

15:53

SessionFactory Object

CoderX Ankit

- **SessionFactory** is an interface present in **org.hibernate** package.
- It is designed based on **factory design pattern**. It provides abstraction towards Session object creation.

I

14:13

SessionFactory Object

CoderX Ankit

- **SessionFactory** is an interface present in **org.hibernate** package.
- It is designed based on **factory design pattern**. It provides abstraction towards Session object creation.
- The configuration object internally reads the entries of configuration file, mapping file and creates different services like jdbc con pooling, caches, dialect, Tx manager, orm meta data^b and etc and uses them in creation of **SessionFactory object**. So, we can say that configuration is based on builder design pattern.



16:01

SessionFactory Object

CoderX Ankit

- **SessionFactory** is an interface present in **org.hibernate** package.
- It is designed based on **factory design pattern**. It provides abstraction towards Session object creation.
- The configuration object internally reads the entries of configuration file, mapping file and creates different services like jdbc con pooling, caches, dialect, Tx manager, orm meta data and etc and uses them in creation of **SessionFactory object**. So, we can say that configuration is based on builder design pattern.
- **Creating object of SessionFactory(I)** means creating the obj of implementation class that implements **org.hibernate.SessionFactory**, using this object we can create **Session object**.

16:28

SessionFactory Object

CoderX Ankit

- **SessionFactory** is an interface present in **org.hibernate** package.
- It is designed based on **factory design pattern**. It provides abstraction towards Session object creation.
- The configuration object internally reads the entries of configuration file, mapping file and creates different services like jdbc con pooling, caches, dialect, Tx manager, orm meta data and etc and uses them in creation of **SessionFactory object**. So, we can say that configuration is based on builder design pattern.
- **Creating object of SessionFactory(I)** means creating the obj of implementation class that implements **org.hibernate.SessionFactory**, using this object we can create **Session object**.
- It is **heavy weight** object because it contains multiple objects representing different components and services.

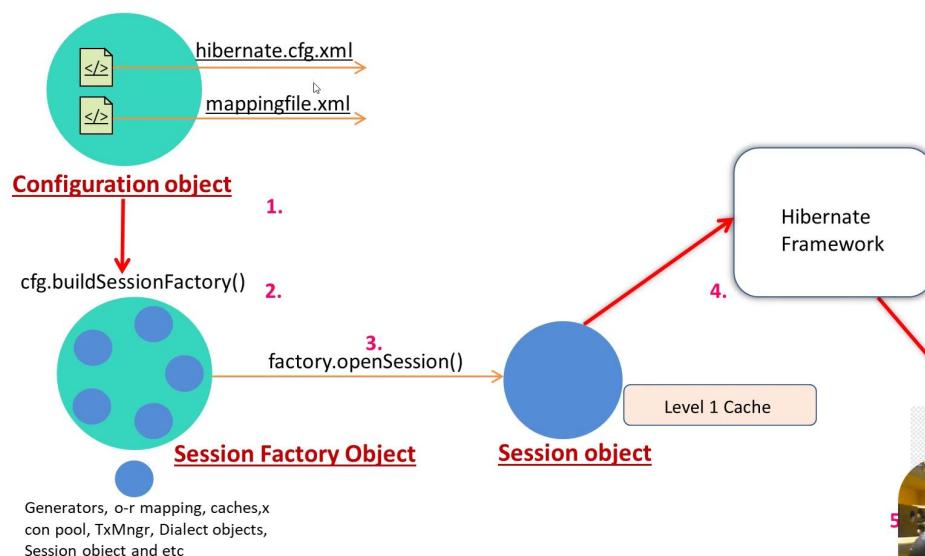


17:18

SessionFactory Object

CoderX Ankit

- **SessionFactory** is an interface present in **org.hibernate** package.
- It is designed based on **factory design pattern**. It provides abstraction towards Session object creation.
- The configuration object internally reads the entries of configuration file, mapping file and creates different services like jdbc con pooling, caches, dialect, Tx manager, orm meta data and etc and uses them in creation of **SessionFactory object**. So, we can say that configuration is based on builder design pattern.
- **Creating object of SessionFactory(I)** means creating the obj of implementation class that implements **org.hibernate.SessionFactory**, using this object we can create **Session object**.
- It is **heavy weight** object because it contains multiple objects representing different components and services.
- It is **immutable and thread safe** object.

20:3021:10

```
SessionFactory factory = cfg.buildSessionFactory();
```

Creates SessionFactory object, first by creating multiple objects by Configuration object and uses them in SessionFactory object creation (builder design pattern).



◀ ▶ ⏷ ⏸

No Ads

Remove Ads from pdf and websites

Pricing



Now you can use Askify in any websites

See How