

Unterrichtsverlaufsplan

Phase	Unterrichtsstruktur (mit Zeitplanung)	Lehrerhandeln	Schülerhandeln	Lernziele (fachliche und überfachliche)
Unterrichtseinstieg	<ul style="list-style-type: none"> – Kahoot (10 min) 	<ul style="list-style-type: none"> – Einstieg in Entwicklungsumgebung – Quellcode erklären – „Hier Seite, alle anmelden“ 	<ul style="list-style-type: none"> – Fragen beantworten 	<ul style="list-style-type: none"> – Motivation zur Nutzung einer mächtigeren Entwicklungsumgebung
Erarbeitung	<ul style="list-style-type: none"> – print-Syntax (5min) – Programmdatei anlegen und speichern (5min) – Wiederholung Variable + (Strings und) Zahlen als Variablen (10min) 	<ul style="list-style-type: none"> – print-Syntax (Bsp. Hello World) in Thonny zeigen (String) – zeigen, wie man Datei speichert + ausführt – Konzept Variable erfragen (Plenum) (mit Analogie Karton) – Einfaches Beispiel in Thonny zeigen, das Variablen enthält (Was sind hier die Variablen? Woran erkennt man diese?) – Darauf basierend Syntax für Variablen in Thonny zeigen – (Beispiele für Strings als Variablen geben + print) 	<ul style="list-style-type: none"> – Konzept Variable erklären: Variable kann Wert abspeichern für spätere Verwendung, veränderlich – eigene Beispiele für geprintete Strings generieren 	<ul style="list-style-type: none"> – Variablenkonzept aufzeigen können – Die SuS können Programmdateien anlegen und abspeichern – Die SuS können wichtige Befehle wie „print“ ausführen
Übung und Anwendung	<ul style="list-style-type: none"> – kleine Aufgaben mit Zahlen (15 min) – kleine Aufgaben mit Strings (15 min) – Puffer + Hausaufgabe (10 min): kombinierte Aufgabe: Steckbrief erstellen (insg. 30 min) 	<ul style="list-style-type: none"> – AB austeilen – Beschreibungen vorgeben – Puffer: Struktogramm einführen, für Minimalbeispiel Steckbrief vorstellen (Name, Alter) -> Hausaufgabe: Struktogramm in python-online.ch umsetzen, dann Struktogramm und Code um weitere (min. 3) Stichpunkte erweitern und schriftlich 	<ul style="list-style-type: none"> – Die SuS initialisieren, verändern und geben Variablen aus – SuS speichern Programme für jeden Aufgabe separat auf Laufwerk H: (benutzername) – Struktogramm in Code umwandeln können – Hausaufgabe bearbeiten, im Coderunner ausprobieren und als Papiermitschrieb 	<ul style="list-style-type: none"> – SuS schreiben durch Nutzung unterschiedlicher Variablentypen kleine Programme, um Problemstellungen zu lösen – Erstellen von Code aus entsprechenden Struktogrammen

(Hinweise zur Ergebnissicherung werden in den Spalten Lehrer- bzw. Schülerhandeln eingetragen)

Unterrichtsverlaufsplan

		festhalten	mitbringen	
--	--	------------	------------	--

(Hinweise zur Ergebnissicherung werden in den Spalten Lehrer- bzw. Schülerhandeln eingetragen)

Unterrichtsverlaufsplan

Phase	Unterrichtsstruktur (mit Zeitplanung)	Lehrerhandeln	Schülerhandeln	Lernziele (fachliche und überfachliche)
Unterrichtseinstieg (15 min)	<ul style="list-style-type: none"> – Struktogramme erklären (5 min) – Hausaufgabe eintippen lassen (5 min) – Hausaufgabenbeispiele besprechen (5 min) 	<ul style="list-style-type: none"> – Begrüßung – vom einfachen Struktogramm zum ganzen (ausführlich ausgearbeitet) – „Struktogramme werden wir nachher noch brauchen, wenn die Programme komplizierter werden, z.B. mit if-Befehlen“ – Bildschirme auf Beamer holen von SuS, die ihre Steckbriefe vorstellen 	<ul style="list-style-type: none"> – Aufmerksam zuhören – Hausaufgaben vorstellen (anfangs in Thonny abtippen -> Bildschirm holen können) – Hausaufgaben bereitstellen 	<ul style="list-style-type: none"> – Die SuS können den Aufbau von Struktogrammen erinnern – Die SuS können aus einem Struktogramm mit Variablendeklarationen, -operationen und Ausgaben ein Programm entwickeln – Die SuS können aus einem Programm ein Struktogramm erstellen, um ihre Umsetzung abstrahiert darzustellen
Erarbeitung 1 (15 min)	<ul style="list-style-type: none"> – Hausaufgabe in Struktogramm (+ die Erweiterungen der Vorgabe, 5 min) – Eingabe; Berechnung mit Eingabeparameter + Einführung Datentypen: string -> int; int -> string (10 min) 	<p>Eingabe und Datentypenkonversion</p> <ul style="list-style-type: none"> – Eingabe einführen (input(String)) – Rechenoperation mit Eingabe ausgeben -> Datentypen: Problem der Typkonversion aufgeworfen, Szenario: eine Variable mit Eingabe addiert ausgeben, (genauer: String und Int innerhalb des print-Befehls, aber trz in einer Zeile ausgeben) – A7 wieder aufrufen, Fehlermeldungen noch bekannt, Lösung mit dem Wissen letzter Woche nicht möglich, da Datentypen nicht bekannt -> str und int einführen, sodass solche Probleme nicht nochmal 	<ul style="list-style-type: none"> – Code in Struktogramm umwandeln – Aufmerksames zuhören – Fragestellungen beantworten (aktive Unterrichtsteilnahme) 	<ul style="list-style-type: none"> – SuS können Anweisungen zur Benutzereingabe von Daten entwickeln – SuS können Variablen von Typ Text in Typ Zahl und umgekehrt umwandeln – SuS können Anweisungen zur Benutzereingabe in Variablen vom Typ Zahl entwickeln

(Hinweise zur Ergebnissicherung werden in den Spalten Lehrer- bzw. Schülerhandeln eingetragen)

Unterrichtsverlaufsplan

		<ul style="list-style-type: none"> auftreten – typische Fehler zeigen oder ausprobieren lassen, int()-Funktion einführen(int(„Hallo“)) -> int in str möglich, str in int nur wenn Zahl 		
Übung und Anwendung 1 (15 min)	<ul style="list-style-type: none"> – Einfache Aufgaben zur Typkonversion, z.B. Ergebnisse einer Summe berechnen und als String ausgeben, formatierte Rechnungen ausgeben (10min) – Besprechung (5min) 	<ul style="list-style-type: none"> – Unterstützung bei Fragen – AB austeilen 	<ul style="list-style-type: none"> – SuS wandeln Strings in Integer und Integer in Strings um 	siehe oben
Erarbeitung 2 (15 min)	Funktionen (Unterprogramme) <ul style="list-style-type: none"> – Wofür Fkt? -> Strukturierung; Verständnis durch Namensgebung von Anweisungsketten; Wiederverwendbarkeit incl. Beispiel Körpergrößen (15 min) 	<ul style="list-style-type: none"> – Motivation von Funktionen erklären (Aufstehen lassen, Durchschnittsgrößen der Gruppen schätzen lassen) – Beispiel Körpergrößen anwenden – Syntax von Funktionen: def (param1, ...) mit Beispiel: (Wiederverwendbarkeit) – Rückgabe erklären -> Ausgabe von Fkt (wie y bei Mathematischer Fkt) 	<ul style="list-style-type: none"> – Aufmerksames zuhören – evtl Aufstehen, schätzen welche Gruppe durchschnittlich am Größten – jeder für sich schreibt Funktionsbeispiel ab, setzt Körpergrößen in seinem Programm ein, vergleicht innerhalb der Gruppe und danach werden alle Gruppen verglichen 	<ul style="list-style-type: none"> – SuS können Funktion und Syntax einer Funktion erläutern und anwenden – SuS können Bedeutung von Parametern als Eingabewerte erläutern – SuS können Funktionen mit Parametern entwickeln – SuS können Funktionsaufrufe in Ihren Programme verwenden – SuS können Rückgabewerte zur Verarbeitung durch weitere Funktionen verwenden
Übung und Anwendung 2 (30 min)	<ul style="list-style-type: none"> – Übung 1: Grammberechnung und Promillerechner implementieren (10 min) – Input: Weitergabe von 	<ul style="list-style-type: none"> – Austeilen AB Promille (Übungsphase 1) – „Gleicht euer erstes Ergebnis mit dem angegebenen Lösungswert ab, falls ihr dann nicht weiter kommt, fragt uns 	<ul style="list-style-type: none"> – SuS implementieren die Berechnungsformel für Promillewert mit Variablen für Gewicht (und Geschlecht) $P = \text{Alk} / (\text{Gew} * 0.65 \rightarrow (20g / (70 \text{ kg} * 0.65)))$ – SuS implementieren die 	siehe oben

(Hinweise zur Ergebnissicherung werden in den Spalten Lehrer- bzw. Schülerhandeln eingetragen)

Unterrichtsverlaufsplan

	<p>Rückgaben an weitere Funktionen (5min)</p> <ul style="list-style-type: none"> – Übung 2: Abbauformel auf Basis der Promilleberechnung (5min) – Besprechung der Übung 2 (5min) 	<p>einfach.“</p> <ul style="list-style-type: none"> – Individuelle Sicherung + Input: Rückgabe an weitere Funktionen (für Abbauformel) – Abbauformel umsetzen lassen (Übungsphase 2) – Besprechung mit Schülerbeispielen (Puffer: evtl. auch fehlerhafte gemeinsam korrigieren) – gesamt: Unterstützung während Übungsphase 	<p>weiteren Aufgaben aus Übungsteil 2</p>	
--	--	---	---	--