**1. Cos'è un Framework**

Un framework è un insieme di librerie che aiutano a risolvere problemi specifici seguendo un'architettura predefinita. MyBatis è un framework ORM (Object-Relational Mapping), il che significa che aiuta a mappare gli oggetti Java con le tabelle di un database relazionale.

**2. Cos'è MyBatis**

MyBatis è un framework ORM che permette di eseguire query SQL scritte in file XML chiamati **mapper**. Questi file specificano le query da eseguire e come mappare i risultati alle classi Java.

**Struttura base di MyBatis**

1. **Mapper**: un file XML dove vengono scritte le query.
2. **DataSource**: un file di configurazione per gestire la connessione al database.
3. **SqlMapFactory**: una classe per gestire la connessione e le sessioni.

**3. Configurazione di un progetto con MyBatis**

**3.1 Aggiungere la libreria MyBatis**

Per aggiungere MyBatis al tuo progetto Java:

1. Clicca con il tasto destro sul progetto.
2. Vai su **Java Build Path** > **Classpath** > **Add External JARs**.
3. Seleziona la libreria MyBatis dal tuo filesystem e clicca su **Apply and Close**.

**3.2 Creare il file di configurazione (datasource.xml)**

Ecco un esempio di base del file **datasource.xml** per la configurazione di MyBatis:



**4. Connessione, Transazione e Sessione**

**4.1 Connessione**

La connessione rappresenta il collegamento tra l'applicazione Java e il database, che avviene tramite credenziali di accesso. In MyBatis, il **DataSource** è responsabile della connessione.

**4.2 Transazione**

Una transazione è un insieme di operazioni che devono essere completate con successo, altrimenti vengono annullate tramite **rollback**.

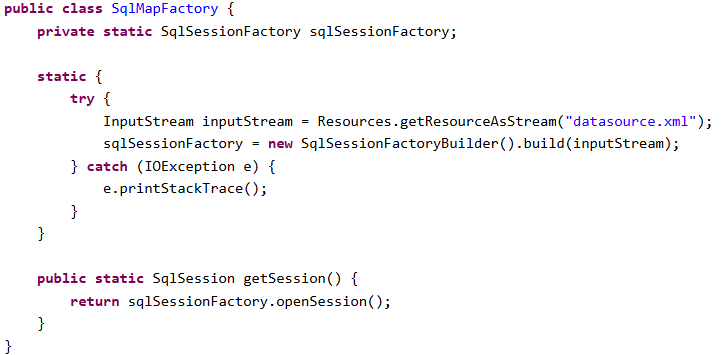
**4.3 Sessione**

Una sessione in MyBatis è l'ambito in cui vengono eseguite le query. Una sessione è creata tramite **SqlSessionFactory** e permette di eseguire le operazioni sul database.

**5. SqlMapFactory: Connessione**

La **SqlSessionFactory** è creata utilizzando il pattern Singleton per garantire che venga creata una sola istanza di sessione.

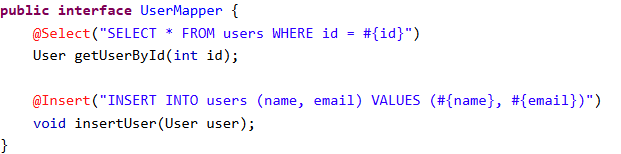
Esempio di implementazione della **SqlSessionFactory**:



**6. Interfacce Mapper**

Un **Mapper** è un'interfaccia Java che contiene metodi per eseguire le query definite nei file XML.

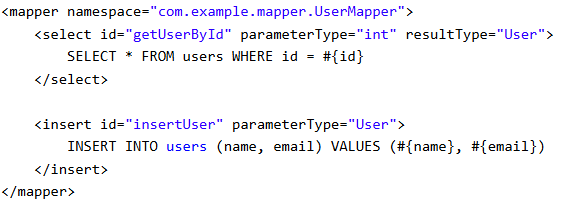
Esempio di **UserMapper.java**:



**7. Mapper XML**

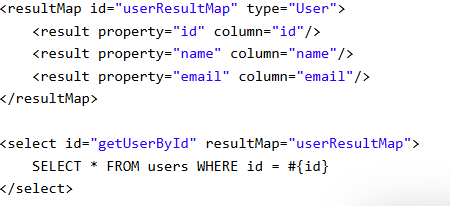
Ogni interfaccia **Mapper** ha un file XML associato, che contiene le query SQL.

Esempio di **UserMapper.xml**:

  
**8. ResultMap**

Il **ResultMap** mappa i risultati di una query SQL a oggetti Java. È utilizzato per personalizzare il mapping tra i campi della tabella e le proprietà della classe.

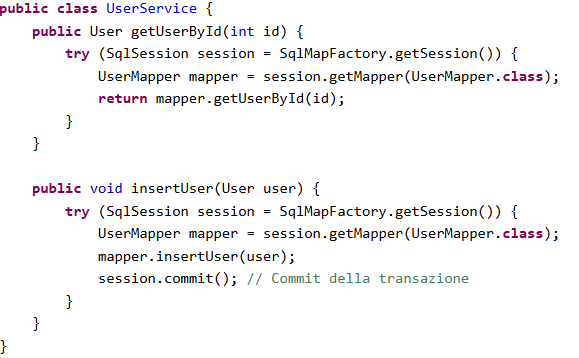
Esempio di **ResultMap** in **UserMapper.xml**:



**9. Esecuzione di una Query**

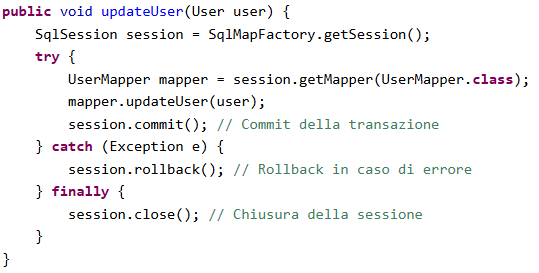
Per eseguire una query e ottenere un risultato, devi chiamare il metodo **getMapper** per ottenere l'implementazione dell'interfaccia e quindi invocare il metodo che esegue la query.

Esempio di esecuzione di una query con MyBatis:



**10. Gestione delle Transazioni in MyBatis**

Per garantire che le modifiche vengano persistite nel database, MyBatis usa i metodi **commit** e **rollback**.



**Conclusione**

Con MyBatis, il flusso di lavoro tipico include la creazione di file di configurazione per la connessione al database, l'uso di mapper per eseguire query SQL e la gestione di transazioni e sessioni per operazioni sicure e efficienti.

**Domande su MyBatis:**

1. **Cos'è MyBatis e in cosa differisce da altri framework ORM come Hibernate?**
   * Qual è il vantaggio principale dell'approccio basato sui mapper XML o annotazioni?
2. **Quali sono i passaggi fondamentali per configurare MyBatis in un progetto?**
   * Cos'è il file mybatis-config.xml e cosa contiene?
   * Come configuri il collegamento al database?
3. **Come funziona un mapper XML in MyBatis?**
   * Cos'è un mapper?
   * Come scrivi una semplice query SELECT in un file XML di mapper?
4. **Che differenza c'è tra l'uso di XML e annotazioni in MyBatis?**
   * Quando conviene usare uno rispetto all’altro?
5. **Come vengono gestite le relazioni tra tabelle con MyBatis?**
   * Cos'è il resultMap e come lo configuri?
   * Come rappresenti relazioni One-to-Many e Many-to-One?
6. **Come puoi eseguire query dinamiche con MyBatis?**
   * Quali tag dinamici sono disponibili (<if>, <choose>, <foreach>, ecc.) e come li utilizzi?
7. **Come si passa un parametro a una query in MyBatis?**
   * Cos'è il parameterType e come lo specifichi?
   * Come passi più parametri a una query?
8. **Come gestisce MyBatis il caching?**
   * Qual è la differenza tra il cache di primo livello e quello di secondo livello?
   * Quando ha senso abilitare il caching di secondo livello?
9. **Come funziona la gestione delle transazioni in MyBatis?**
   * Qual è il ruolo di SqlSession?
   * Come si commette o si annulla una transazione?
10. **MyBatis è considerato un framework ORM?**
    * Perché è definito "semi-ORM" e quali sono i vantaggi di questo approccio?
11. **Come gestisci aggiornamenti e cancellazioni con MyBatis?**
    * Come si scrive una query UPDATE o DELETE in un mapper XML o tramite annotazioni?
12. **Come MyBatis ti permette di gestire query complesse e join?**
    * Come scrivi una join in un mapper XML?
    * Come puoi mappare i risultati di una query con join in un oggetto Java?
13. **Quali sono le best practice per l'uso di MyBatis?**
    * Come scrivere mapper puliti e riutilizzabili?
    * Come gestire errori ed eccezioni in MyBatis?
14. **Quando ha senso usare MyBatis rispetto ad altri framework come JPA/Hibernate?**
    * In quali scenari è particolarmente adatto?
15. **Come esegui test per un'applicazione che usa MyBatis?**
    * Quali strumenti puoi usare per testare i mapper?