User Request

`localhost:8080/user/create`

Controller

```java
@Controller
@RequestMapping(⊙∨"/user")
public class UserController {

    RoleService roleService;                        Injection
    UserService userService;

    public UserController(RoleService roleService, UserService userService) {
        this.roleService = roleService;
        this.userService = userService;
    }

    @GetMapping(⊙∨"/create")
    public String createUser(Model model){          Data Transfer(DTO) to view

        model.addAttribute( attributeName: "user", new UserDTO());
        model.addAttribute( attributeName: "roles", roleService.listAllRoles());  Business/Service Layer
        model.addAttribute( attributeName: "users", userService.listAllUsers());

        return "/user/create";                      View
    }
}
```

Service

```java
public interface RoleService {

    List<RoleDTO> listAllRoles();
    RoleDTO findById(Long id);

}
```

Service Impl

```java
@Service
public class RoleServiceImpl implements RoleService {

    private final RoleRepository roleRepository;            Injection
    private final RoleMapper roleMapper;

    public RoleServiceImpl(RoleRepository roleRepository, RoleMapper roleMapper) {
        this.roleRepository = roleRepository;
        this.roleMapper = roleMapper;
    }

    @Override
    public List<RoleDTO> listAllRoles() {

        List<Role> roleList = roleRepository.findAll();

        List<RoleDTO> roleDTOS = roleList.stream().map(obj -> roleMapper.convertToDto(obj)).collect(Collectors.toList());

        Convert Entity to DTO/Mapper

        return roleDTOS;                Return to Service
    }
}
```

Repository

```java
public interface RoleRepository extends JpaRepository<Role,Long> {

    Role findByDescription(String description);        Query

}
```

Returns Entity