

**一级缓存是SqlSession级别的缓存**。在操作数据库时需要构造 sqlSession对象，在对象中有一个(内存区域)数据结构**（HashMap）**用于存储缓存数据。不同的sqlSession之间的缓存数据区域（HashMap）是互相不影响的。

一级缓存的作用域是同一个SqlSession，在同一个sqlSession中两次执行相同的sql语句，第一次执行完毕会将数据库中查询的数据写到缓存（内存），第二次会从缓存中获取数据将不再从数据库查询，从而提高查询效率。当一个sqlSession结束后该sqlSession中的一级缓存也就不存在了。Mybatis默认开启一级缓存。

二级缓存是**mapper级别的缓存**，多个SqlSession去操作同一个Mapper的sql语句，多个SqlSession去操作数据库得到数据会存在二级缓存区域，多个SqlSession可以共用二级缓存，二级缓存是跨SqlSession的。

  二级缓存是多个SqlSession共享的，其作用域是**mapper的同一个namespace**，不同的sqlSession两次执行相同namespace下的sql语句且向sql中传递参数也相同即最终执行相同的sql语句，第一次执行完毕会将数据库中查询的数据写到缓存（内存），第二次会从缓存中获取数据将不再从数据库查询，从而提高查询效率。Mybatis**默认没有开启二级缓存需要在setting全局参数中配置开启二级缓存。**

**避免使用二级缓存（可能会读到脏数据）**

针对一个表的某些操作不在他独立的namespace下进行。

例如在UserMapper.xml中有大多数针对user表的操作。但是在一个XXXMapper.xml中，还有针对user单表的操作。

这会导致user在两个命名空间下的数据不一致。如果在UserMapper.xml中做了刷新缓存的操作，在XXXMapper.xml中缓存仍然有效，如果有针对user的单表查询，使用缓存的结果可能会不正确。

更危险的情况是在XXXMapper.xml做了insert,update,delete操作时，会导致UserMapper.xml中的**各种操作充满未知和风险。**