

Cél:

- Perzisztencia
- Java Persistence API, Hibernate perzisztenciaszolgáltatóval

I. feladat Egy entitást tartalmazó alkalmazás készítése.

1. lépés:

- Adatbázis elkészítése (Derby, MySql, ...)
- Adatbázis kapcsolat ellenőrzése

2. lépés:

- NetBeans Java Application
- Adatbázis driver hozzáadása
Project – Libraries – Add Library (MySql) vagy Add Jar/Folder (Derby)
- Hibernate JPA library hozzáadása

3. lépés:

- Perzisztencia egység (Persistence Unit)
Persistence Unit Name:
Persistence Library: Hibernate (esetleg TopLink)
Database Connection: kiválasztani jdbc:mysql_3306/oneentity
Table Generation Strategy: Create

Eredmény: *persistence.xml*

```
<?xml version="1.0" encoding="UTF-8"?>
  <persistence version="1.0"
    xmlns="http://java.sun.com/xml/ns/persistence"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/persistence
      http://java.sun.com/xml/ns/persistence/persistence_1_0.xsd">
    <persistence-unit name="JPAOneEntityPU" transaction-type="RESOURCE_LOCAL">
      <provider>org.hibernate.ejb.HibernatePersistence</provider>
      <properties>
        <property name="hibernate.connection.username" value="public"/>
        <property name="hibernate.connection.driver_class"
          value="org.apache.derby.jdbc.ClientDriver"/>
        <property name="hibernate.connection.password" value="public"/>
        <property name="hibernate.connection.url"
          value="jdbc:derby://localhost:1527/oneentity"/>
        <property name="hibernate.cache.provider_class"
          value="org.hibernate.cache.NoCacheProvider"/>
      </properties>
    </persistence-unit>
  </persistence>
```

Java technológiák 2010.

12. Gyakorlat

```
        <property name="hibernate.hbm2ddl.auto" value="update"/>
    </properties>
</persistence-unit>
</persistence>
```

4. lépés:

Entitás osztály

Project – Persistence -- Entity Class

```
@Entity
public class Person implements Serializable {
    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private Long id;

    private String firstName;
    private String lastName;
    ...
    //get, set, equals, hashCode, toString
}
```

5. lépés:

Perzisztálás

```
...
public static void main(String[] args) {
    EntityManagerFactory emf =
        Persistence.createEntityManagerFactory("JPAOneEntityPU");
    EntityManager em = emf.createEntityManager();
    em.getTransaction().begin();
    try {
        Person person = new Person();
        person.setFirstName("Margit");
        person.setLastName("Antal");
        //picture.setId(21);

        em.persist(person);
        em.getTransaction().commit();
    } catch (Exception e) {
        e.printStackTrace();
        em.getTransaction().rollback();
    }

    ...
    em.close();
}
```

Java technológiák 2010.

12. Gyakorlat

6. lépés

Az összes rekord lekérdezése:

```
String queryStr = "select c from Person c";
javax.persistence.Query query = em.createQuery(queryStr);
List<Person> itemList = query.getResultList();
```

II. feladat Két entitást tartalmazó alkalmazás készítése.

Adjon hozzá az alkalmazáshoz egy második entitást, majd ellenőrizze az eredményt az adatbázisrétegben.