



**Generics**

**reified**























































M





















































































































































































































































V

































































V





















































































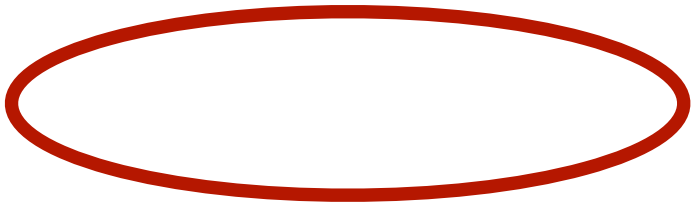


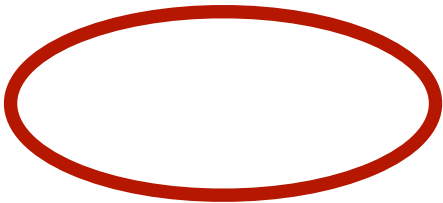












```
inline fun <reified T> ObjectMapper.readValue(content: String): T =  
    readValue(content, object : TypeReference<T>() {})
```

```
val client: User = mapper.readValue(json)
```

```
val client = mapper.readValue<User>(json)
```

```
val clients: List<User> = mapper.readValue(json)
```

# Generics

## reified

```
inline fun <reified T> ObjectMapper.readValue(content: String): T =  
    readValue(content, object : TypeReference<T>() {})
```

```
val client: User = mapper.readValue(json)
```

```
val client = mapper.readValue<User>(json)
```

```
val clients: List<User> = mapper.readValue(json)
```

```
val clients = mapper.readValue<List<User>>(json)
```

**Why do functions that used reified types have to be inline?**