

PrintoutProvider

🌐 Sprache: **Deutsch** · [English](#)

📖 PrintoutProvider

Formulare aus Regeln ausführen – `run(format, bold)` liefert ein `NuclosFile`.

REFERENZ

ENTWICKLER

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

Auf dieser Seite

- [Wozu dient der PrintoutProvider?](#)
- [run\(format, bold\[, params\]\)](#)
- [Verwandte Seiten](#)

Wozu dient der PrintoutProvider?

Der `PrintoutProvider` führt ein **Formular** aus. Anders als beim Report wird ein Formular mit Daten befüllt – daher wird neben dem Ausgabeformat die **ID des Datensatzes** übergeben.

run(format, bold[, params])

Führt das Formular aus und liefert ein `NuclosFile` (z.B. als Anhang oder für den Mailversand). Optional lassen sich Parameter als Map übergeben.

```
public static NuclosFile run(OutputFormat format, Long boId) throws BusinessException;
public static NuclosFile run(OutputFormat format, Long boId, Map<String, Object> params) throws
BusinessException;
```

```
package org.nuclet.businessentity;

import org.nuclet.printout.FormularAngebotFuerKundenPO;
import org.nuclos.api.common.NuclosFile;
import org.nuclos.api.provider.PrintoutProvider;
import org.nuclos.api.rule.UpdateRule;
import org.nuclos.api.context.UpdateContext;
import org.nuclos.api.annotation.Rule;
import org.nuclos.api.exception.BusinessException;

public class AngebotFormularRegel implements UpdateRule {

    public void update(UpdateContext context) throws BusinessException {

        Auftrag a = context.getBusinessObject(Auftrag.class);

        NuclosFile result = PrintoutProvider.run(FormularAngebotFuerKundenPO.Deutsch_PDF, a.getKunde());
        a.insertAttachment(result, "Kommentar: Kundenspezifisches Hinweis-Formular");
    }
}
```



Report vs. Formular

Für **Reports** (ohne Datensatzbezug) siehe [ReportProvider](#).

Verwandte Seiten



ReportProvider



QueryProvider

Reports ausführen.

[Öffnen](#)

Daten laden.

[Öffnen](#)



BusinessObjectProvider

BOs schreiben.

[Öffnen](#)