Digitale Bildung

Programmieren lernen mit BOB3

Lehrerhandreichung – Sekundarstufe I

Geeignet zur Unterstützung der Kompetenzen im MEDIENPASS NRW

Dieses Dokument steht unter der Creative Commons Namensnennung 4.0 International (CC BY-SA 4.0) Lizenz. Autor: Katja Bach. Herausgeber: www.bob3.org, Stolberg, 2018.
DIGITALE BILDUNG: Programmieren lernen mit BOB3, Lehrerhandreichung

Lehrerhandreichung ‚Programmieren lernen mit BOB3‘

Version 8

Liebe Lehrerinnen und Lehrer,

die folgenden Seiten vermitteln Ihnen eine kompakte Grundlage für die praktische Durchführung des Unterrichts mit BOB3. Zusätzlich werden Details zu den Lernzielen, zu den angestrebten Kompetenzen und zum didaktischen Konzept dargestellt.

Da insbesondere im Bereich der Informationstechnologie eine große Heterogenität in den Vorkenntnissen der SuS vorhanden ist, war die Umsetzung der Wissensvermittlung in Form eines individuellen, differenzierten Unterrichts bei der Entwicklung ein zentrales Thema. Das BOB3-Programmier-Tutorial wurde so ausgelegt, dass es von den SuS selbstbestimmt, interaktiv und in individuellem Lerntempo bearbeitet werden kann. Insbesondere wurde darauf geachtet, dass schwächere Schüler die Motivation bei der Durchführung nicht verlieren, und dass leistungsstarken Schülern anspruchsvolle Aufgaben im Bereich der Vertiefungstutorials geboten werden. Da die Gruppe der SuS vom Tutorial angeleitet und motiviert wird, ergibt sich für die Lehrkraft die Möglichkeit, den gewonnenen Freiraum zur individuellen Förderung einzelner SuS einzusetzen. Weitere Informationen zum Konzept unter: www.bob3.org/de/mint

Bildungsziele:

- Die SuS sollen die Bedeutung des Begriffs ‚Programmierung‘ verstehen und die Tätigkeit des ‚Programmierens‘ selbst durchgeführt haben.
- Die SuS sollen erkennen, welche Möglichkeiten sich durch Programmierung ergeben. Sie sollen die Potentiale Informationtechnischer Systeme erkennen und überlegen wo ihre Grenzen sind.
- Die SuS sollen lernen, wie Informationstechnologie entwickelt wird und welchen Einfluss diese auf die Gesellschaft ausübt.
- Die SuS sollen lernen wie man Strategien zur Problemlösung entwickelt und modelliert und sollen diese dann mittels Algorithmen und passenden digitalen Werkzeugen umsetzen.
- Die SuS sollen verstehen, was Programmcode bedeutet und wie man ihn verwendet. Spezielle Kontrollstrukturen wie bedingte Anweisungen und Schleifen-Konstrukte sollen erlernt werden.
- Leistungsstarke SuS sollen zusätzlich einen vertiefenden Einblick in die Prinzipien der Programmierung bekommen, ohne dabei zu stark auf eine spezielle Syntax, theoretische Konstrukte oder Algorithmen einzugehen - dies soll später systematisch im regulären Informatik-Unterricht geschehen, der dabei dann auf die erworbenen Grundkenntnisse aufbauen kann.
- Das Interesse der SuS an Technik, Mathematik & Informatik soll durch das Projekt geweckt und positiv verknüpft werden.

Technische Voraussetzungen:

- Computerarbeitsplätze mit Windows, Linux oder MacOS Betriebssystem und Internet-Zugang
- 1 ProgBob und 1 BOB3 pro Computerarbeitsplatz
- Installiertes BobDude Übertragungsprogramm

Hinweis: Bei der Durchführung in der Sekundarstufe I über 8 UE sollten die Lernziele von Intro I, Intro II und Intro III von den meisten SuS erreicht werden.

Einführung: [1 UE]

Voraussetzungen:

- SuS sollten rudimentäre Kenntnisse im Umgang mit einer Computer-Maus, einer Tastatur und in der Bedienung eines Web-Browsers haben.

Lernziele:

- SuS sollen sich einen Account in der Web-Oberfläche anlegen und die Zugangsdaten notieren
- SuS sollen wissen unter welcher URL das Tutorial zu finden ist
- SuS sollen BOB3 kennen lernen

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Kenntnisse in der Bedienung eines Computers vertiefen

Konzeptbezogene Kompetenzen: Digitale Umgebungen und Werkzeuge zum persönlichen Gebrauch anpassen

Soziale Kompetenzen: gegenseitige Hilfestellung bei der Bedienung des Computers

Vorbereitungen für die erste Unterrichtseinheit:

Benötigt werden:

- Ein Computer pro Schüler (alternativ ein Computer pro zwei SuS)
- Internetzugang + Webbrowser

- Ein Programmier-Helm „ProgBob“ pro Computer
- Ein Roboter „BOB3“ pro Computer

Schritt 1:

Zunächst wird auf jedem Computer das Programm „**BobDude**“ installiert.
Das Programm steht hier für alle Betriebssysteme (Windows, Mac OS X, Linux) kostenfrei zur Verfügung:

http://dude.bob3.org

Schritt 2:

Nach der Installation kann die LK einen **ersten Test** durchführen. Dazu wird der Programmier-Helm auf den BOB3 aufgesteckt und per USB-Kabel an den Computer angeschlossen.
Bei dem Windows Betriebssystem wird nun automatisch der Gerätetreiber installiert, dies kann einige Minuten dauern. Sobald die LEDs am Helm **grün leuchten** ist der BOB3 einsatzbereit!

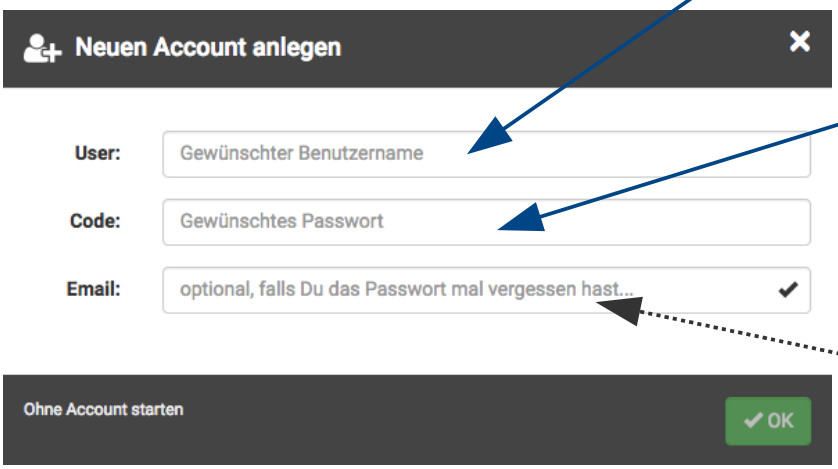
Erste Unterrichtseinheit:

→ Erstellen der Benutzer-Accounts:

Die SuS starten den Webbrowser und gehen auf die Seite **http://www.ProgBob.org**:

Per Klick auf den Button „**Neuer Benutzer**“ öffnet sich der Dialog „*Neuen Account anlegen*“:

Die SuS denken sich jeweils einen eigenen **Benutzernamen** und ein **Passwort** aus.



Tip: Die SuS schreiben sich die Benutzerdaten auf.

Benutzername:

Beispiel: **LuckyLuke**, **BieneMaja** etc.
(mindestens 4 Zeichen)

Passwort:

Beispiel: **Geheimnis13, Schueler27** etc.
(mindestens 6 Zeichen)

Email [optional]:

Die Angabe einer Email-Adresse ist **optional** möglich: Falls man sein Passwort vergessen hat, kann es per Email angefordert werden.

Tutorial ‚Intro I': [1-3 UE]

Voraussetzungen:

- SuS sollten die Lernziele aus der Einführung erreicht haben

Lernziele:

- SuS sollen die Web-Oberfläche der ProgBob Tutorials kennen lernen und bedienen können
- SuS sollen Programm-Quelltext selbstständig compilieren können und auf den Mikrocontroller übertragen
- SuS sollen zunächst vorgegebene kleine Änderungen am Quelltext vornehmen
- SuS sollen den Funktionsaufruf zur Veränderung der Farbe der LEDs kennen lernen und verstehen
- SuS sollen die delay() Funktion kennen lernen um Blinkeffekte mit den LEDs zu realisieren.
- SuS sollen jetzt selbstständig Änderungen am Quelltext vornehmen (Farbe der LEDs frei verändern, Parameter der delay()-Funktion, eigene Blinkroutinen)

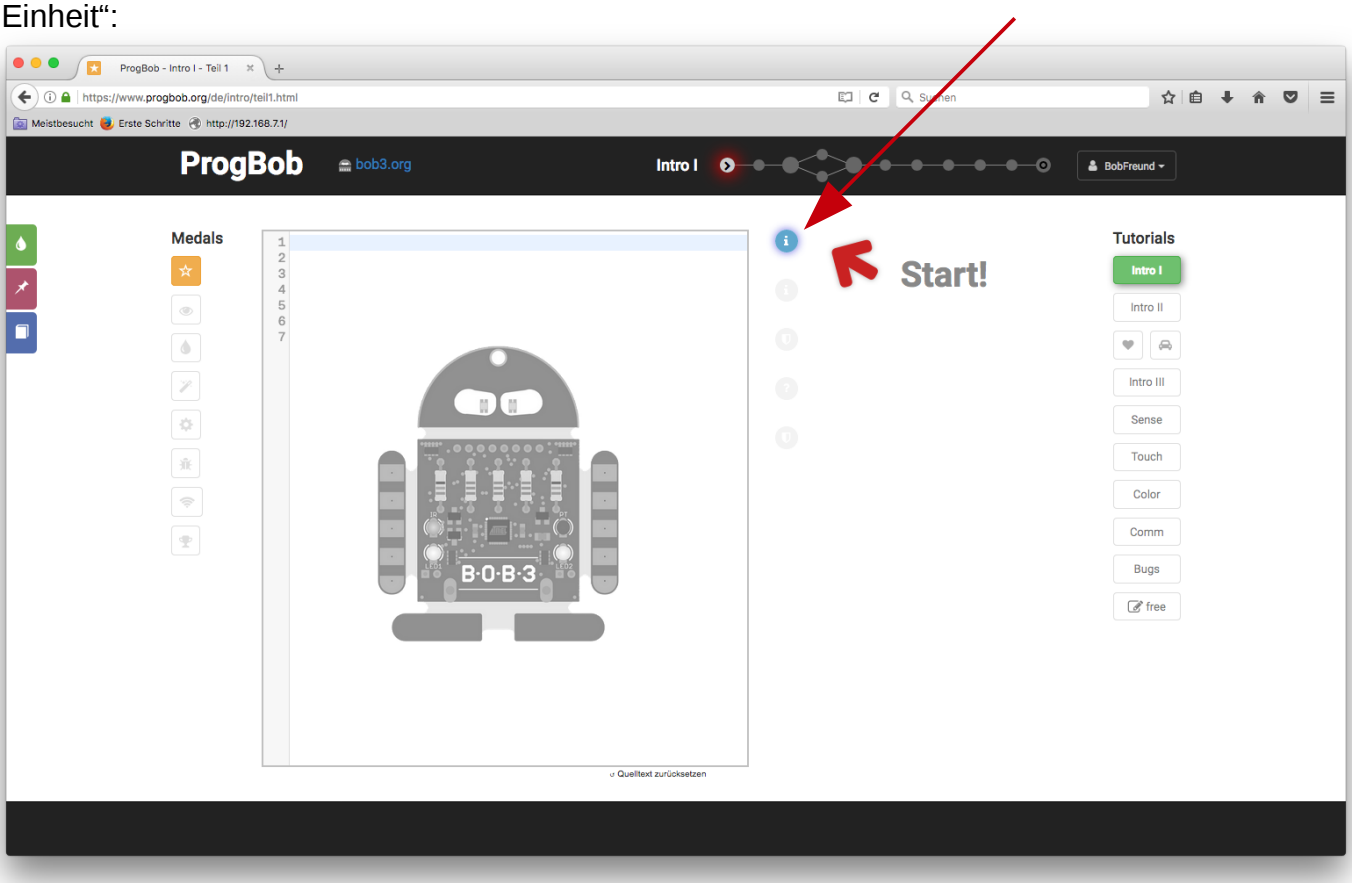
Prozessbezogene Kompetenzen: Bedienung der Oberfläche; compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programmtext durchführen; Vertiefung der Kenntnisse in der Tastaturbedienung (Sondertasten, Groß-Kleinschreibung)

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems planen und verwenden

Soziale Kompetenzen: gemeinsame Theoriebildung über die Funktionsweise des Roboters; gegenseitige Hilfestellung bei der Bedienung des Computers

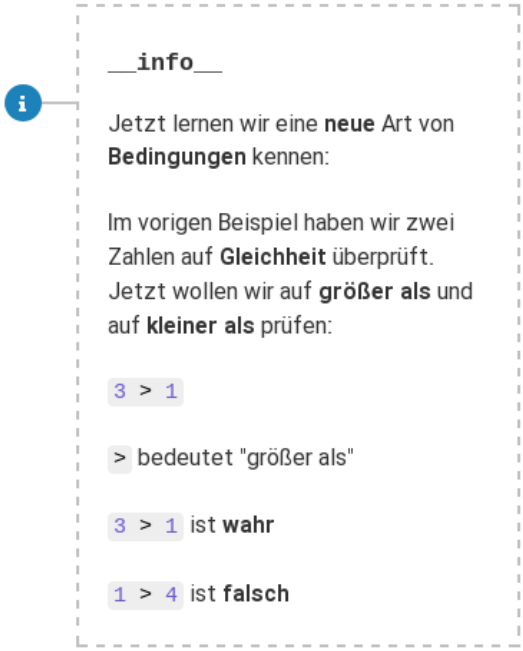
Starten des Tutorials:

Die SuS bestätigen den Dialog „Neuen Account anlegen" mit „OK" und gelangen so direkt zum Hauptfenster des Tutorials. Sie starten das Tutorial per Klick auf die erste „Info-Einheit":



Das Tutorial ist **interaktiv** aufgebaut und **selbsterklärend**, so dass die SuS nun selbstständig und im jeweils eigenen Tempo arbeiten können.

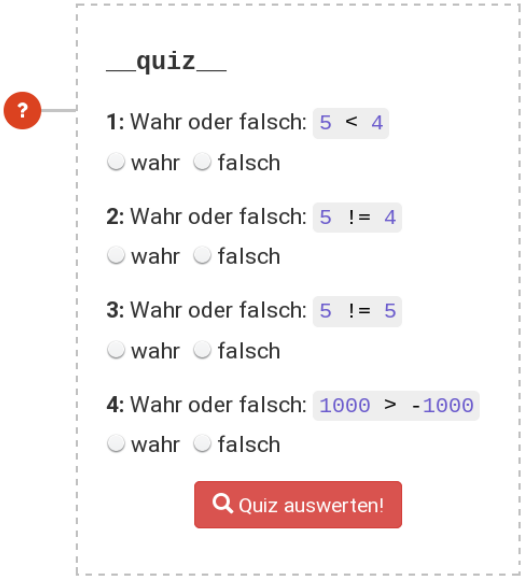
Die blauen „Info-Einheiten“ bieten Lerninhalte, z.B.:



Die grauen „Aufgaben-Einheiten“ bieten Aufgaben, die von den SuS bewältigt werden müssen, z.B.:

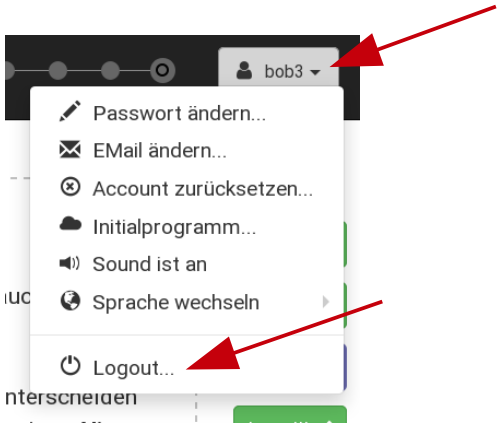
Die roten „Quiz-Einheiten“ dienen zur Wissensüberprüfung und müssen beantwortet werden, z.B.:

Per Klick auf den Button „Quiz auswerten!“ bekommen die SuS direktes Feedback zu ihrer Lösung.

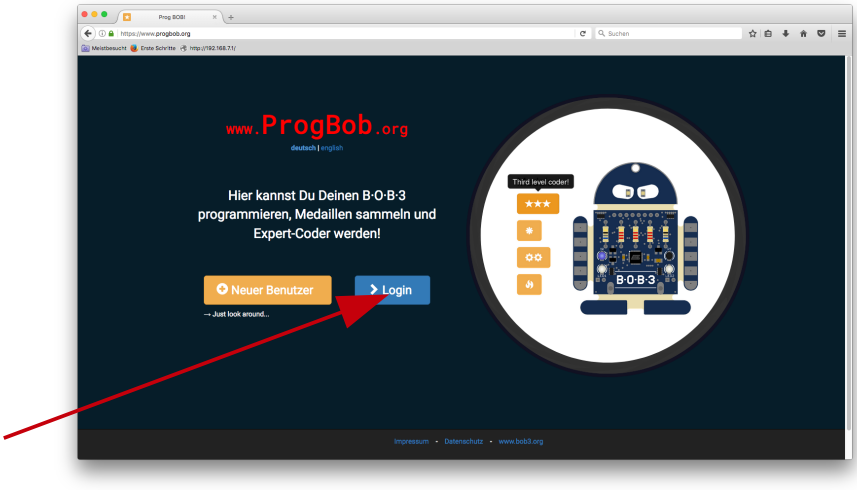


Ende einer Unterrichtseinheit:

Am Ende der Unterrichtseinheit klicken die SuS rechts oben auf den jeweiligen Benutzernamen und dann auf „**Logout**“.



Der **Lernstand** des einzelnen Schülers ist nun **gespeichert** und die SuS können in der nächsten Unterrichtseinheit über den Button „**Login**“ weiterarbeiten:



Tutorial ‚Intro II‘: [1-3 UE]

Intro II ✔

Voraussetzungen:
- SuS sollten die Lernziele aus dem Intro I Tutorial erarbeitet haben

Lernziele:
- SuS lernen Bedingungen wie 'gleich', 'ungleich' oder 'kleiner als' im Kontext einer Programmiersprache kennen
- SuS setzen die Bedingungen mit dem 'if/else' Konstrukt der Programmiersprache ein und experimentieren damit
- SuS lernen das Konzept einer (Ganzzahl-) Variablen kennen
- SuS speichern den Wert eines Sensors in einer Variablen und werten diese mit dem 'if/else' Konstrukt aus
- SuS lernen den IR-Reflex-Sensor und den Temperatur-Sensor kennen
- SuS erlernen den Sinn und Zweck einer 'for'-Schleife zur wiederholten Durchführung
- SuS lernen wie die Parameter der 'for'-Schleife eingesetzt werden können
- SuS lernen wie man Fehler im Programm-Code lokalisieren kann und wie man diese behebt

Prozessbezogene Kompetenzen: Suchen und Beheben von Fehlern im Programm

Konzeptbezogene Kompetenzen: Funktionsweise von analogen Sensoren, Bedingungen; Konzept von Verzweigungen, Ganzzahl-Variablen, Schleifen, Technische Probleme identifizieren; Probleme formalisiert beschreiben, Problemlösestrategien entwickeln und dazu eine strukturierte algorithmische Sequenz planen, diese auch durch Programmieren umsetzen und die gefundene Lösungsstrategie beurteilen

Soziale Kompetenzen: Gegenseitige Unterstützung bei der Suche nach Fehlern im Programmcode.

Zur Motivation der SuS werden im Anschluss die beiden optionalen Kurztutorials „Freundschaftstester“ und „Polizeiblinklicht“ angeboten.

Tutorial ‚Intro III‚: [1-3 UE]

Intro III ✔

Voraussetzungen:
- SuS sollten die Lernziele aus dem Intro II Tutorial erarbeitet haben

Lernziele:
- SuS lernen die Touch-Sensoren als digitale Eingabeelemente kennen
- SuS erlernen die grundlegende Verwendung des 'switch/case'-Konstrukts zur Fallunterscheidung
- SuS erfahren wie eine selbstdefinierte Funktion angelegt werden kann und wie man diese aufruft
- SuS experimentieren mit Funktions-aufrufen zur Vereinfachung von wiederholten Abläufen

Prozessbezogene Kompetenzen: Verringerung des Programmieraufwands durch Verwendung von selbst-definierten Funktionen

Konzeptbezogene Kompetenzen: Funktionsweise von digitalen Sensoren; Konzept von Verzweigungen; Funktionsaufrufe; eine strukturierte algorithmische Sequenz zur Lösung eines Problems planen und verwenden

Vertiefungstutorials: [4-8 UE]

Voraussetzungen:

- SuS sollten die Lernziele aus dem Intro III Tutorial erarbeitet haben

Lernziele:

- SuS lernen die verschiedenen Sensoren des Roboters im Detail kennen
- SuS experimentieren mit Arrays
- SuS lernen Funktionen mit Rückgabewerten kennen
- SuS lernen das Grundprinzip der Ereignisbehandlung anhand der Touch-Sensoren kennen
- SuS lernen die Grundlagen des RGB Farbmodells kennen
- SuS lernen die Grundlagen des HSV Farbmodells kennen
- SuS lernen die Grundlagen der Informationsübertragung zur Kommunikation zwischen technischen Geräten kennen

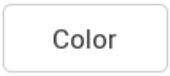
Konzeptbezogene Kompetenzen: Funktionsweise von Arrays; Farbmodelle zum Mischen von Farben; Funktionen mit Rückgabewerten; Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen, nachvollziehen und reflektieren

Soziale Kompetenzen: gemeinsames Arbeiten an einem Programm zur Übertragung von Informationen

Zusätzlich zu den einführenden Tutorials werden vertiefende Tutorials angeboten, die insbesondere die Experten der Klasse beschäftigen können. In diesen Tutorials werden einzelne Sensoren und Fähigkeiten des Roboters tiefergehend behandelt.

Verstehen und Arbeiten mit dem IR-Sensor des Roboters, Experimente mit verschiedenen Lichtarten.

Verstehen und Arbeiten mit dem Multifeld-Touch-Sensoren des Roboters, Lernen wie die Arme angesteuert und abgefragt werden können.



Erlernen des RGB- und des CMYK-Farbmodells. Wie entstehen eigentlich Farben? Experimente mit Farbüberblendeffekten. Verstehen und Anwenden anhand der beiden Multicolour-LEDs des Roboters.

Erlernen der drahtlosen Kommunikation zwischen zwei Roboter-Einheiten. Datenübertragung begreifbar machen anhand des „Alice und Bob“-Modells. Experimente mit Code-Übertragung. Verstehen und Anwenden anhand der IR-Sensorik des Roboters.

Abschlusseinheit: [1 UE]

Voraussetzungen:

- SuS sollten im Rahmen ihrer Entwicklungsstufe möglichst viele Tutorials bearbeitet haben

Lernziele:

- SuS sollen die Parallelen von BOB3 zu anderen, alltäglichen Embedded Systemen erkennen und diskutieren
- SuS sollen sich Gedanken über die technischen Möglichkeiten und Einschränkungen von Computerprogrammen machen und diese gemeinsam reflektieren

Konzeptbezogene Kompetenzen: Einflüsse von Algorithmen und Auswirkung der Automatisierung von Prozessen in der digitalen Welt beschreiben und reflektieren

Soziale Kompetenzen: gemeinsames Diskutieren der Ergebnisse

Am Anfang dieser Einheit sollen die SuS zunächst reflektieren, welche neuen Erkenntnisse sie durch das Programmiertutorial erworben haben und diese Erkenntnisse im Klassenverband diskutieren:

- Wie kommt das Programm auf den Roboter? [Compiler→BobDude→ProgBob→BOB3]
- Was ist eine Programmsequenz? [Die geplante Abfolge von einzelnen Anweisungen]
- Was ist ein Verzweigung? [if/else – in Abhängigkeit von einer Bedingung wird ein bestimmter Programmzweig ausgeführt]
- Was ist eine Schleife? [for – eine bestimmte Programmsequenz wird mehrfach ausgeführt]

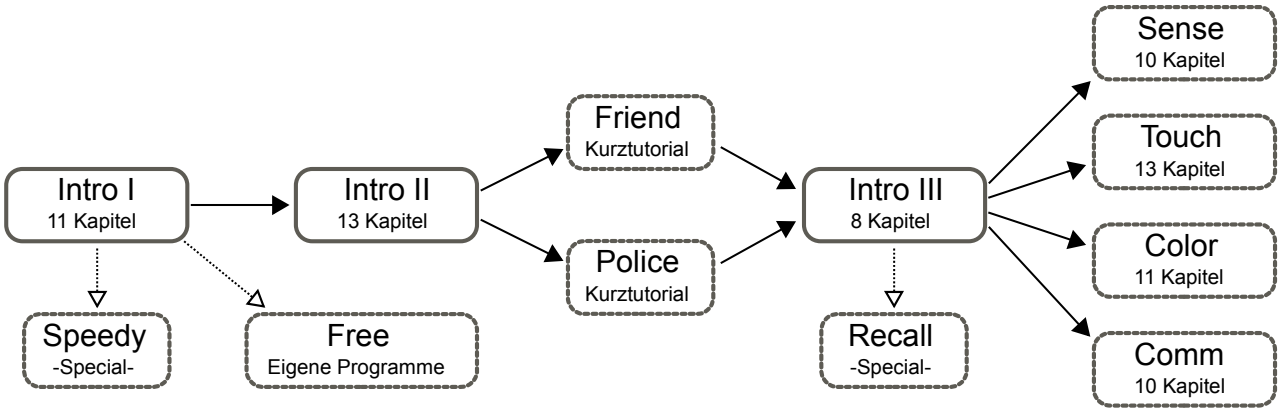
Anschließend sollen die SuS überlegen, in welchen alltäglichen Geräten heutzutage Computer-Systeme eingebaut sind, die man nicht direkt erkennt. Dies sind zum Beispiel: Elektrische Küchengeräte, Waschmaschinen, Fernseher, Kühlschränke, digitale Fieberthermometer, Fensterheber in der Autotür, Spielzeuge und vieles mehr. Dabei sollen die SuS auch erkennen, dass für all diese Systeme spezielle Softwareprogramme geschrieben wurden:

- Diskutiert was passieren würde, wenn man den Programmcode einer Waschmaschine fälschlicherweise auf den Mikrocontroller einer Spülmaschine übertragen würde.
- Überlegt in welchen alltäglichen Geräten ähnliche Mikrocontroller-Systeme eingesetzt werden.
- Überlegt welche Sensoren diese Geräte haben und erstellt daraus eine Liste. Vergleicht eure Listen in der Klasse.
- Überlege diesmal, welche Aktoren diese Geräte haben, erstellt daraus eine Liste und vergleicht diese in der Klasse.

Jetzt sollen die SuS überlegen, welche Parallelen es zwischen Embedded-Systemen wie dem BOB3 und Computern allgemein gibt, und anschließend reflektieren welche Möglichkeiten und Einschränkungen sie daraus ableiten können:

- Überlegt mit euren Erfahrungen, was BOB3 gut kann und was er nicht so gut kann.
- Vergleicht die Fähigkeiten und Einschränkungen von BOB3 mit normalen Computerprogrammen – was können diese gut und was können sie nicht so gut.

Lerninhalte / Überblick Programmiertutorial:



Das Programmier-Tutorial ist in verschiedene Sub-Tutorials unterteilt, die **Grundlagen-Tutorials** (*Intro I, Intro II* und *Intro III*) sollen von allen Lernenden bearbeitet werden, die **Vertiefungs-Tutorials** (*Sense, Touch, Color* und *Comm*) sind zur inneren Differenzierung für leistungsstarke Schüler konzipiert. Im Folgenden wird ein Überblick über die Inhalte gegeben:

Intro I
11 Kapitel

Prinzip der Programmierung eines Embedded-Systems verstehen und anwenden
Begriffsklärungen (Quellcode, Compilieren, Programm)
Bedeutung der Programmier-Syntax erlernen
Programmier-Oberfläche kennenlernen und verwenden
Unterschiede und Verwendung der Funktionen setup() / loop() verstehen
Roboter-LEDs ansteuern und Farben der LEDs verändern
Ein einfaches Blinken als Sequenz mit der delay() Funktion erzeugen und Varianten entwickeln

Intro II
13 Kapitel

Wahrheitswerte verstehen und anwenden (true / false)
Arbeiten mit mathematischen Bedingungen (==, >, <, !=,...)
Erlernen von Verzweigungen anhand der Programmstruktur „if / else“
„if / else“ konkret zur Auswertung des IR-Sensors einsetzen
Variablen kennenlernen und einsetzen
Prinzip und Anwendung von for-Schleifen kennenlernen - Anwenden des zuvor Erlernten:

- → Ein-/Ausschalten der LEDs
- → delay() verwenden
- → Variablen
- → Bedingungsprüfungen
- → einfache Fehlersuche (Klammersetzung, Semikolon, ...)

Intro III
8 Kapitel

Sensoren entdecken, ansteuern und auswerten
switch / case Abfragen erlernen und anwenden
Funktionen mit und ohne Rückgabewert unterscheiden können
Verwendung von Funktionen mit Rückgabewert mittels Sensorik
Eigene Funktionen definieren und deren Potential verstehen

Sense 10 Kapitel	Verschiedene Experimente zur Analyse der Messergebnisse der IR-Sensorik
Touch 13 Kapitel	Ereignisbehandlung durch Auswertung der Multifeld-Touch-Sensoren
Color 11 Kapitel	Farbentheorie anhand der beiden Farbmodelle RGB und HSV, konkrete Experimente mit den Multicolor LEDs wie z.B. Farbüberblendeffekte
Comm 10 Kapitel	Kommunikation und Datenübertragung zwischen mehreren Roboter-Einheiten

Kursverwaltung mit Lernstandskontrolle

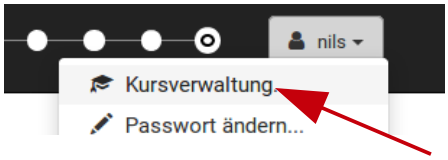
- optional einsetzbar -

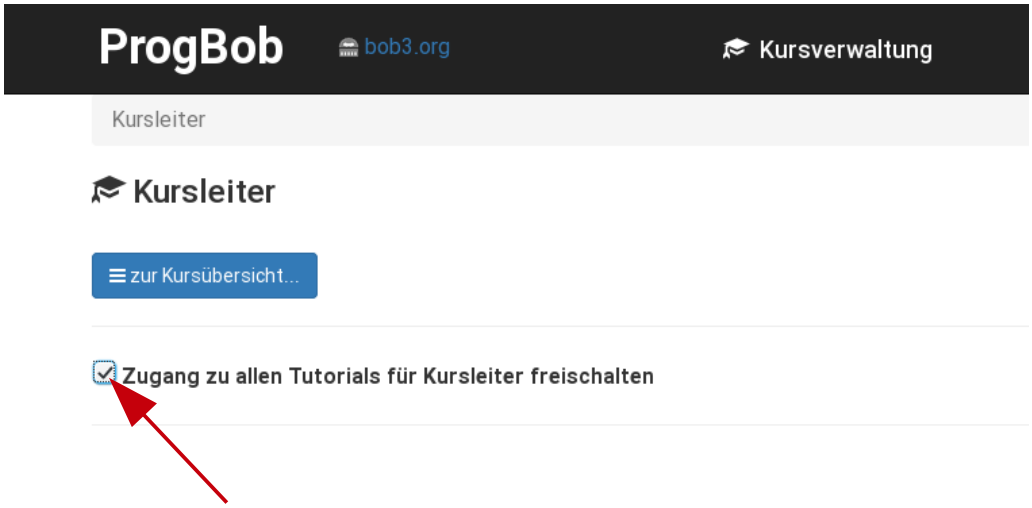
Die ProgBob Lernumgebung bietet speziell für Lehrerinnen und Lehrer eine Kursverwaltung an. Interessierte Lehrer können auf Anfrage ihren Benutzer-Account auf 'Kursleiter' upgraden, dazu bitte eine kurze E-Mail an teacher@bob3.org senden.

Kursleitern stehen die folgenden Funktionen zur Verfügung:

- Zugang zu allen Tutorials für den Kursleiter freischalten
- Teilnehmer können ohne Angabe der persönlichen Daten im Block angelegt werden
- Passwörter der Teilnehmer können zurückgesetzt werden
- Übersicht über den Lernfortschritt der Teilnehmer
- Übersicht über die Zugangsdaten zum Ausdrucken und Verteilen vor dem ersten Termin

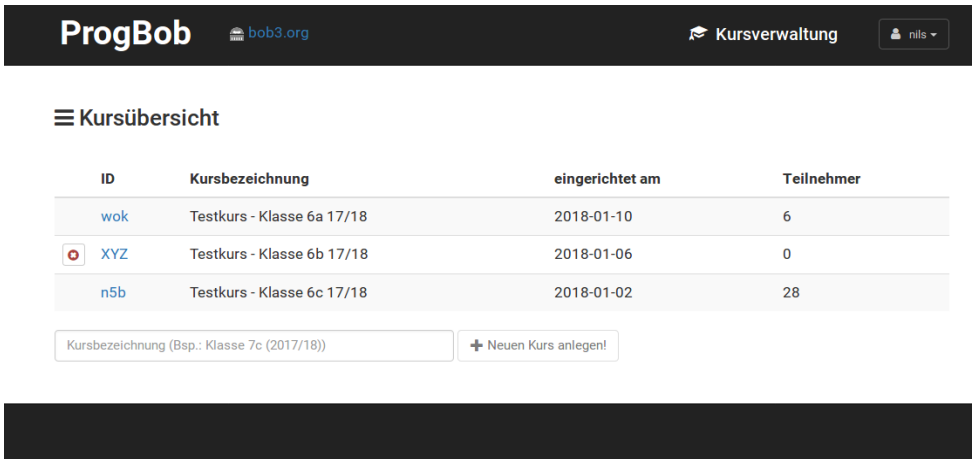
Als Kursleiter bekommt man im Benutzermenü zusätzlich den Menüpunkt „**Kursverwaltung...**“ angezeigt.



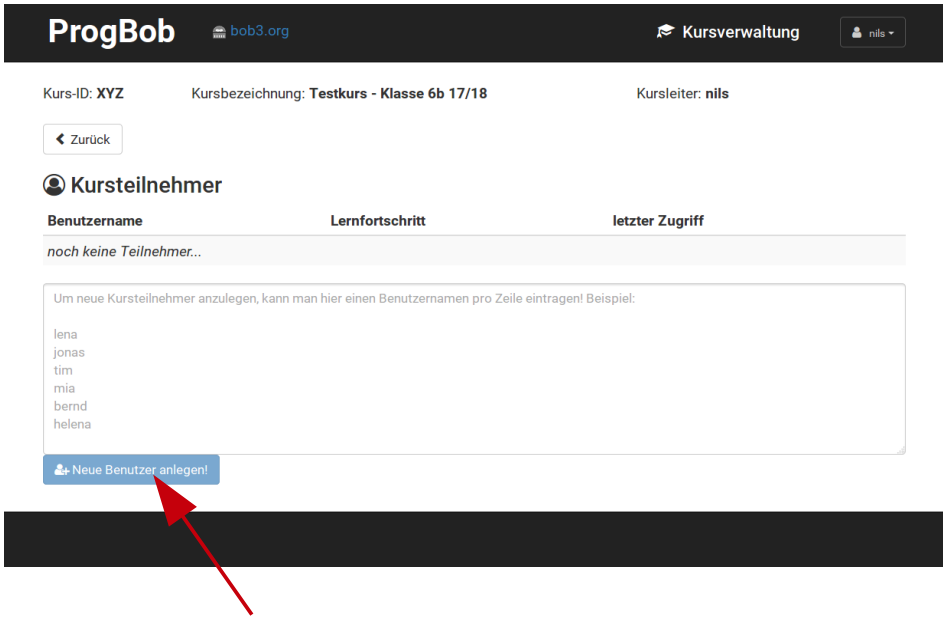


Der **Kursleiter** hat die Möglichkeit, **alle Tutorials** für den eigenen Account **freizuschalten**.

Über den Button „**Zur Kursübersicht**“ gelangt man zur **Übersicht** über die bereits **angelegten Kurse**. Zusätzlich können dort **neue Kurse** angelegt werden. Solange ein Kurs noch keine Teilnehmer hat, kann er dort auch wieder gelöscht werden.



Durch Klick auf die **Kurs-ID** (z.B. **XYZ**) kommt man zum Übersichtsbildschirm über die **Kursteilnehmer**. Ein neu angelegter Kurs hat noch keine Teilnehmer:

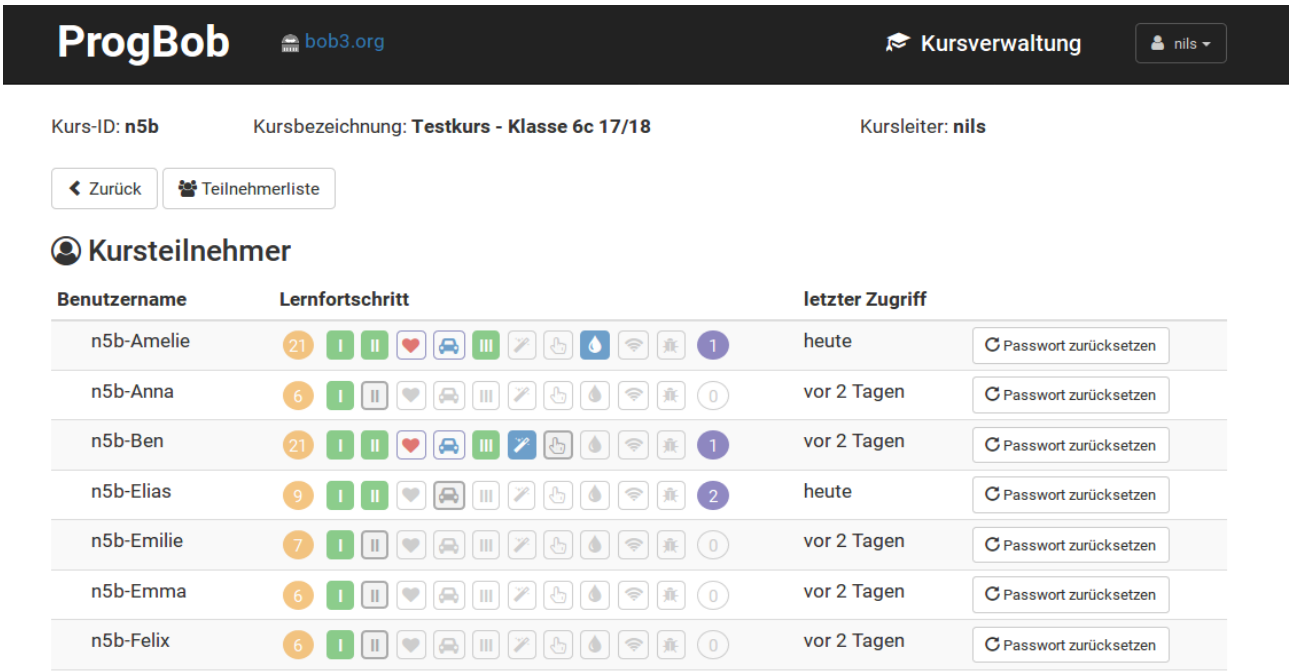


In dieser Übersicht können **neue Kursteilnehmer** angelegt werden.

- Pro Zeile muss ein Benutzername eingegeben werden.
- Neu angelegte Teilnehmer bekommen automatisch erzeugte, zufällige Passwörter. Diese können durch die Teilnehmer später beliebig geändert werden.
- Beim Anlegen der Benutzer wird den Benutzernamen automatisch das Kurskürzel vorangestellt: Lena → XYZ-Lena
- Kursteilnehmer, die sich noch nie eingeloggt haben, können auch wieder gelöscht werden.

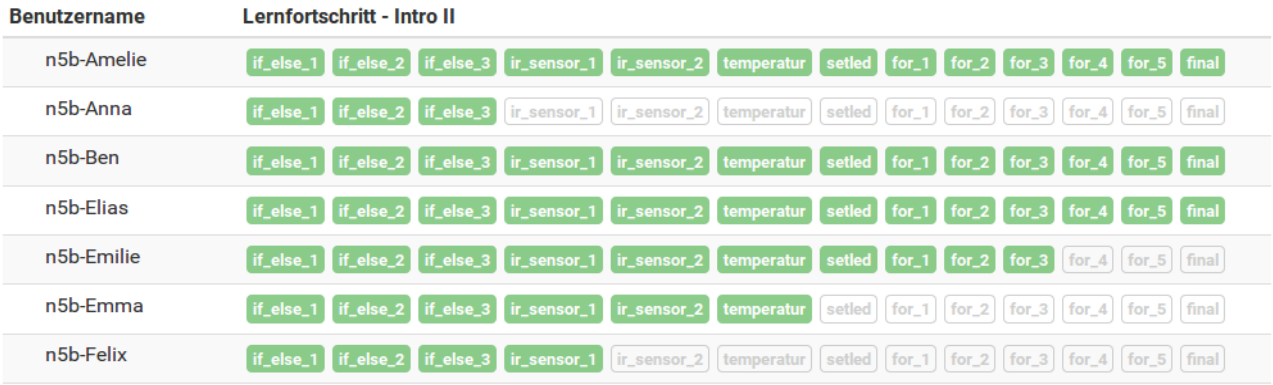
Bei Bedarf können hier die **Passwörter** der einzelnen Schüler **zurückgesetzt** werden!

Sobald die **Schüler** die Aufgaben bearbeiten, werden in der Übersicht die Anzahl der erworbenen Medaillen und die **abgeschlossenen Tutorials** angezeigt.



In diesem Beispiel hat Amelie sich insgesamt 21 Medaillen erarbeitet. Sie hat die Tutorials „Intro I“ und „Intro II“, die beiden Specials „Friend-Detection“ und „Police-Lights“, das „Intro III“ Tutorial und das „Color“ Tutorial fertig bearbeitet und ein eigenes Programm angelegt:

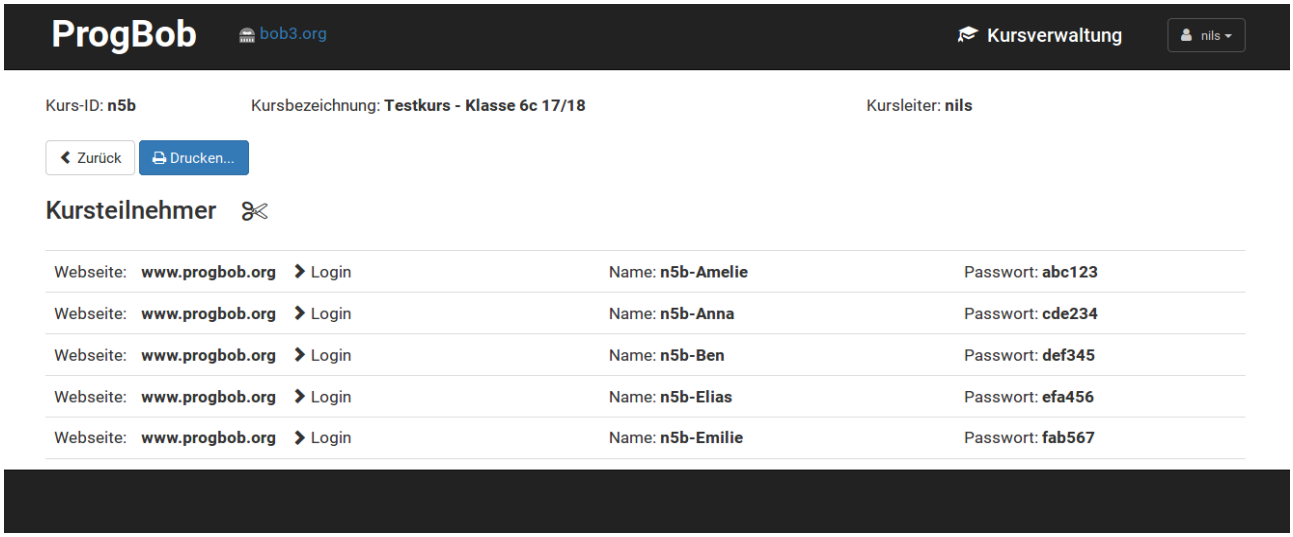
Durch Klick auf ein Tutorial-Symbol werden die einzelnen Lernstände der Schüler im jeweiligen Tutorial angezeigt:



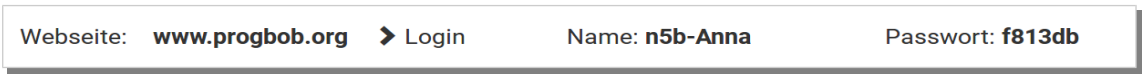
In diesem Beispiel hat Emma 3 Lektionen zum Lernziel „if-else“, 2 Lektionen zum „IR-Sensor“ und eine Lektion zum „Temperatursensor“ fertig bearbeitet. Die Lektionen zu den Lernzielen „for-Schleife“ und „setLed()-Funktion“ muss sie noch bearbeiten.

Zugangsdaten der Teilnehmer

Durch Klick auf den Button „**Teilnehmerliste**“ kommt man zum Übersichtsbildschirm über die **Zugangsdaten**:



Die Liste lässt sich ausdrucken und in Streifen schneiden. Diese können dann an die einzelnen Kursteilnehmer verteilt werden. Hier als Beispiel die Anmeldedaten für Anna:



Arbeitsblatt – Anlegen eines Benutzeraccounts

Aufgabe 1: Starte den Webbrowser auf deinem Computer

Aufgabe 2: Gehe auf die Webseite **www.progbob.org**

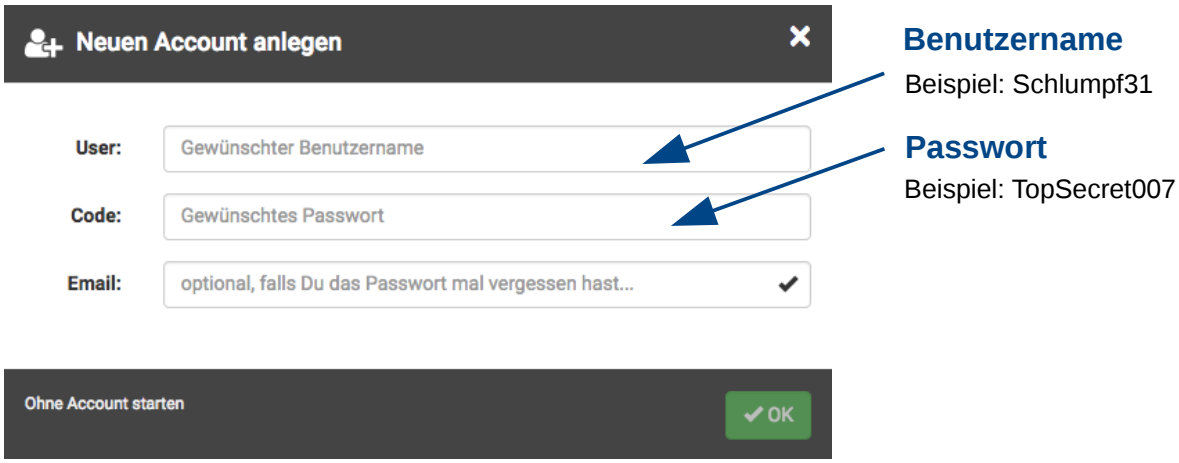
Aufgabe 3: Klicke dort auf den Button **„Neuer Benutzer“**

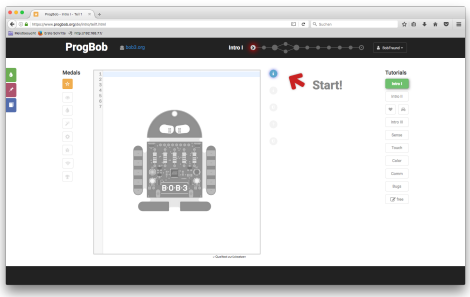
Aufgabe 4: Denke dir einen **Benutzernamen** mit mindestens 4 Buchstaben aus, tippe ihn ein und schreibe ihn zusätzlich hier auf:

_ _ _ _ _ _ _ _ _ _ _ _

Aufgabe 5: Denke dir ein geheimes **Passwort** mit mindestens 6 Buchstaben aus, tippe es ein und notiere es dir hier:

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _



Falls du eine E-Mail Adresse hast, kannst du sie in das Feld „Email“ eintragen, falls nicht ist es auch nicht schlimm ;-)

Aufgabe 6: Wenn du alles ausgefüllt hast, dann klicke auf „**OK**“
→ das Tutorial startet jetzt!