

dr inż. Krzysztof Pancierz
Programowanie obiektowe
Obsługa baz danych w języku Java przy użyciu Hibernate

Baza danych:

	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							

Plik konfiguracyjny Hibernate:

```
<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
    "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

    <session-factory>

        <!-- Database connection settings -->
        <property name="connection.url">jdbc:mysql://localhost/kredyt</property>
        <property name="connection.driver_class">com.mysql.jdbc.Driver</property>
        <property name="connection.username">root</property>
        <property name="connection.password"></property>

        <mapping resource="Kredyt.hbm.xml"/>

    </session-factory>

</hibernate-configuration>
```

Plik odwzorowania:

```
<hibernate-mapping>
    <class name="Kredyt" table="kredyt1">
        <id name="id" column="ID"/>
        <property name="oszczednosci" column="Oszczednosci"/>
        <property name="majatek" column="Majatek"/>
        <property name="dochod" column="Dochod"/>
        <property name="ryzyko" column="Ryzyko"/>
    </class>
</hibernate-mapping>

import java.util.ArrayList;
import org.hibernate.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
```

dr inż. Krzysztof Pancierz
Programowanie obiektowe
Obsługa baz danych w języku Java przy użyciu Hibernate

```
import org.hibernate.cfg.Configuration;

public class Glowna
{
    public static void main(String[] args)
    {
        SessionFactory fabrykaSesji = new
            Configuration().configure("hibernate.cfg.xml").buildSessionFactory();
        Session sesja = fabrykaSesji.openSession();

        Query zapytanie=sesja.createQuery("from Kredyt");

        ArrayList<Kredyt> rekordy = (ArrayList<Kredyt>)zapytanie.list();

        for(Kredyt k: rekordy)
        {
            System.out.println(k.getId()+" "+k.getOszczednosci());
        }

        sesja.close();
    }
}
```

```
import java.io.Serializable;

public class Kredyt implements Serializable
{
    private int id;
    private String oszczednosci;
    private String majatek;
    private int dochod;
    private String ryzyko;

    public Kredyt()
    {}

    public void setId(int id)
    {
        this.id=id;
    }

    public int getId()
    {
        return this.id;
    }

    public void setOszczednosci(String oszczednosci)
    {
        this.oszczednosci=oszczednosci;
    }

    public String getOszczednosci()
    {
        return this.oszczednosci;
    }
}
```

```
public void setMajatek(String majatek)
{
    this.majatek=majatek;
}

public String getMajatek()
{
    return this.majatek;
}

public void setDochod(int dochod)
{
    this.dochod=dochod;
}

public int getDochod()
{
    return this.dochod;
}

public void setRyzyko(String ryzyko)
{
    this.ryzyko=ryzyko;
}

public String getRyzyko()
{
    return this.ryzyko;
}
}
```