mybatis基于注解实现CRUD

1、编辑接口IUserDao

```
public interface IUserDao {

    /**
     * 查询所有用户
     * @return
     */
    @Select("select * from user")
    List<User> findAll();

    /**
     * 保存用户
     * @param user
     */
    @Insert("insert into user(username,address,sex,birthday)values(#{username},#
{address},#{sex},#{birthday})")
    void saveUser(User user);

    /**
     * 更新用户
     * @param user
     */
    @Update("update user set username=#{username},sex=#{sex},birthday=#
{birthday},address=#{address} where id=#{id}")
    void updateUser(User user);

    /**
     * 删除用户
     * @param userId
     */
    @Delete("delete from user where id=#{id} ")
    void deleteUser(Integer userId);

    /**
     * 根据id查询用户
     * @param userId
     * @return
     */
    @Select("select * from user  where id=#{id} ")
    User findById(Integer userId);

    /**
```

```
     * 根据用户名称模糊查询
     * @param username
     * @return
     */
//    @Select("select * from user where username like #{username} ")
    @Select("select * from user where username like '%${value}%' ")
    List<User> findUserByName(String username);

    /**
     * 查询总用户数量
     * @return
     */
    @Select("select count(*) from user ")
    int findTotalUser();
}
```

2、测试：

```
public class AnnotationCRUDTest {
    private InputStream in;
    private SqlSessionFactory factory;
    private SqlSession session;
    private IUserDao userDao;

    @Before
    public  void init()throws Exception{
        in = Resources.getResourceAsStream("SqlMapConfig.xml");
        factory = new SqlSessionFactoryBuilder().build(in);
        session = factory.openSession();
        userDao = session.getMapper(IUserDao.class);
    }

    @After
    public  void destroy()throws  Exception{
        session.commit();
        session.close();
        in.close();
    }


    @Test
    public void testSave(){
        User user = new User();
        user.setUsername("mybatis annotation");
        user.setAddress("北京市昌平区");
        userDao.saveUser(user);
    }

    @Test
    public void testUpdate(){
        User user = new User();
        user.setId(57);
        user.setUsername("mybatis annotation update");
        user.setAddress("北京市海淀区");
        user.setSex("男");
        user.setBirthday(new Date());

        userDao.updateUser(user);
    }


    @Test
    public void testDelete(){
        userDao.deleteUser(51);
    }

    @Test
    public void testFindOne(){
        User user = userDao.findById(57);
        System.out.println(user);
    }


    @Test
    public  void testFindByName(){
//        List<User> users = userDao.findUserByName("%mybatis%");
        List<User> users = userDao.findUserByName("mybatis");
        for(User user : users){
            System.out.println(user);
        }
    }

    @Test
    public  void testFindTotal(){
        int total = userDao.findTotalUser();
        System.out.println(total);
    }
}
```