

Interface Modifiable

🌐 Language: [Deutsch](#) · [English](#)

📖 Interface Modifiable

Write BOs directly via `save()` and `delete()` – the modern way instead of the deprecated provider methods.

REFERENZ

DEVELOPER

UPDATED: JUL 2026

APPLIES TO NUCLOS 4.2026.X

On this page

- [What is Modifiable?](#)
- [save\(\)](#)
- [delete\(id\)](#)
- [Example](#)
- [Related pages](#)

What is Modifiable?

Since Nuclos 4.28 the rule API provides the **Modifiable** interface: business objects are written directly via `save()` and `delete()` – without the [BusinessObjectProvider](#) (whose static `insert/update/delete` are deprecated).

save()

Saves the BO. A new BO is created (and gets a new id), an existing one is updated.

```
void save() throws BusinessException;
```

delete(id)

Deletes the BO with this id. Static method – callable without an instance.

```
static <PK> void delete(PK id) throws BusinessException;
```

Example

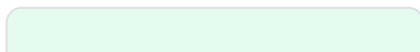
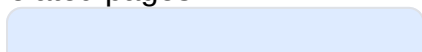
Readable, modern rule code with `get`, `changeStatus`, `save`, `refresh` (from 4.32) and static `delete`:

```
import org.nuclos.api.exception.BusinessException;
public class UtilsNeu {

    public static void workAuftrag(Long id1, Long id2) throws BusinessException {
        Auftrag auftrag = Auftrag.get(id1);
        auftrag.changeStatus(AuftragSM.State_20);
        auftrag.setName("NeuerName");
        auftrag.save();
        auftrag.refresh(); // ab v4.32

        Auftrag.delete(id2);
    }
}
```

Related pages





BusinessObjectProvider

Provider alternative.

[Open](#)



QueryProvider

Load data.

[Open](#)



Server-side rules

Rules.

[Open](#)