

Code aufräumen

In diesem Projekt räumen wir unseren Code auf: Er wird kompakter, besser lesbar und einfacher zu verstehen - und macht trotzdem noch das gleiche wie vorher. Echte Programmierer nennen das Refactoring und es ist eine der wichtigsten Tätigkeiten beim Programmieren.

1. Doppelten Code finden

Wir schauen uns heute die Skripte der Figur Fledi an, in dem die Nachricht "Level Start" verschickt wird:

```

Wenn [ ] angeklickt wird
  gehe zu x: 184 y: 135
  setze Größe auf 50
  sende Level Start an alle
  
```

```

Wenn [ ] angeklickt wird
  wiederhole fortlaufend
    falls wird Farbe [rot] berührt? , dann
      sende Level Start an alle
      gehe zu x: 184 y: 135
      setze Größe auf 50
  
```

```

Wenn [ ] angeklickt wird
  wiederhole fortlaufend
    falls wird Drehtür berührt? , dann
      sende Level Start an alle
      gehe zu x: 184 y: 135
      setze Größe auf 50
  
```

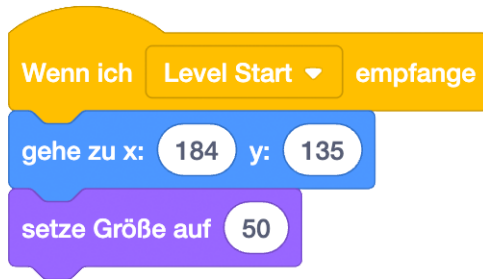
```

Wenn [ ] angeklickt wird
  wiederhole fortlaufend
    falls wird Farbe [grün] berührt? , dann
      spiele Klang Triumph ganz
      wechsle zum nächsten Bühnenbild
      sende Level Start an alle
      setze Größe auf 50
      gehe zu x: 184 y: 135
  
```

- Schau dir den Code noch einmal an und erinnere dich, was er tut.
- Kreise alle Stellen im Bild oben rot ein, an denen "Level Start" gesendet wird
- Jedes Mal, wenn wir "Level Start" senden tun wir gleichzeitig (direkt davor oder danach) auch noch ein paar andere Dinge, die beim Start (oder Neustart) eines Levels immer notwendig sind. Welche? Markiere im Bild oben grün die Blöcke, die wir immer in Kombination mit der "Level Start"-Nachricht ausführen.

2. Doppelten Code eliminieren

Wir fügen den folgenden Code bei Fledi ein:



Das bedeutet, jedes Mal, wenn die Nachricht "Level Start" gesendet wird, dann wird Fledi automatisch auf ihre Level-Startposition gesetzt und bekommt ihre normale Größe.

- Wie kann der Block-Code aus Aufgabe 1 jetzt vereinfacht werden? Entferne alle Blöcke, die jetzt nicht mehr nötig sind.
- Vergleiche den neuen und den alten Code. In welchem Code siehst du einfacher, was immer beim Start eines Levels zu tun ist? Wenn du die Position (184,135) ändern möchtest, in welchem Code geht das einfacher?

3. Schleifen zusammenfassen

Der Code, den wir in Aufgabe 2 erstellt haben, ist schon deutlich kompakter als der aus Aufgabe 1. Wir können ihn noch kompakter machen, in dem wir verschiedene “Wiederhole Fortlaufend”-Schleifen zusammenfassen.

Wir möchten zuerst das Prinzip verstehen: Hier ist ein Skript, das drei verschiedene Aufgaben permanent immer wieder durchführt:



Statt für jede Aufgabe eine eigene Schleife zu nutzen, kann man auch alle Aufgaben nacheinander in einer einzigen Schleife ausführen:



Überlege, welche drei Aufgaben in unserem Skript aus Aufgabe 2 immer wieder in einer Schleife ausgeführt werden. Ändere das Skript so, dass nur eine einzige “wiederhole fortlaufend”-Schleife benötigt wird.