

Einfach

1. Was ist unter dem Begriff *Objekt-relacionales Mapping (ORM)* zu verstehen? Welches Problem versucht ORM zu lösen und wie äußert sich der so genannten *Impedance Mismatch*? Geben Sie einige Beispiele für ORMs.
2. Klären Sie die Begriffe *Entity* und *Value Object* und geben Sie jeweils ein Beispiel. Was unterscheidet beide voneinander in Bezug auf Identität, Objektgleichheit und Lebenszyklus? *Hinweis*: Bei Value Objects lohnt es sich mehrere Beispiel aufzuzeigen.
3. Wie wird bei NHibernate die Beziehung zwischen Objektmodell und Datenbanktabelle hergestellt? Auf welche Arten kann man dieses Mapping ausdrücken?
4. Welche zentrale Aufgaben hat in NHibernate die *ISession*? Ist eine *ISession* mit einer Datenbankverbindung gleichzusetzen? Sind *ISessions* „teure“ Objekte und durch welches Objekt werden sie erzeugt?
5. Welche Möglichkeiten gibt es in NHibernate Abfragen an die Datenbank zu stellen? Zeigen Sie ein Beispiel für jede Variante. Gehen Sie auf das Konzept von *Projection Queries* ein, wozu benötigt man diese?
6. Was versteht man unter *Inheritance* (Objektorientierung) und wann macht es Sinn *Inheritance* zu benutzen. Welche Möglichkeiten gibt es im Entity Framework und bei NHibernate, um Ableitungen in der Datenbank abzubilden.
7. Wie funktioniert *Eager Loading* im Entity Framework. Welche Vorteile und Nachteile hat *Eager Loading*?
8. Was sind *Compiled Queries* im Entity Framework? Welche Vorteile und Nachteile haben *Compiled Queries*?

Fortgeschritten

9. Was verbirgt sich hinter dem Begriff *Persistence Ignorance*? Wie äußert sich *Persistence Ignorance* in einem Domänenmodell?
10. Was ist das *Repository Pattern*? Lohnt es sich aus Ihrer Sicht, für jede Entity ein eigenes Repository zu schreiben, oder bevorzugen Sie ein generisches Repository? Wie können Abfragen an das Repository in *Query Objects* verpackt werden?
11. Was ist unter einer *Unit of Work* zu verstehen? Wie ist die *Unit of Work* in NHibernate, Linq2Sql und dem Entity Framework implementiert?
12. Unterstützen Entity Framework und NHibernate den Aufruf von *Stored Procedures*? Erläutern Sie dies an einem Beispiel, gehen Sie dabei speziell auf die unterstützten Rückgabetypen ein.
13. Welchen Nutzen hat der *EntityState* in einer Entity? Welche Zustände kann der *EntityState* haben? Welche Probleme gibt es bei *Many-to-Many Relations* mit dem *EntityState*?
14. Ist es im Entity Framework möglich eine Datenbanktabelle in mehrere Entities aufzuteilen? Wenn ja wie? Erklären Sie dies an einem Beispiel.

15. Was ist das *SELECT N+1-Problem*? Wie kann man dieses Problem bei der Arbeit mit NHibernate identifizieren? (*Hinweis*: Es gibt ein gesondertes Tool dafür.)
16. Welchen Status (im Rahmen einer State Machine) können Entities in NHibernate in Verbindung mit der *ISession* einnehmen?

Herausforderung

17. Welche Locking-Strategien werden von Entity Framework und NHibernate unterstützt. Gehen Sie speziell auf Versionierung und Optimistic bzw. Pessimistic Locking ein.
18. Wie kann man die Persistenzlogik von Entity Framework und NHibernate erweitern, um z. B. automatisches Audit-Logging umzusetzen.
19. Was ist der Unterschied im Entity Framework zwischen *ObjectContext.GetObjectByKey* und dem *Enumerable.First* Methode? Welchen ist die bessere Methode zum laden eines einzelnen Datensatzes?
20. Was ist unter *Lazy Loading* zu verstehen? Wie wird Lazy Loading in NHibernate aktiviert und welches technische Konzept dient zur Umsetzung des aus Sicht der Anwendung *transparent* Lazy Loading? Kann Lazy Loading für bestimmte Abfragen abgeschaltet werden?