionListener(this);
this.przyciski.add(this.wczytaj);

this.dodaj=new JButton("Dodaj");
this.dodaj.setActionCommand("dodaj");
this.dodaj.addActionListener(this);
this.przyciski.add(this.dodaj);

this.zapisz=new JButton("Zapisz");
this.zapisz.setActionCommand("zapisz");
this.zapisz.addActionListener(this);
this.przyciski.add(this.zapisz);

this.content.add(this.przyciski, BorderLayout.EAST);

this.setDefaultCloseOperation(EXIT_ON_CLOSE);
this.setVisible(true);
}

public static void main(String[] args)
{
    new Program();
}

public void wczytajBaze()
{
    Connection polaczenie;

    try
    {
        Class.forName("com.mysql.jdbc.Driver").newInstance();
        polaczenie=DriverManager.getConnection("jdbc:mysql://localhost/kredyt",
            "root", "");

        System.out.println(polaczenie);

        Statement instrukcja=polaczenie.createStatement();
        ResultSet wynik=instrukcja.executeQuery("SELECT * FROM kredyt");

        while(wynik.next())
        {
            Object[] dane=new Object[5];
```

```
        dane[0]=wynik.getObject("ID");
        dane[1]=wynik.getObject("Oszczednosci");
        dane[2]=wynik.getObject("Majatek");
        dane[3]=wynik.getObject("Dochod");
        dane[4]=wynik.getObject("Ryzyko");
        this.tableModel.addRow(dane);
    }

    polaczenie.close();
}
catch(InstantiationException e)
{
    e.printStackTrace();
}
catch (IllegalAccessException e)
{
    e.printStackTrace();
}
catch(ClassNotFoundException e)
{
    e.printStackTrace();
}
catch (SQLException e)
{
    e.printStackTrace();
}
}

public void dodajWiersz()
{
    Object[] wiersz=new Object[5];
    this.tableModel.addRow(wiersz);
}

public void zapiszBaze()
{
    Connection polaczenie;

    try
    {
        Class.forName("com.mysql.jdbc.Driver").newInstance();
        polaczenie=DriverManager.getConnection("jdbc:mysql://localhost/kredyt",
            "root", "");

        Statement instrukcja=polaczenie.createStatement();
        instrukcja.executeUpdate("DELETE FROM kredyt1");

        for(int i=0; i<this.tableModel.getRowCount(); i++)
        {
            String sql="INSERT INTO kredyt1 SET ";
            sql+="Oszczednosci='"+this.tableModel.getValueAt(i, 1)+"', ";
            sql+="Majatek='"+this.tableModel.getValueAt(i, 2)+"', ";
            sql+="Dochod='"+this.tableModel.getValueAt(i, 3)+"', ";
            sql+="Ryzyko='"+this.tableModel.getValueAt(i, 4)+"'";

            System.out.println(sql);
        }
    }
}
```

```
        instrukcja.executeUpdate(sql);
    }

    polaczenie.close();
}
catch(InstantiationException e)
{
    e.printStackTrace();
}
catch (IllegalAccessException e)
{
    e.printStackTrace();
}
catch(ClassNotFoundException e)
{
    e.printStackTrace();
}
catch (SQLException e)
{
    e.printStackTrace();
}
}

public void actionPerformed(ActionEvent e)
{
    if(e.getActionCommand().equals("wczytaj"))
    {
        this.wczytajBaze();
    }

    if(e.getActionCommand().equals("dodaj"))
    {
        this.dodajWiersz();
    }

    if(e.getActionCommand().equals("zapisz"))
    {
        this.zapiszBaze();
    }
}
}
```