

2025년 상반기 K-디지털 트레이닝

member 테이블

[KB] IT's Your Life



☑ 테이블 생성

```
-- 사용자 정보 테이블
drop table if exists tbl member;
create table tbl_member
                                                                   -- 사용자 id
                                 varchar(50) primary key,
           username
                                                                              -- 암호화된 비밀번호
                                 varchar(128) not null,
           password
                                             varchar(50) not null,
           email
                                 datetime default now(),
           reg date
           update_date datetime default now()
);
-- 사용자 권한 테이블
drop table if exists tbl_member_auth;
create table tbl_member_auth
                                                                   -- 사용자 id
                                 varchar(50) not null,
           username
                                                                              -- 권한 문자열 ROLE ADMIN, ROLE MANAGER, ROLE ME
                                             varchar(50) not null,
           auth
MBER 등
                                                                              -- 복합키
           primary key(username, auth),
           constraint fk authorities users foreign key (username) references tbl member(username)
);
```

🗹 데이터 추가

☑ 권한 데이터 추가

```
insert into tbl_member_auth(username, auth)
values

('admin','ROLE_ADMIN'),
    ('admin','ROLE_MANAGER'),
    ('admin','ROLE_MEMBER'),
    ('user0','ROLE_MEMBER'),
    ('user0','ROLE_MEMBER'),
    ('user1','ROLE_MEMBER'),
    ('user2','ROLE_MEMBER'),
    ('user3','ROLE_MEMBER'),
    ('user4','ROLE_MEMBER');

select * from tbl_member_auth order by auth;
```

- o ADMIN(최고 관리자)
 - ROLE_ADMIN, ROLE_MANAGER, ROLE_MEMBER
- o MANAGER(일반 관리자)
 - ROLE_MANAGER, ROLE_MEMBER
- o MEMBER(일반 회원)
 - ROLE_MEMBER

4

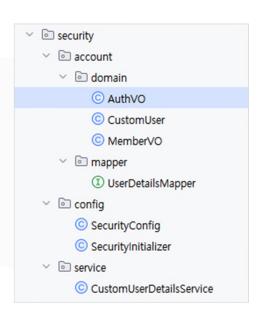
security.account.domain.AuthVO.java

```
@Data
public class AuthVO implements GrantedAuthority {
   private String username;
   private String auth;

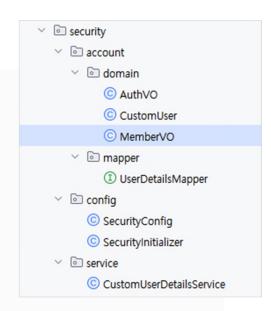
@Override
   public String getAuthority() {
      return auth;
   }
}
```

O GrantedAuthority 인터페이스

권한 정보 추출
 public interface GrantedAuthority extends Serializable {
 String getAuthority();
 }



security.account.domain.MemberVO.java



✓ resources::mybatis-config.xml

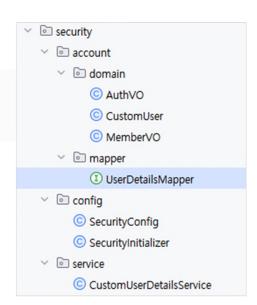
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN" "http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>

✓ □ resources

  <settings>
                                                                                                                  org.scoula
    <setting name="mapUnderscoreToCamelCase" value="true" />
                                                                                                                     > mapper
  </settings>
                                                                                                                     security.account.mapper
  <typeAliases>
                                                                                                                          MemberMapper.xml
    <package name="org.scoula.security.account.domain" />
                                                                                                                    @ application.properties
  </typeAliases>
                                                                                                                    </>log4j.xml
                                                                                                                     log4jdbc.log4j2.properties
</configuration>
                                                                                                                    mybatis-config.xml
```

UserDetailsMapper.java

- o tbl_membe와 tbl_member_auth를 Join처리
 - MemberVO의 authList를 설정
 - MyBatis의 resultMap으로 Join 쿼리 결과를 MemberVO로 맵핑



MyBatis의 <resultMap>

select m.username, password, email, reg_date, update_date, auth from

tbl_member m left outer join tbl_member_auth a on m.username = a.username

where m.username = 'admin';



	⁸⁸ username ▼	password	^{ABC} email	● reg_date	update_date	aut 🔻
1	admin	\$2a\$10\$EsIMfxbJ6NuvwX7N	admin@galapgos.org	2023-11-24 17:24:04	2023-11-24 17:24:04	ROLE_ADMIN
2	admin	\$2a\$10\$EsIMfxbJ6NuvwX7N	admin@galapgos.org	2023-11-24 17:24:04	2023-11-24 17:24:04	ROLE_MANAGER
3	admin	\$2a\$10\$EsIMfxbJ6NuvwX7N	admin@galapgos.org	2023-11-24 17:24:04	2023-11-24 17:24:04	ROLE_USER

1개의 MemerVO 인스턴스로 구성

```
@Data
public class MemberVO {
    private String username;
    private String password;
    private String email;
    private Date regDate;
    private Date updateDate;

private List<AuthVO> authList
```

ResultMap

○ select 쿼리 결과를 VO객체로 맵핑하는 규칙 정의

```
<ResultMap type="VO 객체 클래스명" id="식별자">
  <id property="VO 속성명" column="컬럼명"/>
  <result property="VO 속성명" column="컬럼명"/>
    ...
</ResultMap>
```

○ 1:N의 Join 결과 맵핑

```
<ResultMap type="VO 객체 클래스명" id="식별자">
  <id property="VO 속성명" column="컬럼명"/>
  <result property="VO 속성명" column="컬럼명"/>
    ...
  <collection property="부모객체의 콜렉션 속성명" resultMap="자식 테이블 맵핑 resutlmap ID">
  </ResultMap>
```

MyBatis의 <resultMap>

```
<resultMap id="memberMap" type="MemberVO">
 <id property="username"
                                                        column="username" />
 <result property="password"
                                 column="password" />
 <result property="email"
                                                        column="email" />
 <result property="regDate"
                                             column="reg date" />
 <result property="updateDate"
                                 column="update date" />
 <collection property="authList" resultMap="authMap" />
                                                                    List < AuthVO>로 구성해줌
</resultMap>
<resultMap id="authMap" type="AuthVO">
 <result property="username"
                                             column="username" />
 <result property="auth"
                                                        column="auth" />
</resultMap>
```

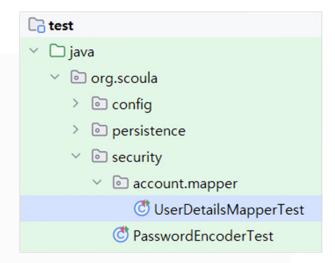
✓ resources :: UserDetailsMapper.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="org.scoula.security.account.mapper.UserDetailsMapper">
                                                                                                     resources
  <resultMap id="authMap" type="AuthVO">
                                                                                                     org.scoula
   <result property="username" column="username" />
   <result property="auth"
                                                        column="auth" />
                                                                                                        > mapper
 </resultMap>
                                                                                                        security.account.mapper
                                                                                                             UserDetailsMapper.xml
  <resultMap id="memberMap" type="MemberVO">
                                                                                                        application.properties
   <id property="username"
                                                                   column="username" />
   <result property="username"
                                             column="username" />
                                                                                                        </>√> log4j.xml
   <result property="password"
                                             column="password" />
                                                                                                        log4jdbc.log4j2.properties
   <result property="email"
                                                        column="email" />
                                                                                                        mybatis-config.xml
                                                        column="reg date" />
   <result property="regDate"
   <result property="updateDate"
                                 column="update date" />
   <collection property="authList" resultMap="authMap" />
  </resultMap>
```


SecurityConfig.java

```
@Configuration
@EnableWebSecurity
@Log4j2
@MapperScan(basePackages = {"org.scoula.security.account.mapper"})
public class SecurityConfig extends WebSecurityConfigurerAdapter {
...
}
```

test :: UserDetailsMapperTest.java



INFO jdbc.sqlonly(sqlOccurred:228) - select m.username, password, email, reg_date, update_date, auth from tbl_member m left outer join tbl_member_auth a on m.username = a.username where m.username = 'admin'

INFO org.scoula.security.account.mapper.UserDetailsMapperTest(get:24) - MemberVO(username=admin, password=\$2a\$10\$EsIMfxbJ6NuvwX7MDj4WqOYFzLU9U/ld dCyn0nic5dFo3VfJYrXYC, email=admin@galapgos.org, regDate=Fri Jan 17 13:12:21 KST 2025, updateDate=Fri Jan 17 13:12:21 KST 2025, authList=[AuthVO(username=admin, auth=ROLE_ADMIN), AuthVO(username=admin, auth=ROLE_MEMBER)])

 $INFO\ org. scoula. security. account. mapper. User Details Mapper Test (get: 27) - AuthVO (username=admin, auth=ROLE_ADMIN)$

INFO org.scoula.security.account.mapper.UserDetailsMapperTest(get:27) - AuthVO(username=admin, auth=ROLE_MANAGER)

INFO org.scoula.security.account.mapper.UserDetailsMapperTest(get:27) - AuthVO(username=admin, auth=ROLE_MEMBER)