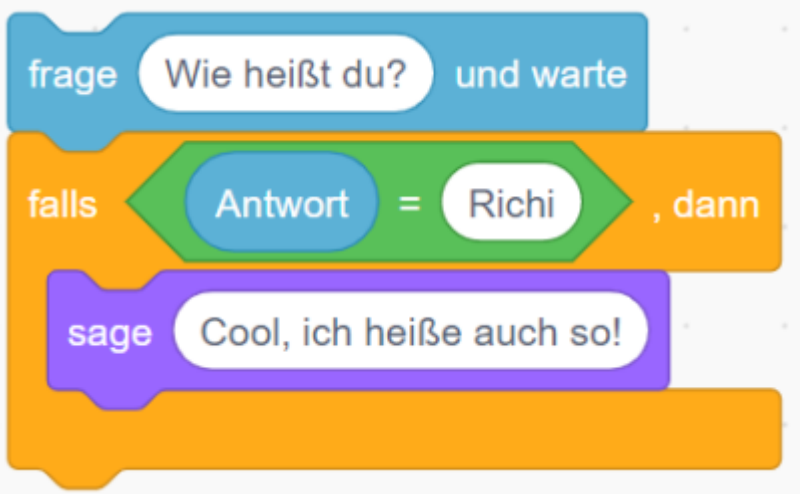
LU03a - Selektion/Auswahl

Einleitung

Applikationen sollen Abläufe basierend auf gewisse Bedingungen durchlaufen. Beispielweise dürfen Zigaretten, Alkohol oder Kreditkarten nicht an Minderjährige ausgegeben werden. Umgekehrt gibt es bei MacDonalds die Kinderportion nur für Kinder.

Bei Social Media-Apps wie Tinder, Instagramm oder ähnlichen markieren wir unsere Vorliegen durch Swipes bzw. Likes. Basierend auf diese Markierungen wird dann die Treffermenge entsprechend eingeschränkt.



Das vorliegende Kapitel beschäftigt sich mit diesen Entscheidungen in Applikationen (logische Entscheidungen).

Was sind SELEKTIONEN ?

Der Begriff Selektion bedeutet schlicht Auswahl – also das gezielte Herausgreifen bestimmter Elemente, Informationen oder Zustände aus einer größeren Menge.

Selektion bedeutet allgemein: Aus einer Menge von Daten gezielt bestimmte Elemente auswählen.

Operatoren

Um den Vergleich in Selektionen durchführen zu können braucht es sogenannte *OPERATOREN*. Dabei vergleicht der *OPERATOR* immer zwei Elemene, ähnlich wie in der Mathematik. Das Ergebnis des Vergleiches ist immer TRUE oder FALSE. Oder anders gesagt, der Vergleich ist korrekt oder er ist nicht

korrekt.

Operator	Bedeutung	Beispiel	Ergebnis
==	gleich (ohne Typprüfung)	5 == „5“	true
===	**streng gleich** (mit Typprüfung)	5 === „5“	false
!=	ungleich (ohne Typprüfung)	5 != „6“	true
!==	**streng ungleich** (mit Typprüfung)	5 !== „5“	true
>	größer als	7 > 3	true
<	kleiner als	2 < 4	true
>=	größer oder gleich	3 >= 3	true
< =	kleiner oder gleich	2 < = 5	true
&&	**und** (beide müssen true sein)	(x > 0 && x < 10)	true, wenn x zwischen 1 und 9, beide Bedingungen müssen erfüllt sein
||	**oder** (mindestens eine Bedingung true)	(x < 0 || x > 100)	true, wenn außerhalb 0–100, nur eine Bedinung muss erfüllt sein.

Arten von Selektionen

In der Informatik (und auch in Logik, Statistik oder Biologie) gibt es verschiedene Arten, wie man solche Selektionen unterscheiden kann. Grundsätzlich unterscheiden wir vier Arten von Selections:

1. **Einfach: if**
2. **Zweifach: if-else**
3. **Mehrfach: if-elsif-else**
4. **Mehrfach: switch-case**

1. Einseitige Seleketion: if

Die einseitige Selektion ist die grundlegendste Form der Auswahl. Eine Anweisung wird nur dann ausgeführt, wenn eine Bedingung wahr (true) ist. Wenn sie falsch (false) ist, passiert nichts.

Beispiel:

```
if (temperatur > 30) {
  console.log("Es ist heiss!");
}
```

Erläuterung: Wenn temperatur größer als 30 ist, wird die Nachricht ausgegeben. Sonst: Schweigen im Code-Wald.

Merkmal:

- Nur eine Richtung der Entscheidung.
- Keine Alternative, kein „sonst“.

2. Zweiseitige Selktion: if-else

Die zweiseitige Selektion prüft ob eine Bedinung erfüllt ist, und kann weitere Schritte unternehmen, wenn sie nicht erfüllt ist

„Wenn Bedingung erfüllt ist, dann tue etwas.“

Hier gibt es also zwei mögliche Wege:

- Wenn die Bedingung wahr ist → führe diesen Code aus.
- Wenn sie falsch ist → führe den anderen Code aus.

Beispiel

```
if (punktzahl >= 50) {
  console.log("Bestanden");
} else {
  console.log("Nicht bestanden");
}
```

Erläuterung: Immer wird etwas ausgeführt — je nachdem, ob die Bedingung erfüllt ist oder nicht.

Merkmal:

- Zweiwegentscheidung
- Erlaubt Alternative Aktionen
- Sehr häufig in Programmflusssteuerung

3. Mehrfache Seleketion: if-elsif-else

Das nachfolgende JavaScript-Codebeispiel entscheidet beispielsweise, basierend auf einen Wert der Variable *temperatur*, ob es draussen Kühl ist oder nicht.

Beispiel

```
let temperature = 15;
if (temperature > 30) {
  console.log("Es ist brütend heiss!");
} else if (temperature >= 20) {
  console.log("Angenehm warm.");
} else if (temperature >= 10) {
  console.log("Etwas frisch, aber okay.");
} else {
  console.log("Kalt – Jacke nicht vergessen!");
}
```

Die Logik des obigen Codebeispiels ist wie folgt:

- Ist der Wert der Variable grösser als 30, wird ausgegeben, dass es *brütend heiss ist*.
- Ist die Temperatur zwischen 20 und 29 Grad, wird *angenehm Warm* ausgegeben.

- Bei Werten zwischen 10 und 19 Grad ist die Ausgabe *Frisch, aber OK*
- Und ansonsten *Kalta - Jacke nicht vergessen*

4. Einfache Selketion: switch-case

Ein switch-case-Block vergleicht den Wert einer Variablen mit mehreren möglichen Fällen und führt den Code des **passenden Falls** aus – ähnlich wie eine Reihe von if-Abfragen, nur übersichtlicher.

Beispiel

```
let wochentag = "Mittwoch";
switch (wochentag) {
  case "Montag":
    console.log("Wochenstart – Kaffee intravenös!");
    break;
  case "Dienstag":
    console.log("Schon ein Tag geschafft.");
    break;
  case "Mittwoch":
    console.log("Halbzeit! Durchhalten.");
    break;
  case "Donnerstag":
    console.log("Fast geschafft.");
    break;
  case "Freitag":
    console.log("Endspurt!");
    break;
  default:
    console.log("Wochenende – endlich Ruhe.");
}
```

Wichtig: Jedes Switch-Case benötigt einen Default-Fall, also einen ELSE-Block.

Lernvideos

Was sind if-else?

Zusatzmaterial

- W3School- JavaScript Conditions
- W3School- JavaScript Operators
- SelfHTML - JS Selections