Programming Applications with Databases

Exercise Set 9

1. Prepare the environment and execute the following examples presented during lectures: HelloWorld, FactoryCreatingCost, UsingEnumerations, UsingComponents, and IdentityMap.
   [2p]

2. Following inheritance mapping examples presented during lectures, implement the following hierarchy:

   - Part { OEM, Manufacturer, Number, Description },
   - Tire extends Part { Speed, Rating }
   - NewTire extends Tire { OnStock }
   - UsedTire extends Tire { ProductionYear, ComsumptionLevel }
   - RefurbishedTire extends UsedTire { ComsumptionLevelAfterFixing }

   Prepare two approaches using *table to hierarchy* and *table to joined-subclass* methods. Generate test data and perform experiments checking time for creating objects and retrieving objects. Compare both implemented approaches.
   [2p]

3. Introduce NHibernate to your project by creating a repository for each aggregate and implementing the following methods:

   - *Find(int id)*, can be based on the structure like *session.Get<User>(id)*, and
   - *FindAll()*, can be based on the structure like *session.CreateQuery("from User").List<User>()*.

   Ensure you introduce appropriate relations between objects within aggregates applying collections or associations.
   [2p]

4. Using the implementation prepared as a solution for the Exercise 3, take one repository and introduce there all presented during the lectures mapping methods, i.e. based on XML, Mapping By Code, Fluent, and Attributes.
   [2p]

5. Complete all remaining methods (i.e. create, update, delete) for each repository created in Exercise 3 and introduce generic repository implementation.
   [2p]

Paweł Rajba