



Programmieren mit dem MIT App Inventor 2

Wenn du dich mit Scratch gut auskennst, wirst du auch schnell mit dem MIT App Inventor 2 (AI2) zurecht kommen.

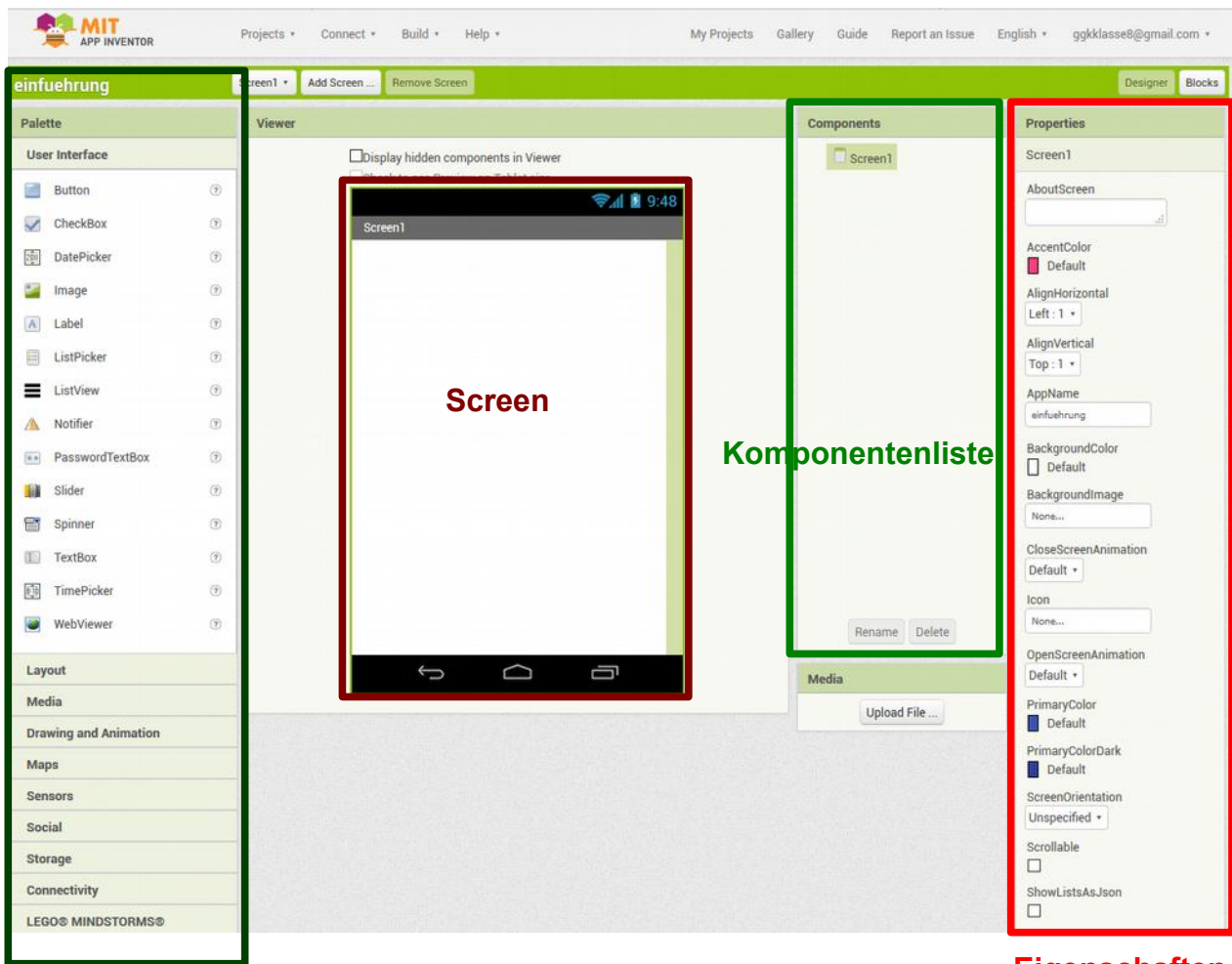
Wie du dich anmeldest, um den AI2 nutzen zu können, erfährst du von deiner Lehrerin. Sie zeigt dir auch, wie du deinen Programmcode testen kannst. Da gibt es mehrere Möglichkeiten, je nachdem, welche Voraussetzungen gegeben sind.

Zunächst lernst du die Oberfläche des AI2 kennen.

Von Scratch kennst du eine Bühne, auf der du die Figuren (Objekte) platzieren kannst, unter der Bühne befindet sich die Figurenliste, in der du sehen kannst, welche Figuren du auf deiner Bühne hast, die Blockpalette, in der du alle Blöcke findest, die du zum Programmieren brauchst und schließlich den Programmierbereich, in den du die Blöcke ziehen kannst, um ein Programm zu entwerfen. Du kannst gerne Scratch zum Vergleich noch einmal öffnen und nachschauen.

Im AI2 sieht es ein bisschen anders aus, aber du wirst dich schnell zurecht finden.

Im Designer siehst du folgende Oberfläche:



Komponentenpalette

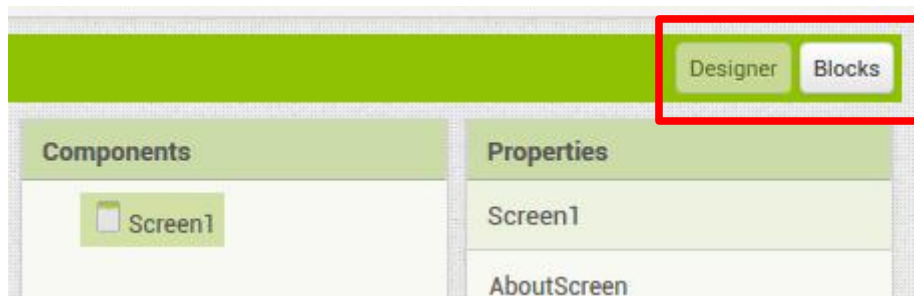
Eigenschaften



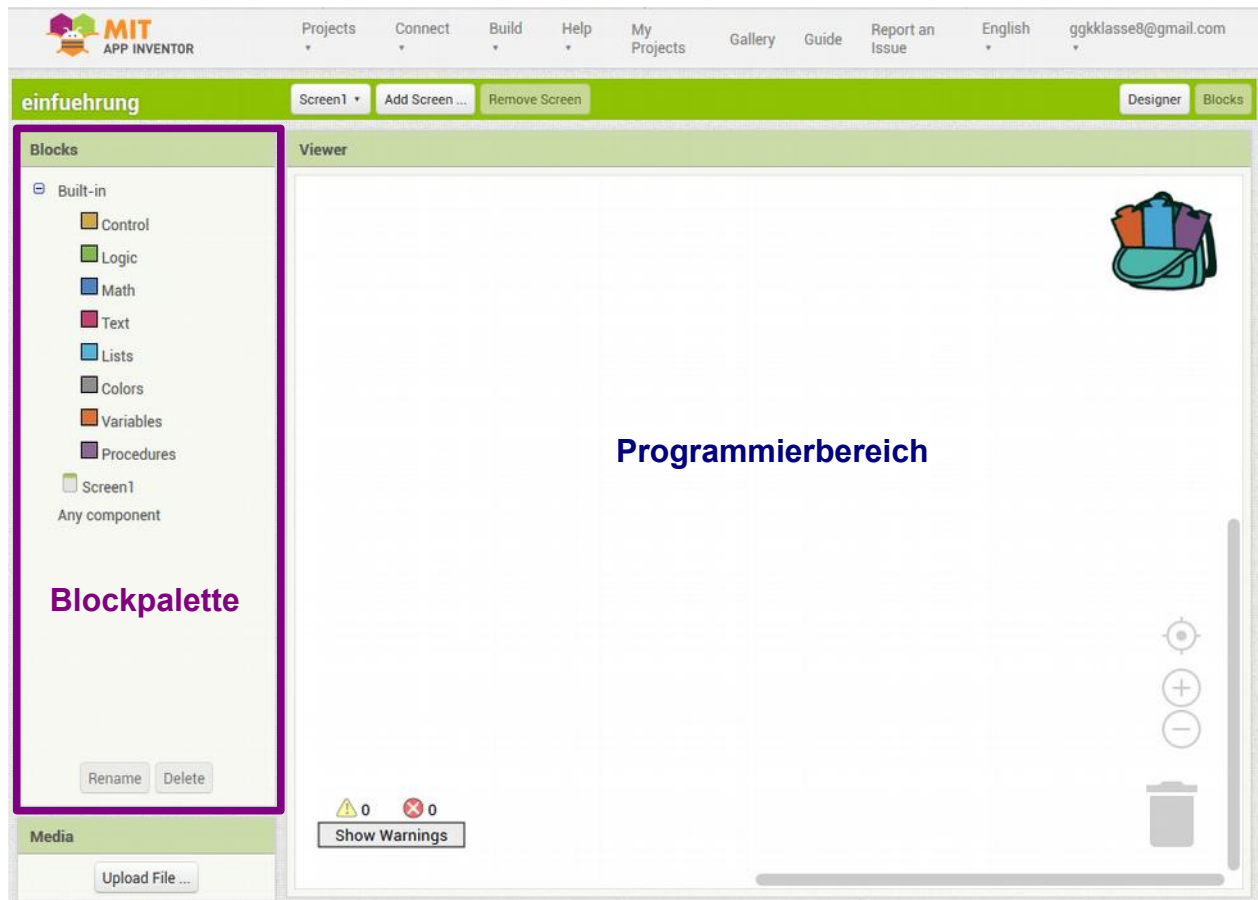
Wie der Name App-Inventor schon sagt, programmierst du hier im AI2 eigene Apps für Smartphones. Der **Screen** zeigt genau dessen Oberfläche. Du kannst dort Komponenten aus der **Komponentenpalette** einfügen.

Jede der eingefügten Komponenten siehst du dann aufgelistet in der **Komponentenliste**. Um die Eigenschaften einer Komponente im Entwicklungsmodus (also beim Programmieren) zu ändern, wählst du eine Komponente aus und änderst die gewünschten Dinge unter **Eigenschaften** ganz rechts auf der Oberfläche.

Bevor wir genau das bei ein paar Komponenten ausprobieren, schauen wir uns den Blockbereich an. Du kannst zwischen Designer und Blockbereich umschalten, wenn du die entsprechende Schaltfläche drückst:



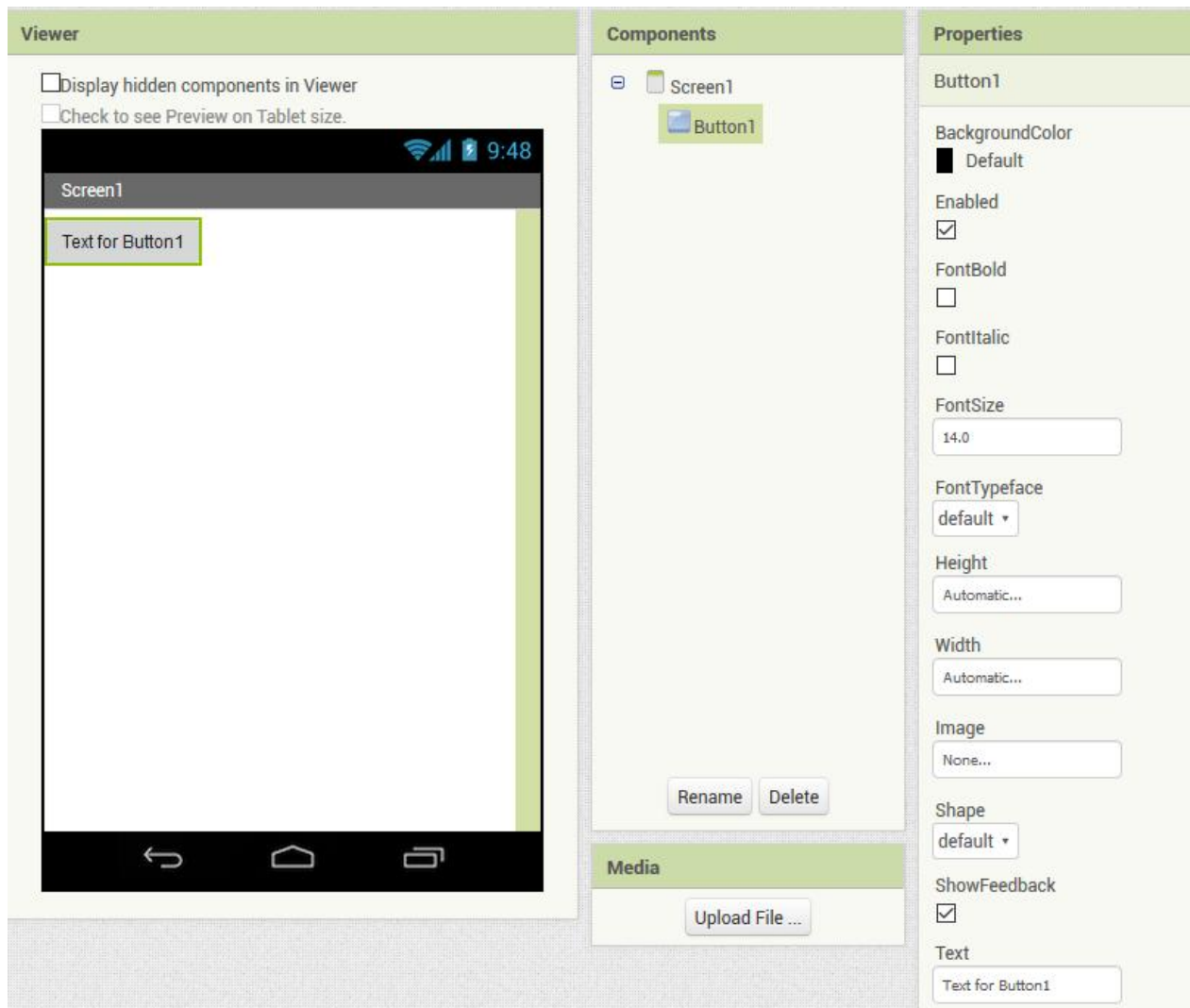
Im Blockbereich siehst du folgende Oberfläche:





Du kennst das schon so ähnlich aus Scratch. In der **Blockpalette** kannst du dir verschiedene Blöcke auswählen und mit gedrückter Maustaste in den **Programmierbereich** ziehen. Dort können sie wie gewohnt aneinander gehängt werden.

Wir wollen das an einem einfachen Beispiel ausprobieren. Gehe zurück in den Designer und ziehe aus der Komponentenpalette aus dem obersten Bereich (User Interface) eine Schaltfläche (Button) mit gedrückter Maustaste auf den Screen. Er wird automatisch links oben ins Eck gesetzt.



Siehst du die Veränderungen im Designer?

In der Komponentenliste wird der Button als Button1 aufgeführt und rechts sind Eigenschaften des Buttons angezeigt.

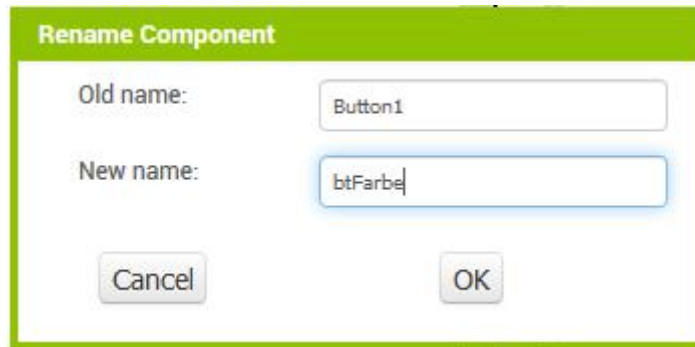
Gib dem Button zunächst einen neuen Namen. Hier ist es wichtig, dass du folgende Dinge beachtest:

1. Der Name einer Komponente darf keine Umlaute, keine Leer- und Sonderzeichen enthalten.
2. Der Name jeder Komponente sollte zu erkennen geben, um welche Art von Komponente es sich handelt. Der Namen eines Buttons könnte z.B. mit „bt“ beginnen.

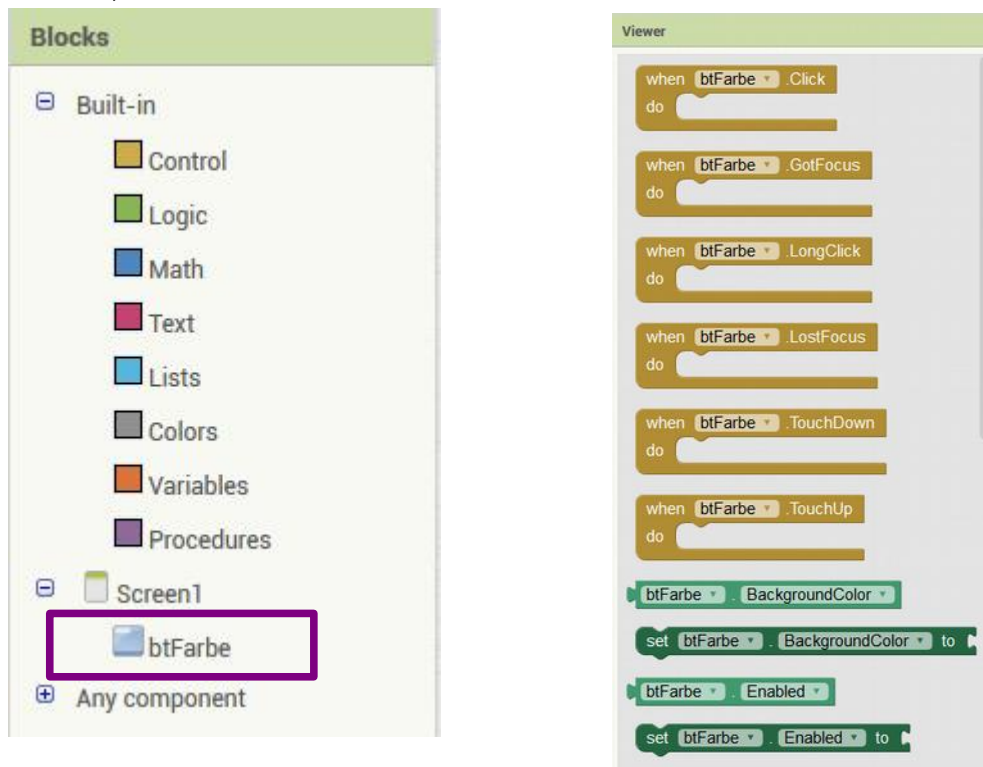


3. Um beim Programmieren die Übersicht nicht zu verlieren, soll der Name jeder Komponente sinnvoll sein. Wenn unser Button dafür zuständig ist, die Hintergrundfarbe des Screens zu ändern, so könnte er z.B. „btFarbe“ heißen.

Die Namensänderung kannst du ganz unten in der Komponentenliste vornehmen. Du findest dort einen Button mit Aufschrift „Rename“. Gib deinem Button den Namen „btFarbe“.



Ändere dann seine Eigenschaft „Text“ von „Text for Button1“ zu „Farbe ändern“. Hier darfst du jetzt Umlaute, Leer- und Sonderzeichen verwenden.



Wenn du jetzt in den Blockbereich umschaltest, findest du links in der Blockpalette auch deinen Button wieder. Auch für ihn werden Blöcke zur Verfügung gestellt.

Welche Blöcke es gibt, siehst du, wenn du dir in der Blockpalette etwas auswählst.

Klicke auf „btFarbe“. Es öffnet sich eine neue Palette, in der du alle Blöcke findest, die es speziell für deinen Button gibt. Im oberen Bereich findest du die Ereignisse, die bei deinem Button eintreten können (hellbraun – die Farben kennst du schon aus Scratch), im unteren findest du zu jeder Eigenschaft zwei Blöcke: einen, der den Wert der Eigenschaft liefert und einen, bei dem du den Wert der Eigenschaft ändern kannst.



Aufträge:

1. *Programmiere das Ereignis, das eintritt, wenn auf den Button geklickt wird, und lasse die Farbe des Buttons und seine Aufschrift ändern. Du kennst sicher die nötigen englischen Begriffe und findest auch Text und Farben in der Blockpalette.*
2. *Erweitere dein Programm so, dass man seinen Vornamen in ein Eingabefeld (TextBox) eingeben kann und beim Klick auf einen zweiten Button, eine Begrüßung in einem Textfeld (Label) erscheint.
Wähle wieder aussagekräftige Namen für deine Komponenten. Als Abkürzung für die TextBox kannst du „tb“ und für das Label „lb“ wählen. Um Zeichenketten aneinanderzuhängen, nutzt man eine „join“-Block (Text).*
3. *In einer Auswahlliste (ListPicker - „lp“) soll eine Obstsorte gewählt werden können. Nach der Auswahl wird das gewählte Obst in einer Bildkomponente (Image - „im“) angezeigt. Die gewünschten Bilddateien findest du im Tauschordner und kannst sie unter Media hochladen. Die Elemente, die in der Auswahlliste angezeigt werden sollen, gibst du bei der Eigenschaft „ElementsFromString“ durch Komma getrennt ein.*

Schaue dir jetzt zunächst einmal die Lösungsblätter zu den ersten drei Aufgaben an. Vielleicht entdeckst du noch etwas, das du brauchen kannst.

Mache dann mit Aufgabe 4 und 5 weiter.

4. *Vielleicht hast du dir schon überlegt, wie man Komponenten anders anordnen kann. Dazu brauchst du sog. Layouts. Teste aus, wie du Komponenten nebeneinander oder untereinander anordnen kannst. Spiele auch mit den Eigenschaften Breite (width) und Höhe (height) der Komponenten und des Layouts*
5. *Schaue dich jetzt noch ein bisschen auf der Designeroberfläche um und probiere weitere Komponenten aus. Schaue dann, was du auf der Blockoberfläche zu den Komponenten findest.
Schreibe einen Steckbrief einer Komponente. Beschreibe, wofür sie genutzt werden kann, stelle ein paar ihrer Eigenschaften vor und beschreibe mindestens ein Ereignis. Überlege dir dann einen kleinen Code und füge ihn als Beispiel auf dem Steckbrief ein.
(Nutze ein Textverarbeitungsprogramm und füge Screenshots (z.B. mit SnippingTool) ein.)*

Der MIT App Inventor (<http://appinventor.mit.edu>) wurde ursprünglich von einem Entwicklerteam um Mark Friedman und Hal Abelson bei Google entwickelt und 2012 an das MIT übergeben.

Der MIT App Inventor wird unter der Creative Commons Attribution-ShareAlike 3.0 Unported License veröffentlicht: <https://creativecommons.org/licenses/by-sa/3.0>

Scratch wird von der Lifelong-Kindergarten-Group am MIT-Media-Lab entwickelt. Siehe <http://scratch.mit.edu>.
Scratch ist lizenziert unter CC BY-SA 2.0 (<https://creativecommons.org/licenses/by-sa/2.0/deed.en>).



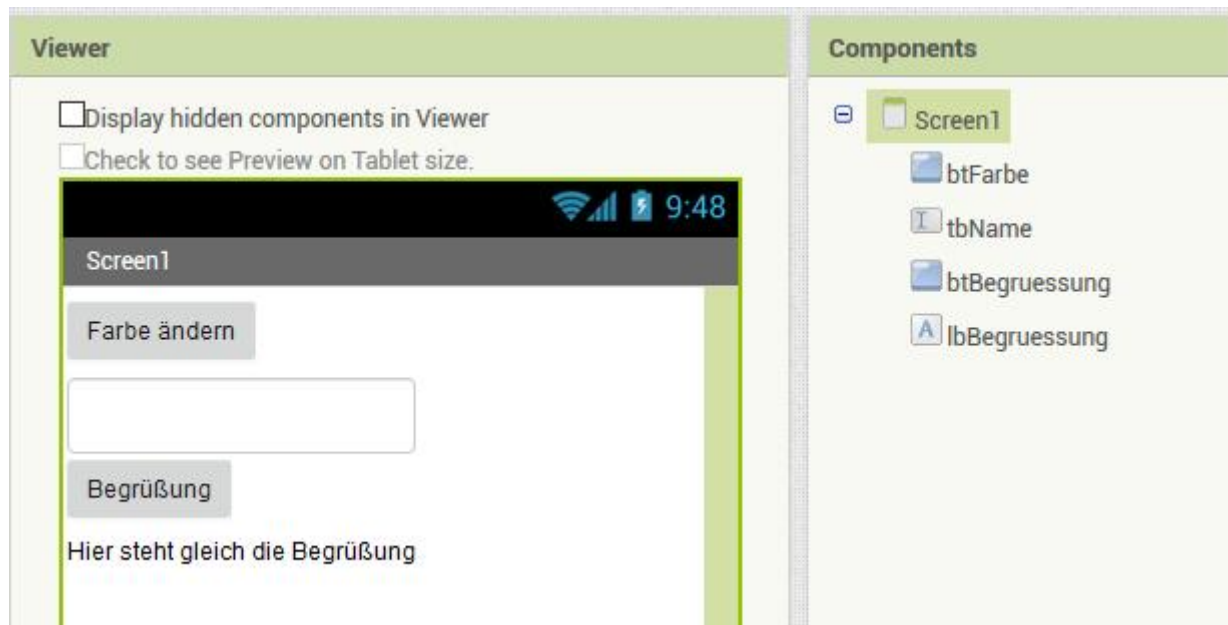
Mögliche Lösungen für die Aufgaben

Aufgabe 1:

```

when btFarbe . Click
do
  set btFarbe . BackgroundColor to #00FF00
  set btFarbe . Text to "Ich habe mich verändert"
    
```

Aufgabe 2:



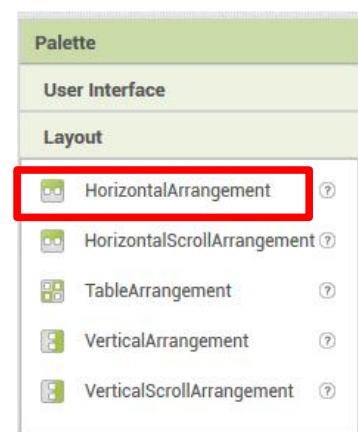
```

when btBegrueassung . Click
do
  set lbBegrueassung . Text to join "Hallo " tbName . Text
    
```

Zeichenketten (Strings) kann man mit dem join-Block aneinanderhängen.

Übrigens kannst du Blöcke löschen, indem du sie auf den Mülleimer rechts unten im Eck im Programmierbereich ziehst.

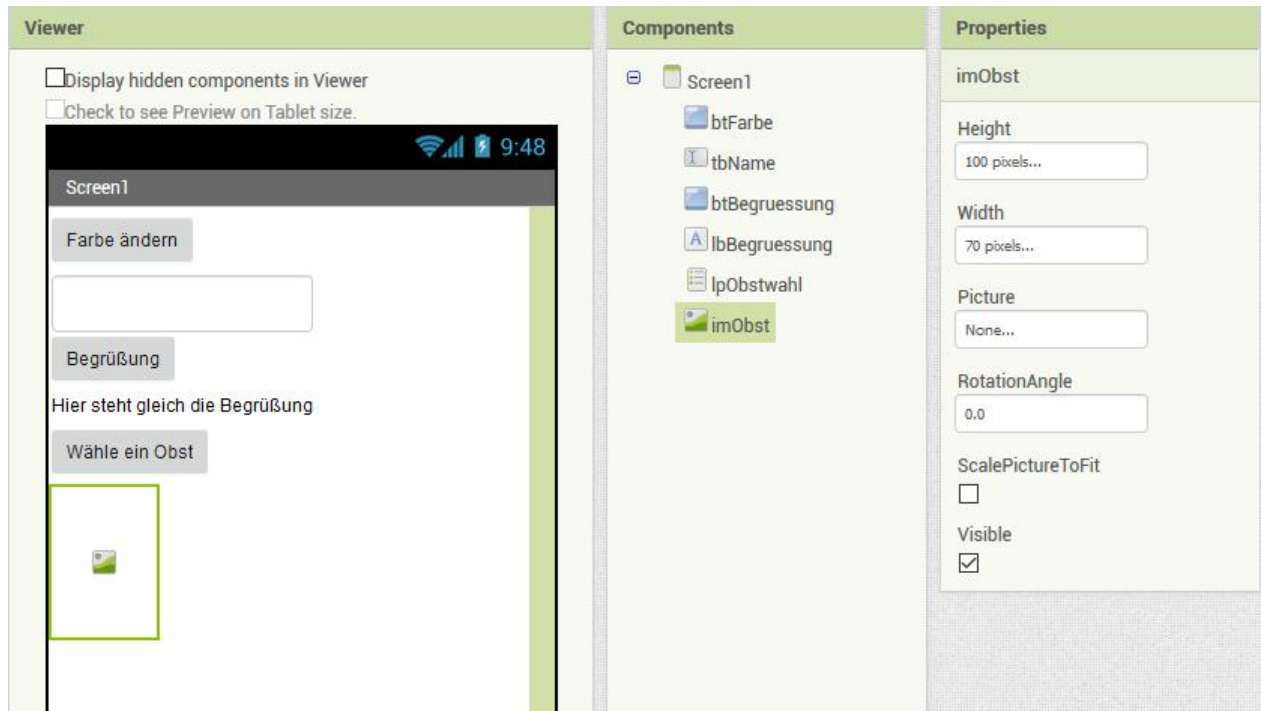
Willst du Komponenten nebeneinander anordnen, so brauchst du ein anderes Layout. Die möglichen Layouts findest du im gleichnamigen Untermenü der Komponentenpalette. Ziehe zunächst das Layout auf den Screen und dann die Komponenten in den Layoutrahmen.



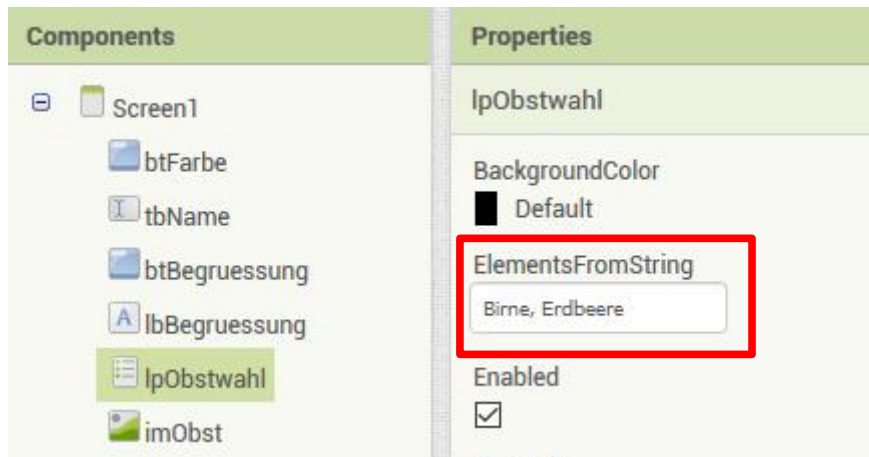


Aufgabe 3:

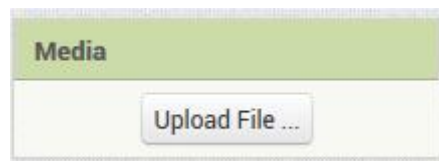
Die beiden Bilder im Tauschordner sind jeweils 70 Pixel breit und 100 Pixel hoch. Stelle deshalb bei den Eigenschaften der Bildkomponente die Höhe (Height) und die Breite (Width) entsprechend ein.



In der nächsten Abbildung siehst du die Eingabe der Elemente der Auswahlliste:

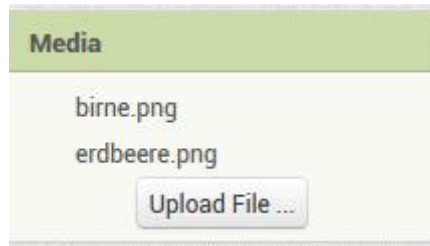


Bilddateien (oder auch Sounddateien) kannst du unter Media hochladen:



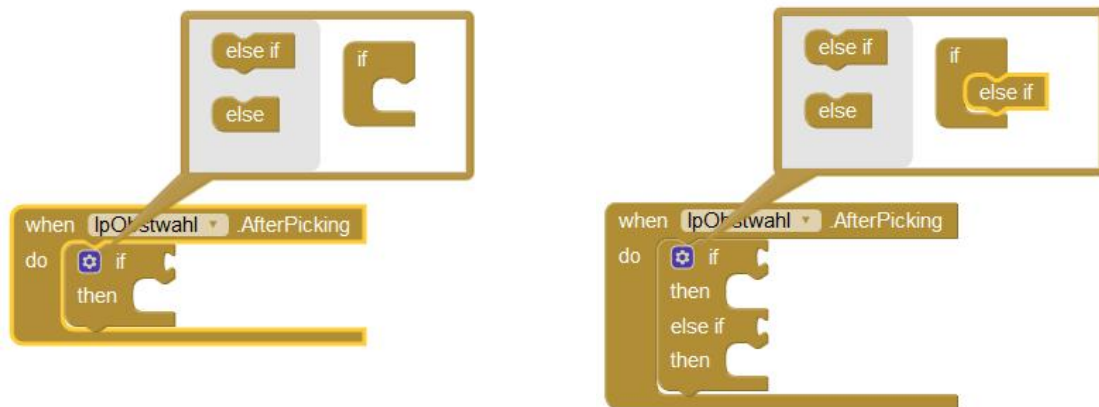


Wenn es geklappt hat, sollte es jetzt so aussehen:

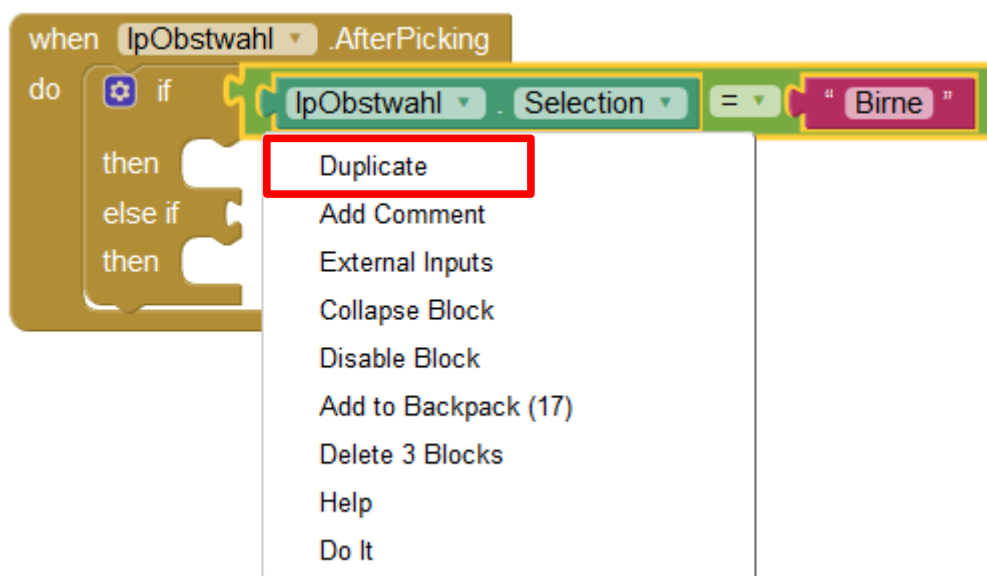


Du hast bestimmt das richtige Ereignis gefunden, das du für die Auswahlliste brauchst: AfterPicking („nachdem ausgewählt wurde“).

Die nötige Verzweigung findest du in der Blockpalette unter Control. Du siehst in der folgenden Abbildung, wie die einfache Verzweigung erweitert werden kann. Nach dem Klick auf das Rädchen, öffnet sich ein Auswahlbereich. Dort kann man mit der Maus die gewünschte(n) Ergänzung(en) nach rechts ziehen.



Die Bedingung in der Verzweigung brauchst du zweimal mit unterschiedlichem Obstanten. Du kannst die Blöcke duplizieren, wenn du mit der rechten Maustaste auf den Block klickst.



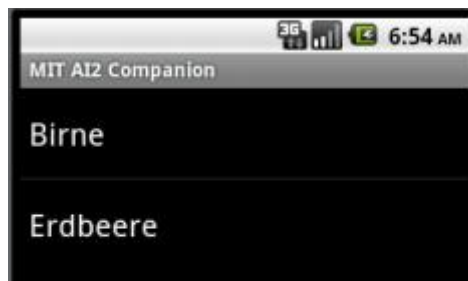


Insgesamt könnte dein Code jetzt so aussehen:

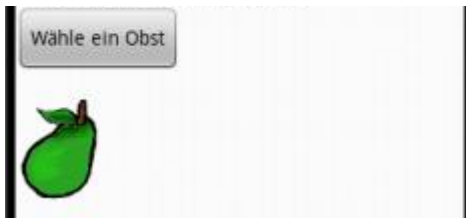
```

when IpObstwahl . AfterPicking
do
  if IpObstwahl . Selection = " Birne "
  then set imObst . Picture to " birne.png "
  else if IpObstwahl . Selection = " Erdbeere "
  then set imObst . Picture to " erdbeere.png "
  
```

Wenn du den Emulator gestartet hast und auf den ListPicker klickst, öffnet sich die Auswahlliste:



Nach der Auswahl siehst du das gewählte Bild in der Bildkomponente:



In der Abbildung rechts siehst du alle drei Aufgaben im Emulator.