

JPA

Что такое JPA?

JPA – Java persistence API.

JPA реализует концепцию ORM(object-relational mapping).

Что такое Entity?

Entity – POJO класс, связанный с бд с помощью аннотации или xml конфигурации.

Entity класс должен:

- 1) Должен иметь пустой конструктор.
- 2) Не может быть final, не может использовать final поля.
- 3) Должен содержать поле @Id.

Что такое JPQL

JPQL – Java persistence query language. Используется для написания sql-подобных запросов, оперирующих сущностями.

Какие есть реализации JPA?

-Hibernate

-Toplink

-Eclipselink

-OpenJpa

-DataNucleus

-ObjectDb

Сравнение производительности JPA библиотек

	Hibernate	EclipseLink	OpenJPA	DataNucleus
1000 inserts	2621ms	1936ms	2383	4400
1000 finds	1297ms	577ms	1584	363
1000 updates	2039ms	2396ms	1485	314
1000 deletes	2279ms	2687ms	3382	2698

Hibernate

Configuration properties

hibernate.dialect

hibernate.show_sql

hibernate.format_sql

hibernate.default_schema

hibernate.default_catalog

hibernate.session_factory_name

hibernate.max_fetch_depth

hibernate.default_batch_fetch_size

hibernate.default_entity_mode

hibernate.order_updates

hibernate.generate_statistics

hibernate.use_identifier_rollback

hibernate.use_sql_comments

Configuration properties

hibernate.jdbc.fetch_size

hibernate.jdbc.batch_size

hibernate.jdbc.batch_versioned_data

hibernate.jdbc.factory_class

hibernate.jdbc.use_scrollable_resultset

hibernate.jdbc.use_streams_for_binary

hibernate.jdbc.use_get_generated_keys

hibernate.connection.provider_class

hibernate.connection.isolation

hibernate.connection.autocommit

hibernate.connection.release_mode

Configuration properties

hibernate.cache.provider_class

hibernate.cache.use_minimal_puts

hibernate.cache.use_query_cache

hibernate.cache.use_second_level_cache

hibernate.cache.query_cache_factory

hibernate.cache.region_prefix

hibernate.cache.use_structured_entries

Configuration properties

hibernate.transaction.factory_class

jta.UserTransaction

hibernate.transaction.manager_lookup_class

hibernate.transaction.flush_before_completion

hibernate.transaction.auto_close_session

Configuration properties

hibernate.current_session_context_class

hibernate.query.factory_class

hibernate.query.substitutions

hibernate.hbm2ddl.auto

hibernate.cglib.use_reflection_optimizer

Dialects

org.hibernate.dialect.DB2Dialect
org.hibernate.dialect.DB2390Dialect
org.hibernate.dialect.MySQLDialect
org.hibernate.dialect.MySQLMyISAMDialect
org.hibernate.dialect.Oracle9iDialect
org.hibernate.dialect.SybaseDialect
org.hibernate.dialect.SQLServerDialect
org.hibernate.dialect.InformixDialect
org.hibernate.dialect.IngresDialect
org.hibernate.dialect.MckoiDialect
org.hibernate.dialect.PointbaseDialect
org.hibernate.dialect.FirebirdDialect
org.hibernate.dialect.DB2400Dialect
org.hibernate.dialect.PostgreSQLDialect
org.hibernate.dialect.MySQLInnoDBDialect
org.hibernate.dialect.OracleDialect
org.hibernate.dialect.Oracle10gDialect
org.hibernate.dialect.SybaseAnywhereDialect
org.hibernate.dialect.SAPDBDialect
org.hibernate.dialect.HSQLDialect
org.hibernate.dialect.ProgressDialect
org.hibernate.dialect.InterbaseDialect
org.hibernate.dialect.FrontbaseDialect

Application Configuration

Самостоятельная работа

hibernate.cfg.xml

```
<?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>

<property name="connection.driver_class">?</property>
<property name="connection.url">?</property>
<property name="connection.username">?</property>
<property name="connection.password">?</property>
<property name="connection.pool_size">?</property>
<property name="dialect">?</property>
<property name="show_sql">?</property>
<property name="hbm2ddl.auto">?</property>
<mapping class="?" />
</session-factory>
</hibernate-configuration>
```

JPA

javax.persistence VS org.hibernate.annotations

Entities

@Entity

@Table

@Id

@GeneratedValue

@Column

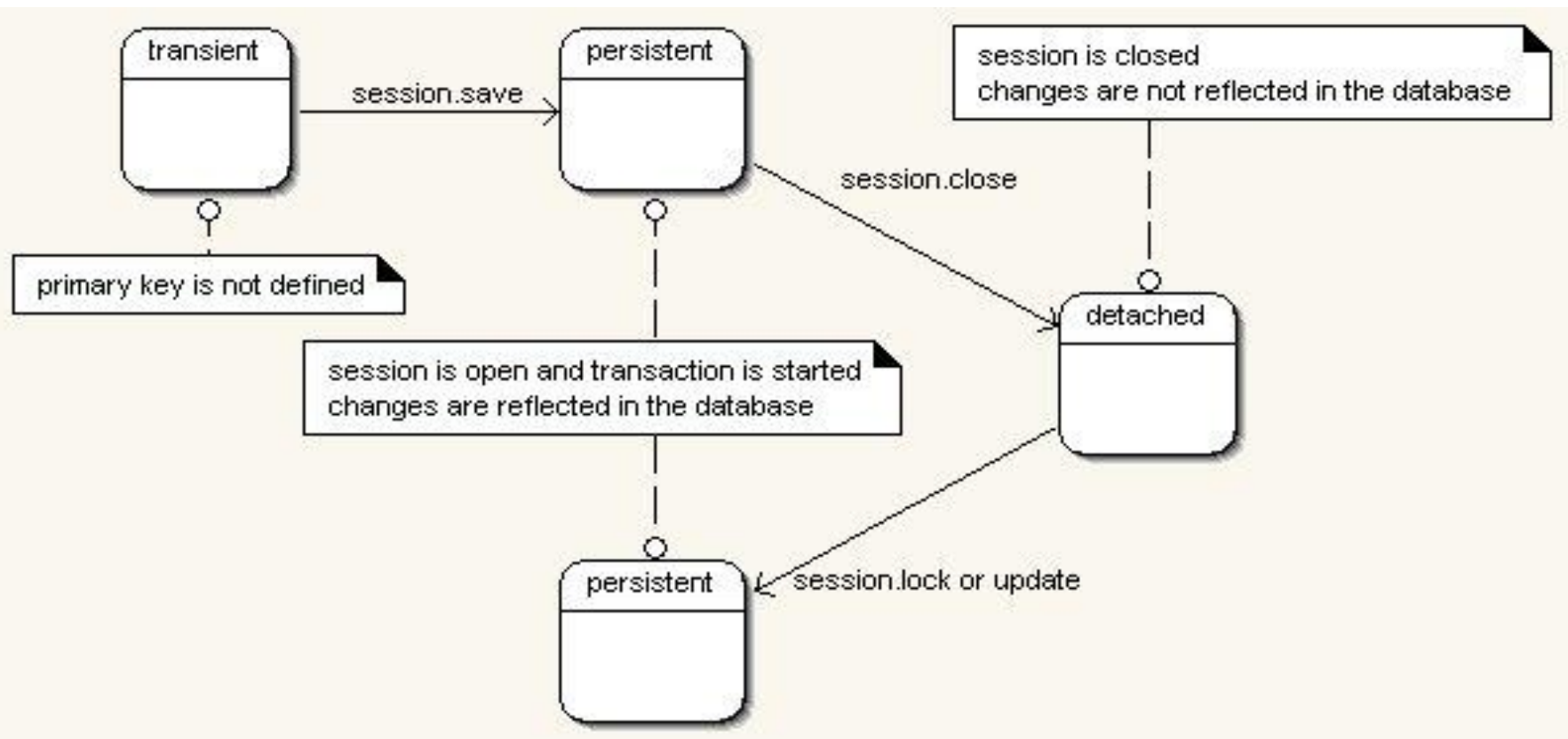
@OneToMany

@ManyToOne

@ManyToMany

@OneToOne

Entity states



Methods of session

persist, save

update, merge

delete

load, get

evict

@ManyToMany

@ManyToMany

@JoinTable(name="?")

@ManyToMany(mappedBy = "?")

Cascading

CascadeType:

ALL

DETACH

MERGE

PERSIST

REFRESH

REMOVE

FetchType

Eager

Lazy

Composite PK

@Embeddable

Serializable

Additional annotations

@Transient

@OrderColumn

@NamedQuery

@NamedNativeQuery

@Sort

@OrderBy

@Where

Inheritance

Single table

Joined

Table-per-class

Single table(table per hierarchy)

Parent:

```
@Inheritance(strategy = InheritanceType.SINGLE_TABLE)
```

```
@DiscriminatorColumn(name = "?",discriminatorType = DiscriminatorType.?)
```

Child:

```
@DiscriminatorValue("dog")
```

Table per class(table per concrete class)

Parent:

```
@Inheritance(strategy = InheritanceType.TABLE_PER_CLASS)
```

Child:

```
@AttributeOverrides({
```

```
  @AttributeOverride(name="?", column=@Column(name="?"))
```

```
})
```

Table per subclass, Joined

Parent:

```
@Inheritance(strategy = InheritanceType.JOINED)
```

Child:

```
@PrimaryKeyJoinColumn(name = "?", referencedColumnName = "?")
```

Isolation levels

Read uncommitted(Dirty read, Non repeatable read, Phantom read)

Read committed(Non repeatable read, Phantom read)

Repeatable Read(Phantom read)

Serializable

Isolation levels

```
session.doWork(connection -> {  
    connection.setTransactionIsolation(Connection.?);  
});
```

Optimistic vs pessimistic locking

@Version

where version = ?

Caching

Level 1 (session)

Level 2 (session factory)

Level 2 caching

EHCache

Infinispan

OSCache

SwarmCache

Modes:

NORMAL, GET, PUT

Level 2 caching, EHCache

POM.xml:

```
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-ehcache</artifactId>
  <version>?</version>
</dependency>
```

hibernate.cfg.xml:

```
<property name="cache.use_second_level_cache">true</property>
<property name="cache.provider_class">org.hibernate.cache.EhCacheProvider</property>
<property name="hibernate.cache.region.factory_class">org.hibernate.cache.ehcache.SingletonEhCacheRegionFactory</property>
<property name="net.sf.ehcache.configurationResourceName">cache-config.xml</property>
```

cache-config.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<ehcache>
<defaultCache maxElementsInMemory="100" timeToIdleSeconds="30" timeToLiveSeconds="120" />
</ehcache>
```

HQL

from ? alias where _____

select alias.? from ?

from ? alias where param > :paramIdentifier .setParameter("paramIdentifier", ?)

from ? x inner join fetch x.? as ?

Named queries

```
@NamedQuery(name = "?", query="?")
```

```
session.getNamedQuery("?")
```

Pagination

`query.setFirstResult`

`query.setMaxResults`

Query ->

list

iterate

uniqueResult

Criteria api

`session.createCriteria(Entity.class)`

`.add(Restrictions.?)`

`.addOrder(Order.?)`

`.setProjections(Projections.?)`

Interceptors

? extends EmptyInterceptor