

LOB Tiplerle Çalışmak



Binary ve LOB Tipleri: CLOB & BLOB

- java.lang.String, char[], Character[], veya java.sql.Clob tipi, **CLOB** sütunla eşlenmek istenirse **@Lob** anotasyonu kullanılır
- Benzer biçimde byte[], Byte[], or java.sql.Blob tipleri de **@Lob** anotasyonu kullanılarak **BLOB** sütunla eşlenir

```
@Entity
public class Image {
    ...

    @Lob
    private String description;

    @Lob
    private byte[] content;
}
```

Binary ve LOB Tipleri: CLOB & BLOB

```
@Entity
public class Image {

    ...

    @Lob
    private java.sql.Clob description;

    @Lob
    private java.sql.Blob content;
}
```

Binary ve LOB ORM Tipleri

| Java Tipi | Eşleme Tipi | SQL Tipi |
|------------------------|--------------|-----------|
| byte[] | binary | VARBINARY |
| java.lang.Serializable | serializable | VARBINARY |
| java.lang.String | text | CLOB |
| java.sql.Clob | clob | CLOB |
| java.sql.Blob | blob | BLOB |

Binary ve LOB Tipleri: CLOB & BLOB

- `Hibernate.getLobLocator(session).createBlob/`
`createClob` metodları `java.sql.Blob/Clob`
değerlerini oluşturmak ve set etmek için
kullanılabilir

`byte[]` veya `InputStream`
tipinde bir nesnedir

```
Blob blobContent = Hibernate.getLobCreator(session).createBlob(content);  
Clob clobDesc = Hibernate.getLobCreator(session).createClob(description);
```

```
Image image = new Image();  
image.setContent(blobContent);  
image.setDescription(clobDesc);
```

`String` veya `Reader` tipinde
bir nesnedir

Binary ve LOB Tipleri: Serializable

- Hibernate ve JPA, **Serializable** tipinde bir property'nin değerini java serialization ile yönetebilir
- user preferences, login session, workspace gibi **verileri tutumak için idealdir**
- Entity kaydedilirken Serializable property değeri **byte stream'e dönüştürülerek, VARBINARY/BLOB** sütununda saklanır
- Entity yüklendiği vakit ise DB'deki değer **deserialize** edilir ve değişkene atanır

LOB Değerlerin Yüklenmesi

- Normalde property değerleri DB'den **okuma sırasında hemen yüklenir**
- Bu yaklaşım **BLOB ve CLOB** değerler için **uygun değildir**
- İlgili entity yüklendiğinde LOB property aslında bir locator nesnedir, yani **pointer**'dir
- LOB değer property **erişildiğinde yüklenir**

LOB Değerlerin Yüklenmesi

- Yükleme ancak Hibernate **Session açık** olduğu müddetçe yapılabilir
- Ayrıca LOB değerlerin bu lazy davranışı **DB sürücülerine göre** değişiklik gösterebilir
- Eğer DB sürücüsü **desteklemiyorsa** LOB değer **eager** yüklenecektir

İletişim

- Harezmi Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- <http://www.java-egitimleri.com>
- info@java-egitimleri.com

