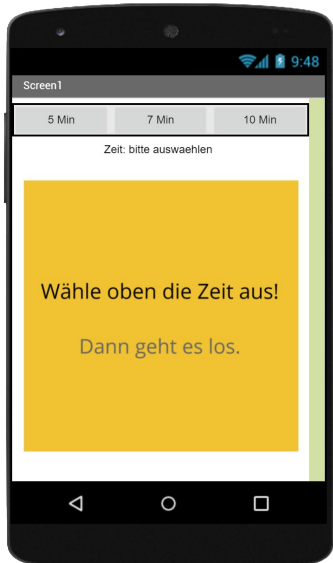




Du brauchst diese Komponenten:

- 1 Label
- 1 Clock

Tipp

Falls ihr das Projekt aus der ersten Sitzung nicht mehr habt, könnt ihr unsere Musterlösung nutzen:

appcamps.link/aia_workout_einfach

Ladet diese herunter und importiert sie im App Inventor unter "Import project (.aia) from my computer ..."

Aufgabe

1. Gebt appinventor.mit.edu in den Browser ein und klickt oben links auf [Create App!](#).
2. Loggt euch ein und öffnet euer Projekt aus der ersten Sitzung über "My Projects".

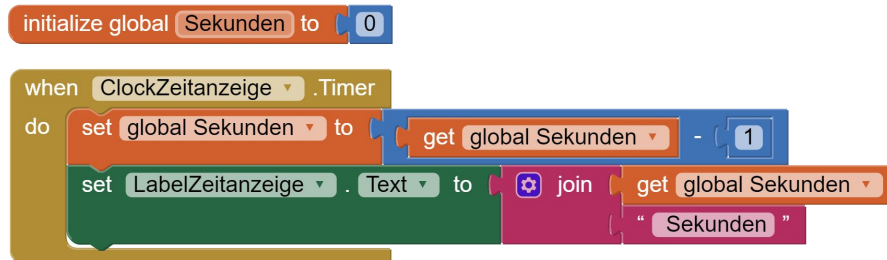
Ihr solltet euch nun im Design befinden.

Designer Blocks

3. Fügt ein Label unter dem Horizontal Arrangement ein und nennt es *LabelZeitanzeige*.
4. Setzt den Text auf "Zeit: bitte auswaehlen".
5. Fügt außerdem eine Uhr *ClockZeitanzeige* hinzu und entfernt den Haken bei *TimerEnabled*.
6. Die Uhr soll einmal pro Sekunde auslösen, um die Sekunden herunter zu zählen. Deshalb muss das *TimerInterval* 1.000 ms (= 1 Sekunde) sein.

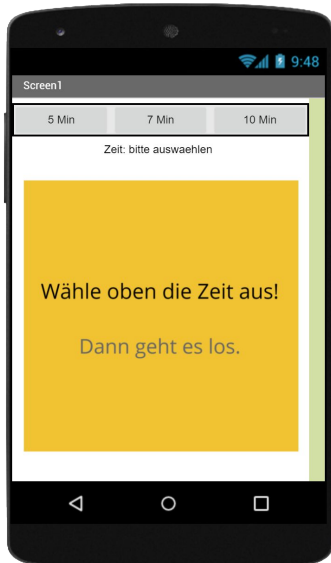
Wechselt in die Programmierung.

7. Erstellt eine Variable *Sekunden*, die sich die noch verbleibenden Sekunden merkt.
8. Initialisiert sie mit 0. Sie wird später auf den richtigen Wert gesetzt.
9. Sobald die *ClockZeitanzeige* abgelaufen ist, zieht 1 von der Variablen *Sekunden* ab.
10. Zeigt außerdem im *LabelZeitanzeige* den Wert der Sekunden und das Wort "Sekunden" an.



Nächste Aufgabe

Da nun schon drei Uhren gestartet und gestoppt werden müssen, erstellt dafür Prozeduren, in denen *TimerEnabled* von allen Uhren auf true bzw. false gesetzt wird.

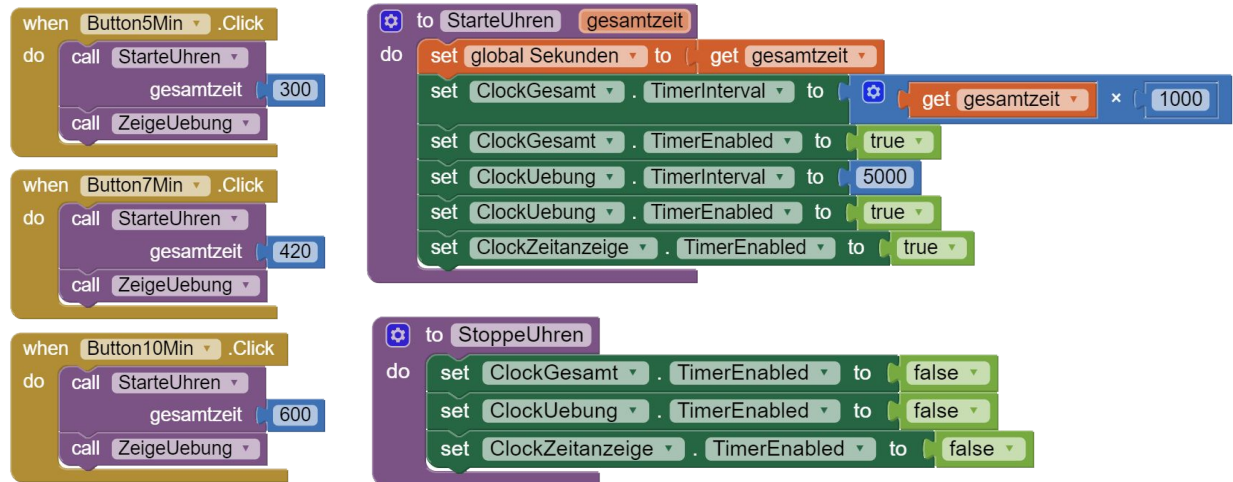
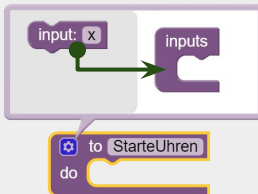


Aufgabe

1. Da nun schon drei Uhren gestartet und gestoppt werden müssen, erstellt zwei Prozeduren.
2. *StoppeUhren* setzt *TimerEnabled* von allen drei Uhren auf false.
3. Die zweite Prozedur *StarteUhren* braucht einen Parameter *gesamtzeit* (siehe Tipp).
4. Die *gesamtzeit* soll in der Variablen *Sekunden* gespeichert werden.
5. Berechnet damit außerdem das *TimerInterval* der *ClockGesamt* in Millisekunden.
6. Setzt das *TimerInterval* der *ClockUebung* hier auf 5000 (statt in *ZeigeUebung*) und setzt bei allen drei Uhren *TimerEnabled* auf true, um sie zu aktivieren.
7. Anstatt bei Klick auf die Buttons die *ClockGesamt* zu aktivieren, muss dort nur noch die Funktion *StarteUhren* mit der jeweiligen Zeit in Sekunden aufgerufen werden.

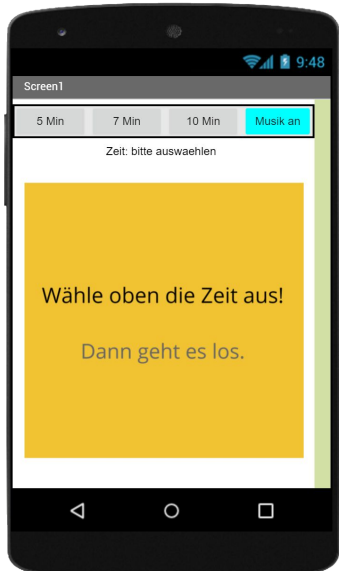
Tipp

Um einen Parameter an die Prozedur übergeben zu können, klickt auf das blaue Zahnrad, zieht *input* rechts unter *inputs* und ändert x zu dem richtigen Namen.



Nächste Aufgabe

Fügt einen weiteren Button ein. Nutzt einen Player, um eine Hintergrundmusik abzuspielen, wenn dieser angeklickt wird. Wird erneut auf den Button geklickt, stoppt die Musik wieder.



Du brauchst diese Komponenten:

1 Player

1 Button

Tipp

Unsere Bilder und Klänge¹ findet ihr hier: appcamps.link/workout

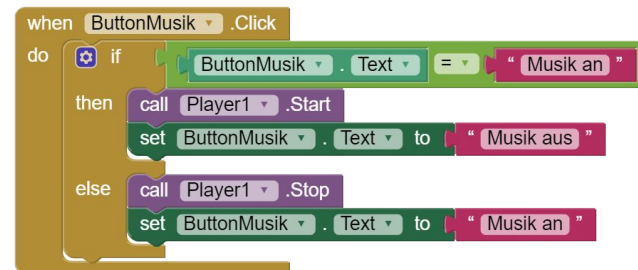
Aufgabe

Wechselt in die Design Ansicht.

1. Ladet die hintergrundmusik.mp3 unter *Media* hoch.
2. Fügt einen Player ein und wählt unter *Source* die hintergrundmusik.mp3 aus.
3. Setzt außerdem einen Haken bei *Loop*, damit die Musik nicht vorzeitig aufhört.
4. Fügt nun einen Button in das Horizontal Arrangement ein und nennt ihn *ButtonMusik*.
5. Ändert den Text auf dem Button zu "Musik an" und gebt ihm eine andere Farbe.

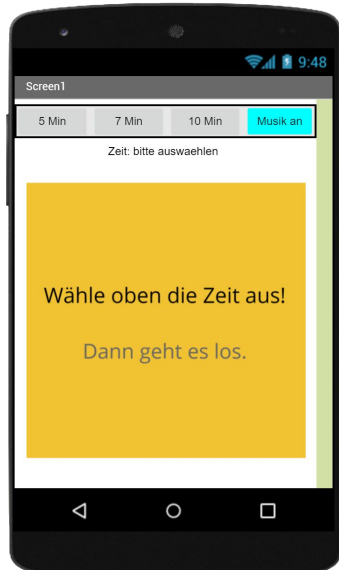
Wechselt in die Programmierung.

6. Wenn der Button geklickt wird, soll entweder die Musik starten oder stoppen. Nutzt den Text, der darauf steht, um zu entscheiden, was passieren soll.
7. Wenn der Text "Musik an" zu sehen ist, startet den *Player1* und ändert den Text auf dem Button zu "Musik aus".
8. Ansonsten stoppt den *Player1* und zeigt den Text "Musik an" auf dem Button an.



Nächste Aufgabe

Wenn die *ClockGesamt* abgelaufen ist, rufe die neue Prozedur *StoppeUhren* auf, statt alle Uhren einzeln zu stoppen. Stoppe außerdem den Player und zeige wieder den ursprünglichen Text im *LabelZeitanzeige* an.



Aufgabe

1. Sobald die *ClockGesamt* abgelaufen ist, wird das Workout beendet.
2. Ruft dafür die neue Prozedur *StoppeUhren* auf, anstatt die Uhren einzeln zu beenden.
3. Stoppt außerdem die Hintergrundmusik, also den *Player1* und setzt den Text auf dem *ButtonMusik* zurück auf "Musik an".
4. Zeigt wie bisher auch das Startbild wieder im *Image1* an.
5. Setzt zuletzt den Text des *LabelZeitanzeige* wieder zurück auf den ursprünglichen Text: "Zeit: bitte auswahlen".

```
when ClockGesamt .Timer
do
  call StoppeUhren
  call Player1 .Stop
  set ButtonMusik . Text to "Musik an"
  set Image1 . Picture to "start.png"
  set LabelZeitanzeige . Text to "Zeit: bitte auswahlen"
```

Schon fertig?

Super! Dann teste deine App wieder und starte ein neues Workout!

Du hast noch viel Zeit übrig? Dann überlege doch mal, wie du die App noch um eigene Funktionen erweitern kannst. Lasse den / die NutzerIn zum Beispiel zwischen verschiedenen Musikarten wählen. Oder teile die Übungen in Kategorien ein, aus denen eine ausgewählt werden kann. Dir fällt bestimmt auch noch etwas ein. Viel Spaß dabei!