

Inventor iLogic

Einführungskurs

Lektion 01 – Meine erste iLogic-Regel

Teilnehmer	_____
Datum	_____
Inventor-Version	2024
Projekt	iLogic Kurs

Praxisorientiertes Kurshandbuch zum selbstständigen Nachvollziehen der Lektion

1. Ziel dieser Lektion

In dieser ersten Lektion lernen Sie den grundlegenden Arbeitsablauf von iLogic kennen. Sie erstellen zunächst ein einfaches parametrisches Inventor-Bauteil und steuern es anschließend mit einer iLogic-Regel. Dabei geht es noch nicht um komplexe Programmierung, sondern um das Verständnis des Prinzips des regelbasierten Designs.

iLogic ist ein Dienstprogramm innerhalb von Inventor, mit dem benutzerdefinierte Skripte Daten und Geometrie erstellen oder verändern können. Dadurch lässt sich eigene Logik direkt in ein Inventor-Modell integrieren.

2. Lernziele

- ☐ ein parametrisches Inventor-Bauteil für iLogic vorbereiten
- ☐ Bemaßungen bzw. Modellparameter sinnvoll benennen
- ☐ eine iLogic-Regel anlegen, bearbeiten und ausführen
- ☐ Parameterwerte über eine Regel verändern
- ☐ Eigenschaften eines Features über iLogic verändern
- ☐ MessageBox und InputBox einsetzen
- ☐ String-, Zahlen- und logische Werte unterscheiden
- ☐ einen String mit CDbI() in einen Zahlenwert umwandeln
- ☐ einen Modellparameter auslesen und in einer Meldung anzeigen
- ☐ iLogic-Snippets einsetzen
- ☐ die Modellaktualisierung automatisieren

3. Ausgangssituation

Einstellung	Vorgabe
Inventor-Version	Inventor 2024
Inventor-Projekt	iLogic Kurs
Bauteilvorlage	Standard
Maßeinheit	Millimeter

4. Das Übungsbauteil erstellen

4.1 Projekt und neues Bauteil

Nr.	Arbeitsschritt	Kontrolle	✓
1	Inventor 2024 starten.	Inventor ist geöffnet.	☐
2	Projekt „iLogic Kurs“ aktivieren bzw. neu erstellen.	Projekt ist aktiv.	☐
3	Neues Teil mit der Vorlage „Standard“ erstellen.	Neues Bauteil ist geöffnet.	☐
4	Maßeinheit kontrollieren.	Millimeter.	☐

4.2 Rechteckskizze

Erstellen Sie eine neue 2D-Skizze auf der XY-Ebene.

Nr.	Arbeitsschritt	Eingabe / Einstellung	✓
1	Rechteckfunktion wählen.	Mitte mit 2 Punkten	☐
2	Ersten Wert eingeben.	40 mm	☐
3	Mit TAB in das zweite Eingabefeld wechseln.	20 mm	☐
4	Eingabe bestätigen und Rechteckfunktion beenden.	Enter / Escape	☐
5	Bemaßung 40 öffnen: Rechtsklick → Bemaßungseigenschaften.	Name = Länge	☐
6	Skizze beenden.	Beenden	☐

4.3 Extrusion

Richten Sie die Ansicht bei Bedarf über den ViewCube aus. Aktivieren Sie anschließend den Befehl „Extrusion“ und verwenden Sie die vorgeschlagene Höhe von 10 mm.

Sollmaß	Wert
Länge	40 mm
Breite	20 mm
Höhe	10 mm

☐ Der Grundkörper 40 × 20 × 10 mm ist erstellt.

4.4 Extrusionshöhe benennen

Klicken Sie im Browser mit der rechten Maustaste auf „Extrusion1“ und wählen Sie „Bemaßungen anzeigen“. Öffnen Sie die Bemaßungseigenschaften der Extrusionshöhe und vergeben Sie den Namen „Höhe“.

MERKE

Bemaßungen können auch über das Parameterfenster umbenannt werden. Verständliche Parameternamen machen iLogic-Regeln wesentlich leichter lesbar.

5. Die erste iLogic-Regel

5.1 Regel anlegen

Öffnen Sie auf der Registerkarte „Verwalten“ die Gruppe „iLogic“ und wählen Sie „Regel hinzufügen“.

Einstellung	Wert
Regelname	Lektion 01

Im iLogic-Editor befindet sich oberhalb des Codebereichs die Bauteilstruktur. Klicken Sie dort mit der rechten Maustaste auf „Extrusion1“ und wählen Sie „Aktuellen Status erfassen“. Inventor erzeugt daraus automatisch mehrere Codezeilen.

5.2 Kommentare verstehen

```
' Dies ist eine Kommentarzeile
```

MERKE

Jede Codezeile, die mit einem Apostroph beginnt, wird von Inventor ignoriert. Kommentare sollten erklären, was eine Regel macht und warum. Das erleichtert das spätere Verstehen und Ändern der Regel.

5.3 Länge per Regel verändern

Suchen Sie im automatisch erzeugten Code die Anweisung für den Parameter „Länge“ und ändern Sie den Wert von 40 mm auf 60 mm. Klicken Sie danach auf „Speichern und Ausführen“.

Prüfung	Sollzustand	✓
Länge nach Ausführung	60 mm	☐
Falls Modell noch unverändert	Lokale Aktualisierung ausführen	☐

6. Eigenschaften mit iLogic verändern

6.1 Farbe setzen

Öffnen Sie die Regel erneut. Verwenden Sie die Anweisung für Extrusion1 als Grundlage und setzen Sie die Eigenschaft Color auf den Textwert „red“.

```
Feature.Color("Extrusion1") = "red"
```

Speichern und führen Sie die Regel aus. Das Bauteil sollte nun rot dargestellt werden.

MERKE

True und False sind logische Werte. Ein Farbname wie "red" ist dagegen ein Textwert (String) und steht deshalb in Anführungszeichen.

6.2 MessageBox einsetzen

Ergänzen Sie vor der Farbänderung eine Meldung:

```
i = MessageBox.Show("Meine Nachricht", "Lektion 01 - Bauteilinformation")
```

☐ Die MessageBox erscheint beim Ausführen der Regel.

6.3 Reihenfolge der Befehle beobachten

Erweitern Sie die Regel so, dass der Ablauf sichtbar wird:

Reihenfolge	Aktion
1	Farbe auf "gold" setzen
2	MessageBox anzeigen
3	Farbe auf "red" setzen

MERKE

Die Anweisungen einer einfachen iLogic-Regel werden in der hier gezeigten Struktur von oben nach unten abgearbeitet. Die Reihenfolge der Codezeilen bestimmt deshalb den beobachtbaren Ablauf.

7. Werte vom Anwender abfragen

7.1 InputBox verwenden

Öffnen Sie den Editor und ergänzen Sie eine InputBox:

```
i = InputBox("Neuen Wert für Länge", "Werteingabe")
```

Die InputBox liefert einen Textwert (String). Der Inventor-Parameter „Länge“ benötigt dagegen einen Zahlenwert. Daher wird der eingegebene Text mit CDb1() in einen Double-Wert umgewandelt.

```
Länge = CDb1(i)
```

Testen Sie die Regel mit dem Wert 20.

Kontrolle	Sollwert	✓
Bauteillänge	20 mm	☐

7.2 Datentypen

Beispiel	Datentyp
"red"	String / Text
True	Boolean / logisch
"20"	String / Text
20	Zahlenwert
45.5	Zahlenwert

MERKE

Bei der Übergabe von Werten ist der Datentyp wichtig. CDb1() wandelt den Textwert aus der InputBox in einen Double-Zahlenwert um.

8. Aktuelle Parameterwerte auslesen

Öffnen Sie links im iLogic-Editor den Bereich „Snippets“ und anschließend „Parameter“. Fügen Sie den Eintrag „Parameter (dynamisch)“ ein und passen Sie ihn an:

```
p = Parameter("Länge")
```

Verwenden Sie den ausgelesenen Wert anschließend in der MessageBox:

```
"Bauteillänge: " & p
```

MERKE

Mit dem &-Zeichen werden Text und Werte zu einem gemeinsamen Text verbunden. Die Snippets im iLogic-Editor helfen dabei, häufig benötigte Befehle einzufügen, ohne die vollständige Syntax auswendig kennen zu müssen.

9. Modell automatisch aktualisieren

Fügen Sie über „Snippets → Dokument → Nach Abschluss aktualisieren“ die Aktualisierungsanweisung ein:

```
iLogicVb.UpdateWhenDone = True
```

Testen Sie die Regel mit einer neuen Länge von 45 mm.

Kontrolle	Sollzustand	✓
Eingabe	45	☐
Bauteillänge	45 mm	☐
Aktualisierung	automatisch	☐

10. Abschlussübung

Führen Sie die folgenden Aufgaben möglichst ohne das Kursskript aus. Die Übung zeigt, ob Sie den grundlegenden iLogic-Ablauf bereits selbstständig beherrschen.

- ☐ Aktuelle Länge des Bauteils auslesen.
- ☐ Aktuelle Länge in einer MessageBox anzeigen.
- ☐ Neue Länge mit einer InputBox abfragen.
- ☐ Eingabewert mit CDbI() in einen Zahlenwert umwandeln.
- ☐ Neue Länge dem Modellparameter zuweisen.
- ☐ Modell nach Abschluss automatisch aktualisieren.

Test	Eingabe	Ergebnis korrekt?
1	30 mm	☐
2	75 mm	☐
3	50 mm	☐

11. Wissens-Check

1. Wo finden Sie die iLogic-Funktionen in Inventor?

2. Wie legen Sie eine neue iLogic-Regel an?

3. Woran erkennen Sie eine Kommentarzeile?

4. Was ist ein String?

5. Warum wird bei der InputBox CDbI() verwendet?

6. Was bewirkt Parameter("Länge")?

7. Wofür wird das Zeichen & verwendet?

8. Was bewirkt iLogicVb.UpdateWhenDone = True?

9. Welchen Vorteil bieten die Snippets im iLogic-Editor?

12. Persönliche Abschlusskontrolle

Thema	Sicher	Noch üben
Parameter benennen	☐	☐
iLogic-Regel erstellen	☐	☐
Regel bearbeiten und ausführen	☐	☐
Parameter verändern	☐	☐
MessageBox einsetzen	☐	☐
InputBox einsetzen	☐	☐
Datentypen unterscheiden	☐	☐
Cdbl() einsetzen	☐	☐
Parameter auslesen	☐	☐
Snippets verwenden	☐	☐
Modell automatisch aktualisieren	☐	☐

13. Zusammenfassung

In dieser Lektion haben Sie den kompletten Grundablauf einer einfachen iLogic-Anwendung kennengelernt: Ein parametrisches Inventor-Modell wird vorbereitet, eine Regel wird erzeugt, Parameter und Eigenschaften werden verändert, Eingaben werden abgefragt und das Modell wird automatisch aktualisiert.

iLogic verwendet Visual Basic .NET. Für den Einstieg müssen Sie die Syntax jedoch nicht vollständig auswendig kennen: Der iLogic-Editor, automatisch erfasste Modellzustände und die integrierten Snippets unterstützen Sie beim Aufbau eigener Regeln.

„Es gibt nichts Gutes, außer: Man tut es.“

– Erich Kästner

Legen Sie los!