springboot □□mybatis-plus□□CRUD

CRUD

mybatis-plust□□□□□CRUD□**RSuDvice Service**□□□□□□□□□□□□□□□□CURD

□□□□□□CURD□□□

□□□□□□□□□**Service**□□□□□□□□□□□□□□□□

```
@Slf4j
public class CrudServiceImpl<ENT, DTO> implements CrudService<ENT, DTO> {



    // □□□□□□□□□□□□□□□□□mapper
    @Autowired
    protected BaseMapper<ENT> baseMapper;

    private Class<DTO> getDTOClass() {
        return (Class<DTO>) ReflectionKit.getSuperClassGenericType(this.getClass(),
CrudService.class, 1);
    }

    public List<DTO> list(@Param("ew") Wrapper<ENT> queryWrapper) {

        List<ENT> list = baseMapper.selectList(queryWrapper);
        if (list == null) {
            return null;
        }
        // □□□□□□□□□□□□□□□□
        List<DTO> listDTO = new ArrayList<>(list.size());
        Class<DTO> dtoClass = getDTOClass();
        for (ENT ent : list) {
            try {
                DTO dto = dtoClass.newInstance();
                BeanUtils.copyProperties(ent, dto);
                listDTO.add(dto);
            } catch (Exception ex) {
                log.error("CrudServiceImpl.list() error:", ex);
            }
        }
        return listDTO;
    }
}
```

□□□□□□□□□□□□CRUD□□

```
public interface UserService extends CrudService<UserEntity, UserDTO> {
}

@Service
public class UserServiceImpl extends CrudServiceImpl<UserEntity, UserDTO> implements
UserService {

}
```

□□□□□□□

□□□□□□□

□□□□□□□□□□

```
@Slf4j
public class ConvertUtils {

    public static <T> T sourceToTarget(Object source, Class<T> target) {
        if (source == null) {
            return null;
        }
        T targetObject = null;
        try {
            targetObject = target.newInstance();
            BeanUtils.copyProperties(source, targetObject);
        } catch (Exception e) {
            log.error("convert error ", e);
        }

        return targetObject;
    }

    public static <T> List<T> sourceToTarget(Collection<?> sourceList, Class<T> target) {
        if (sourceList == null) {
            return null;
        }
```

```
        List targetList = new ArrayList<>(sourceList.size());
        try {
            for (Object source : sourceList) {
                T targetObject = target.newInstance();
                BeanUtils.copyProperties(source, targetObject);
                targetList.add(targetObject);
            }
        } catch (Exception e) {
            log.error("convert error ", e);
        }

        return targetList;
    }
}
```

□□□□□□□□□□□□

```
@Service
@Slf4j
public class CrudServiceImpl<ENT, DTO> implements CrudService<ENT, DTO> {

    @Autowired
    protected BaseMapper<ENT> baseMapper;

    private Class<DTO> getDTOClass() {
        return (Class<DTO>) ReflectionKit.getSuperClassGenericType(this.getClass(),
CrudService.class, 1);
    }

    public List<DTO> list(@Param("ew") Wrapper<ENT> queryWrapper) {

        List<ENT> list = baseMapper.selectList(queryWrapper);
        // □□□□□□□□□□
        List<DTO> listDTO = ConvertUtils.sourceToTarget(list, getDTOClass());
        return listDTO;
    }
}
```

□□□ #3
□ □□□ □□□ 2 □□ 2025 03:36:44
□ □□□ □□□ 29 □□ 2026 16:40:23