

Informatik 7

Arbeitsheft mit 33 Lernseiten, Projekten und Lösungshinweisen

So arbeitest du mit dem Heft

1. Lies Kurz erklärt und markiere Fachbegriffe.
2. Löse zuerst ohne Lösungsteil.
3. Prüfe mit einem Testfall.
4. Verbessere deine Antwort sichtbar.

Dieses Heft ist vollständig eigenständig für SchlauStart erstellt. Es enthält keine kopierten Buchseiten oder fremden Arbeitsblätter. Für Aufgaben werden keine Namen, Kontodaten oder anderen personenbezogenen Angaben benötigt.

Inhalt

01	Scratch
02	Scratch vertiefen
03	Nachrichten zwischen Figuren
04	Klone und Listen
05	Eigene Blöcke in Scratch
06	Ein Lernspiel entwickeln
07	Programmieren: Grundlagen
08	Vom Problem zum Programm
09	Probleme zerlegen
10	Anforderungen und Testfälle
11	Pseudocode schreiben
12	Daten und Steuerung
13	Variablen und Datentypen
14	Eingabe und Ausgabe
15	Bedingungen programmieren
16	Schleifen auswählen
17	Programme strukturieren
18	Funktionen und Parameter
19	Rückgabewerte verwenden
20	Fehler systematisch suchen
21	Robotik
22	Sensoren, Verarbeitung und Aktoren
23	Ozobot mit Farbcodes steuern
24	Calliope mini programmieren
25	LEGO-Roboter programmieren
26	Open Roberta im Simulator
27	Ein Robotikprojekt planen
28	Cybermobbing
29	Cybermobbing erkennen
30	Rollen und Gruppendynamik
31	Beweise sichern und Vorfälle melden
32	Betroffene wirksam unterstützen
33	Klassenregeln für digitale Räume

01 · Scratch

Name: _____

Datum: _____

Kurz erklärt

Programmieren mit Scratch Scratch stellt Programmbefehle als zusammensteckbare Blöcke dar. So werden Abläufe, Ereignisse, Schleifen, Bedingungen und Variablen direkt sichtbar.

Beispiel: Ein Programm startet mit der grünen Flagge, bewegt eine Figur und reagiert auf eine gedrückte Taste.

Drei Fachbegriffe

Ereignis	Ein Startsignal löst ein Skript aus.
Zustand	Position, Variable oder Kostüm speichern den aktuellen Stand.
Nebenläufigkeit	Mehrere Skripte können gleichzeitig reagieren.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane ein Mini-Spiel mit Start-Ereignis, Bewegung, Hindernis, Punktestand und eindeutigem Spielende.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt in drei Etappen ein eigenes Lernspiel: erst steuerbare Figur, dann Spielregel, zuletzt Punkte und Ende. Dokumentiert nach jeder Etappe einen Test.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

02 · Scratch vertiefen

Name: _____

Datum: _____

Kurz erklärt

Scratch vertiefen Größere Scratch-Projekte brauchen Nachrichten, Listen, Klone und eigene Blöcke. Diese Werkzeuge teilen das Programm in verständliche Zuständigkeiten.

Beispiel: Eine Nachricht startet gleichzeitig Musik und Gegner; ein eigener Block bündelt die wiederholte Bewegung eines Hindernisses.

Drei Fachbegriffe

Ereignis	Ein Startsignal löst ein Skript aus.
Zustand	Position, Variable oder Kostüm speichern den aktuellen Stand.
Nebenläufigkeit	Mehrere Skripte können gleichzeitig reagieren.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane ein Mini-Spiel mit Start-Ereignis, Bewegung, Hindernis, Punktestand und eindeutigem Spielende.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt in drei Etappen ein eigenes Lernspiel: erst steuerbare Figur, dann Spielregel, zuletzt Punkte und Ende. Dokumentiert nach jeder Etappe einen Test.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

03 · Nachrichten zwischen Figuren

Name: _____

Datum: _____

Kurz erklärt

Nachrichten zwischen Figuren Mit Senden und Empfangen reagieren Figuren auf gemeinsame Ereignisse, ohne direkt das Skript einer anderen Figur zu verändern.

Beispiel: Die Spielfigur sendet 'Runde vorbei'; Bühne, Gegner und Punkteanzeige führen daraufhin jeweils ihr eigenes Endskript aus.

Drei Fachbegriffe

Ereignis	Ein Startsignal löst ein Skript aus.
Zustand	Position, Variable oder Kostüm speichern den aktuellen Stand.
Nebenläufigkeit	Mehrere Skripte können gleichzeitig reagieren.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane ein Mini-Spiel mit Start-Ereignis, Bewegung, Hindernis, Punktestand und eindeutigem Spielende.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt in drei Etappen ein eigenes Lernspiel: erst steuerbare Figur, dann Spielregel, zuletzt Punkte und Ende. Dokumentiert nach jeder Etappe einen Test.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

04 · Klone und Listen

Name: _____

Datum: _____

Kurz erklärt

Klone und Listen Klone erzeugen viele gleichartige Figuren aus einer Vorlage. Listen speichern mehrere Werte in fester Reihenfolge und können während des Spiels verändert werden.

Beispiel: Ein Fangspiel erzeugt Münzen als Klone und speichert die Namen der erreichten Level in einer Liste.

Drei Fachbegriffe

Ereignis	Ein Startsignal löst ein Skript aus.
Zustand	Position, Variable oder Kostüm speichern den aktuellen Stand.
Nebenläufigkeit	Mehrere Skripte können gleichzeitig reagieren.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane ein Mini-Spiel mit Start-Ereignis, Bewegung, Hindernis, Punktestand und eindeutigem Spielende.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt in drei Etappen ein eigenes Lernspiel: erst steuerbare Figur, dann Spielregel, zuletzt Punkte und Ende. Dokumentiert nach jeder Etappe einen Test.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

05 · Eigene Blöcke in Scratch

Name: _____

Datum: _____

Kurz erklärt

Eigene Blöcke in Scratch Eigene Blöcke fassen zusammengehörige Befehle unter einem sprechenden Namen zusammen. Parameter machen denselben Ablauf mit unterschiedlichen Werten nutzbar.

Beispiel: Der Block 'zeichne Vieleck mit Seitenlänge' erhält Seitenzahl und Länge und ersetzt mehrere fast gleiche Skripte.

Drei Fachbegriffe

Ereignis	Ein Startsignal löst ein Skript aus.
Zustand	Position, Variable oder Kostüm speichern den aktuellen Stand.
Nebenläufigkeit	Mehrere Skripte können gleichzeitig reagieren.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane ein Mini-Spiel mit Start-Ereignis, Bewegung, Hindernis, Punktestand und eindeutigem Spielende.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt in drei Etappen ein eigenes Lernspiel: erst steuerbare Figur, dann Spielregel, zuletzt Punkte und Ende. Dokumentiert nach jeder Etappe einen Test.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

06 · Ein Lernspiel entwickeln

Name: _____

Datum: _____

Kurz erklärt

Ein Lernspiel entwickeln Ein Lernspiel verbindet fachlich richtige Aufgaben mit verständlicher Rückmeldung, steigendem Anspruch und einer fairen Spielregel.

Beispiel: Ein Vokabelspiel erklärt nach jeder Antwort den Fehler, mischt die Reihenfolge und beendet eine Runde erst nach zehn geprüften Aufgaben.

Drei Fachbegriffe

Ereignis	Ein Startsignal löst ein Skript aus.
Zustand	Position, Variable oder Kostüm speichern den aktuellen Stand.
Nebenläufigkeit	Mehrere Skripte können gleichzeitig reagieren.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane ein Mini-Spiel mit Start-Ereignis, Bewegung, Hindernis, Punktestand und eindeutigem Spielende.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt in drei Etappen ein eigenes Lernspiel: erst steuerbare Figur, dann Spielregel, zuletzt Punkte und Ende. Dokumentiert nach jeder Etappe einen Test.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

07 · Programmieren: Grundlagen

Name: _____

Datum: _____

Kurz erklärt

Programmieren: Grundlagen Programmieren übersetzt eine Problemlösung in eindeutige Daten und Anweisungen. Planung, kleine Programmteile, Tests und Überarbeitung gehören zum gesamten Prozess.

Beispiel: Ein Fahrkartenprogramm wird zuerst in Eingaben, Tarifregeln, Berechnung und Ausgabe zerlegt und danach schrittweise umgesetzt.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

08 · Vom Problem zum Programm

Name: _____

Datum: _____

Kurz erklärt

Vom Problem zum Programm Vor dem Code werden Ziel, Eingaben, Ausgaben, Regeln und Sonderfälle verstanden. Erst dann entsteht ein Algorithmus und schließlich ein Programm.

Beispiel: Für einen Notenrechner werden zulässige Punkte, Maximalpunktzahl, Rundung und Fehlermeldungen festgelegt, bevor eine Formel geschrieben wird.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

09 · Probleme zerlegen

Name: _____

Datum: _____

Kurz erklärt

Probleme zerlegen Dekomposition teilt eine große Aufgabe in kleine Teilprobleme mit klaren Schnittstellen. Jedes Teil lässt sich einzeln verstehen und testen.

Beispiel: Ein Quiz zerfällt in Frage anzeigen, Eingabe lesen, Antwort prüfen, Punkte ändern und Ergebnis ausgeben.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

10 · Anforderungen und Testfälle

Name: _____

Datum: _____

Kurz erklärt

Anforderungen und Testfälle Anforderungen sagen überprüfbar, was ein Programm leisten soll. Testfälle verbinden Eingaben mit erwarteten Ergebnissen und decken Normal-, Rand- und Fehlerfälle ab.

Beispiel: Ein Altersprüfer wird mit 17, 18, 19, leerer Eingabe und Text statt Zahl getestet.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

11 · Pseudocode schreiben

Name: _____

Datum: _____

Kurz erklärt

Pseudocode schreiben Pseudocode beschreibt Programmlogik mit Einrückung und klaren Schlüsselwörtern, ohne an die genaue Schreibweise einer Programmiersprache gebunden zu sein.

Beispiel: WIEDERHOLE für jede Zahl, WENN Zahl gerade, DANN addiere sie zur Summe beschreibt den Kern unabhängig von Python oder Scratch.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

12 · Daten und Steuerung

Name: _____

Datum: _____

Kurz erklärt

Daten und Steuerung Programme speichern Werte und verändern ihren Ablauf durch Sequenzen, Verzweigungen und Schleifen. Datenzustand und Steuerfluss müssen gemeinsam betrachtet werden.

Beispiel: Ein Kontostand wird als Zahl gespeichert; eine Bedingung verhindert eine Auszahlung, wenn der Betrag nicht ausreicht.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

13 · Variablen und Datentypen

Name: _____

Datum: _____

Kurz erklärt

Variablen und Datentypen Variablen benennen gespeicherte Werte. Datentypen legen fest, welche Werte und Operationen sinnvoll sind, etwa Zahl, Text oder Wahrheitswert.

Beispiel: Die Zeichenfolge '12' wird an '3' zu '123' angehängt, während die Zahlen 12 und 3 zu 15 addiert werden.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

14 · Eingabe und Ausgabe

Name: _____

Datum: _____

Kurz erklärt

Eingabe und Ausgabe Eingaben liefern Daten von Menschen, Sensoren oder Dateien. Ausgaben machen Ergebnisse sichtbar oder steuern andere Systeme; Validierung trennt gültige von ungültigen Werten.

Beispiel: Ein Temperaturprogramm liest einen Sensorwert, prüft den Messbereich und zeigt Wert oder verständliche Fehlermeldung.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

15 · Bedingungen programmieren

Name: _____

Datum: _____

Kurz erklärt

Bedingungen programmieren Bedingungen ergeben wahr oder falsch und steuern alternative Programmwege. Reihenfolge und vollständige Fälle verhindern widersprüchliche Ergebnisse.

Beispiel: Bei Punkten ab 90 gilt Gold, ab 70 Silber, sonst Bronze; die höchste Grenze wird zuerst geprüft.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

16 · Schleifen auswählen

Name: _____

Datum: _____

Kurz erklärt

Schleifen auswählen Eine Zählschleife passt zu bekannter Wiederholungszahl, eine Bedingungsschleife zu offenem Ende. Jede Schleife braucht veränderlichen Zustand und erreichbaren Abbruch.

Beispiel: Fünf Messwerte werden mit einer Zählschleife gelesen; eine Passworteingabe läuft bis zur korrekten Eingabe oder maximal drei Versuchen.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

17 · Programme strukturieren

Name: _____

Datum: _____

Kurz erklärt

Programme strukturieren Sprechende Namen, kleine Funktionen und klar getrennte Zuständigkeiten machen Programme verständlich, testbar und wiederverwendbar.

Beispiel: Statt einer langen Folge übernehmen `preis_berechnen`, `rabatt_pruefen` und `beleg_ausgeben` jeweils eine Aufgabe.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

18 · Funktionen und Parameter

Name: _____

Datum: _____

Kurz erklärt

Funktionen und Parameter Eine Funktion bündelt einen benannten Ablauf. Parameter liefern veränderliche Eingaben, sodass derselbe Code unterschiedliche Fälle bearbeiten kann.

Beispiel: flaeche(breite, hoehe) berechnet Rechtecke verschiedener Größe, ohne die Formel zu kopieren.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

19 · Rückgabewerte verwenden

Name: _____

Datum: _____

Kurz erklärt

Rückgabewerte verwenden Ein Rückgabewert übermittelt das Ergebnis einer Funktion an die aufrufende Stelle. Ausgabe auf dem Bildschirm und Rückgabe sind unterschiedliche Aufgaben.

Beispiel: `ist_gerade(14)` gibt wahr zurück; erst der aufrufende Programmteil entscheidet, ob daraus eine Meldung entsteht.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

20 · Fehler systematisch suchen

Name: _____

Datum: _____

Kurz erklärt

Fehler systematisch suchen Debugging reproduziert einen Fehler, grenzt seine Stelle ein, prüft Annahmen und verändert jeweils nur eine Ursache. Danach laufen alte und neue Tests erneut.

Beispiel: Ein falscher Punktestand wird mit protokollierten Werten vor und nach der Trefferbedingung auf eine doppelte Erhöhung eingegrenzt.

Drei Fachbegriffe

Variable	Ein Name verweist auf einen veränderlichen Wert.
Kontrollstruktur	Sequenz, Auswahl und Wiederholung steuern den Ablauf.
Funktion	Ein benannter Teil löst eine klar begrenzte Aufgabe.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Formuliere für einen Punkte-Rechner Eingaben, Ausgabe, zwei Regeln und drei Testfälle.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen kleinen Alltagshelfer. Erstellt Anforderungsliste, Pseudocode, Testtabelle und anschließend eine funktionsfähige Umsetzung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

21 · Robotik

Name: _____

Datum: _____

Kurz erklärt

Robotik Roboter verbinden Mechanik, Elektronik und Programme. Sensoren erfassen die Umwelt, ein Programm entscheidet und Aktoren verändern Bewegung, Licht oder Ton.

Beispiel: Ein Fahrroboter misst den Abstand, bremst unter 15 Zentimetern und dreht die Motoren anschließend in verschiedene Richtungen.

Drei Fachbegriffe

Sensor	Er wandelt eine Umwelteigenschaft in einen Messwert um.
Aktor	Er setzt ein Steuersignal in eine Handlung um.
Regelkreis	Messung und Reaktion beeinflussen sich fortlaufend.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane eine Hindernissteuerung mit Abstandssensor, zwei Motoren und einem sicheren Stopp.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Baut oder simuliert einen Roboterparcours. Dokumentiert Sensorwerte, drei Testläufe, eine Fehlersuche und die begründete Verbesserung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

22 · Sensoren, Verarbeitung und Aktoren

Name: _____

Datum: _____

Kurz erklärt

Sensoren, Verarbeitung und Aktoren Sensoren wandeln physikalische Größen in Daten um. Die Verarbeitung vergleicht Werte mit Regeln; Aktoren setzen Entscheidungen in Handlungen um.

Beispiel: Ein Lichtsensor liefert einen Zahlenwert, das Programm prüft einen Grenzwert und schaltet als Aktor eine LED ein.

Drei Fachbegriffe

Sensor	Er wandelt eine Umwelteigenschaft in einen Messwert um.
Aktor	Er setzt ein Steuersignal in eine Handlung um.
Regelkreis	Messung und Reaktion beeinflussen sich fortlaufend.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane eine Hindernissteuerung mit Abstandssensor, zwei Motoren und einem sicheren Stopp.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Baut oder simuliert einen Roboterparcours. Dokumentiert Sensorwerte, drei Testläufe, eine Fehlersuche und die begründete Verbesserung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

23 · Ozobot mit Farbcodes steuern

Name: _____

Datum: _____

Kurz erklärt

Ozobot mit Farbcodes steuern Ozobot folgt Linien und erkennt definierte Farbfolgen als Befehle. Saubere Linien, passende Abstände und kontrollierte Geschwindigkeit sind Teil des Programms.

Beispiel: Eine Farbfolge vor einer Kreuzung kann die Fahrtrichtung festlegen; ein Testparcours zeigt, ob Farbe und Position zuverlässig gelesen werden.

Drei Fachbegriffe

Sensor	Er wandelt eine Umwelteigenschaft in einen Messwert um.
Aktor	Er setzt ein Steuersignal in eine Handlung um.
Regelkreis	Messung und Reaktion beeinflussen sich fortlaufend.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane eine Hindernissteuerung mit Abstandssensor, zwei Motoren und einem sicheren Stopp.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Baut oder simuliert einen Roboterparcours. Dokumentiert Sensorwerte, drei Testläufe, eine Fehlersuche und die begründete Verbesserung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

24 · Calliope mini programmieren

Name: _____

Datum: _____

Kurz erklärt

Calliope mini programmieren Der Calliope mini kombiniert Tasten, Sensoren, LEDs, Funk und Anschlüsse auf einem Mikrocontroller. Programme reagieren auf Ereignisse und Messwerte.

Beispiel: Beim Drücken von Taste A zeigt die LED-Matrix ein Symbol; bei hoher Lautstärke wechselt es zu einer Warnanzeige.

Drei Fachbegriffe

Sensor	Er wandelt eine Umwelteigenschaft in einen Messwert um.
Aktor	Er setzt ein Steuersignal in eine Handlung um.
Regelkreis	Messung und Reaktion beeinflussen sich fortlaufend.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane eine Hindernissteuerung mit Abstandssensor, zwei Motoren und einem sicheren Stopp.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Baut oder simuliert einen Roboterparcours. Dokumentiert Sensorwerte, drei Testläufe, eine Fehlersuche und die begründete Verbesserung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

25 · LEGO-Roboter programmieren

Name: _____

Datum: _____

Kurz erklärt

LEGO-Roboter programmieren Ein LEGO-Roboter steuert Motoren über Leistung, Richtung und Dauer und nutzt Sensorwerte für Rückmeldung. Bauweise und Programm beeinflussen gemeinsam das Fahrverhalten.

Beispiel: Ein Abstandssensor löst Bremsen aus; ungleiche Räder oder Motorleistungen müssen beim Geradeausfahren berücksichtigt werden.

Drei Fachbegriffe

Sensor	Er wandelt eine Umwelteigenschaft in einen Messwert um.
Aktor	Er setzt ein Steuersignal in eine Handlung um.
Regelkreis	Messung und Reaktion beeinflussen sich fortlaufend.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane eine Hindernissteuerung mit Abstandssensor, zwei Motoren und einem sicheren Stopp.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Baut oder simuliert einen Roboterparcours. Dokumentiert Sensorwerte, drei Testläufe, eine Fehlersuche und die begründete Verbesserung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

26 · Open Roberta im Simulator

Name: _____

Datum: _____

Kurz erklärt

Open Roberta im Simulator Open Roberta stellt Programme mit NEPO-Blöcken dar und kann viele Abläufe ohne Hardware simulieren. Simulation prüft Logik, ersetzt aber nicht jeden realen Sensorfehler.

Beispiel: Ein virtueller Roboter folgt einer Linie; danach wird geprüft, welche Werte und Geschwindigkeiten am echten Gerät angepasst werden müssen.

Drei Fachbegriffe

Sensor	Er wandelt eine Umwelteigenschaft in einen Messwert um.
Aktor	Er setzt ein Steuersignal in eine Handlung um.
Regelkreis	Messung und Reaktion beeinflussen sich fortlaufend.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane eine Hindernissteuerung mit Abstandssensor, zwei Motoren und einem sicheren Stopp.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Baut oder simuliert einen Roboterparcours. Dokumentiert Sensorwerte, drei Testläufe, eine Fehlersuche und die begründete Verbesserung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

27 · Ein Robotikprojekt planen

Name: _____

Datum: _____

Kurz erklärt

Ein Robotikprojekt planen Ein Robotikprojekt braucht Aufgabe, Umweltannahmen, Sensoren, Aktoren, Zustände, Sicherheitsgrenzen und messbare Tests.

Beispiel: Ein Rettungsroboter soll einen markierten Bereich finden; Testfälle prüfen Kurven, verlorene Linie, Hindernis und leere Batterie.

Drei Fachbegriffe

Sensor	Er wandelt eine Umwelteigenschaft in einen Messwert um.
Aktor	Er setzt ein Steuersignal in eine Handlung um.
Regelkreis	Messung und Reaktion beeinflussen sich fortlaufend.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Plane eine Hindernissteuerung mit Abstandssensor, zwei Motoren und einem sicheren Stopp.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Baut oder simuliert einen Roboterparcours. Dokumentiert Sensorwerte, drei Testläufe, eine Fehlersuche und die begründete Verbesserung.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

28 · Cybermobbing

Name: _____

Datum: _____

Kurz erklärt

Cybermobbing Cybermobbing ist wiederholte digitale Herabsetzung, Ausgrenzung oder Bedrohung mit Machtungleichgewicht. Reichweite, Kopierbarkeit und ständige Erreichbarkeit verstärken die Wirkung.

Beispiel: Ein peinliches Bild wird trotz Bitte weiterverbreitet und mit abwertenden Kommentaren in mehreren Gruppen geteilt.

Drei Fachbegriffe

Machtgefälle	Betroffene können sich nicht gleich wirksam wehren.
Reichweite	Kopien vergrößern Publikum und Dauer eines Angriffs.
Hilfeweg	Er nennt sichere Schritte und verantwortliche Ansprechpersonen.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Erstelle einen Hilfeplan für einen verletzenden Gruppenchat mit sicheren Sofortschritten und zwei Ansprechstellen.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen Klassenkodex mit konkreten Beispielen, Meldeweg, Unterstützerrollen und einer Übung für den Ernstfall.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

29 · Cybermobbing erkennen

Name: _____

Datum: _____

Kurz erklärt

Cybermobbing erkennen Unterschieden werden einmaliger Streit, Konflikt und systematisches Mobbing. Wiederholung, Absicht, Wirkung, Öffentlichkeit und Machtverhältnis liefern Hinweise.

Beispiel: Ein einzelner unhöflicher Kommentar ist ernst zu nehmen, eine wochenlange koordinierte Ausgrenzung erfüllt zusätzlich typische Mobbingmerkmale.

Drei Fachbegriffe

Machtgefälle	Betroffene können sich nicht gleich wirksam wehren.
Reichweite	Kopien vergrößern Publikum und Dauer eines Angriffs.
Hilfeweg	Er nennt sichere Schritte und verantwortliche Ansprechpersonen.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Erstelle einen Hilfeplan für einen verletzenden Gruppenchat mit sicheren Sofortschritten und zwei Ansprechstellen.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen Klassenkodex mit konkreten Beispielen, Meldeweg, Unterstützerrollen und einer Übung für den Ernstfall.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

30 · Rollen und Gruppendynamik

Name: _____

Datum: _____

Kurz erklärt

Rollen und Gruppendynamik Neben betroffener und angreifender Person beeinflussen Mitmachende, Zuschauende, Unterstützende und Moderierende den Verlauf. Schweigen kann als Zustimmung wirken.

Beispiel: Eine Person leitet nicht weiter, widerspricht sachlich, sichert Hilfe und verändert damit die Gruppennorm.

Drei Fachbegriffe

Machtgefälle	Betroffene können sich nicht gleich wirksam wehren.
Reichweite	Kopien vergrößern Publikum und Dauer eines Angriffs.
Hilfeweg	Er nennt sichere Schritte und verantwortliche Ansprechpersonen.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Erstelle einen Hilfeplan für einen verletzenden Gruppenchat mit sicheren Sofortschritten und zwei Ansprechstellen.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen Klassenkodex mit konkreten Beispielen, Meldeweg, Unterstützerrollen und einer Übung für den Ernstfall.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

31 · Beweise sichern und Vorfälle melden

Name: _____

Datum: _____

Kurz erklärt

Beweise sichern und Vorfälle melden Vor dem Blockieren oder Löschen werden relevante Inhalte mit Datum, Kontext, Nutzernamen und URL gesichert. Meldung erfolgt an Plattform und vertrauenswürdige Erwachsene.

Beispiel: Screenshots zeigen nicht nur die Beleidigung, sondern auch Verlauf und Zeitpunkt; Passwörter und private Daten werden dabei nicht weiterverteilt.

Drei Fachbegriffe

Machtgefälle	Betroffene können sich nicht gleich wirksam wehren.
Reichweite	Kopien vergrößern Publikum und Dauer eines Angriffs.
Hilfeweg	Er nennt sichere Schritte und verantwortliche Ansprechpersonen.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Erstelle einen Hilfeplan für einen verletzenden Gruppenchat mit sicheren Sofortschritten und zwei Ansprechstellen.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen Klassenkodex mit konkreten Beispielen, Meldeweg, Unterstützerrollen und einer Übung für den Ernstfall.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

32 · Betroffene wirksam unterstützen

Name: _____

Datum: _____

Kurz erklärt

Betroffene wirksam unterstützen Unterstützung nimmt die Erfahrung ernst, vermeidet Schuldzuweisung, stellt Sicherheit her und bezieht geeignete Vertrauenspersonen ein.

Beispiel: Statt 'geh einfach offline' fragt eine Mitschülerin, was gebraucht wird, begleitet zur Lehrkraft und hilft beim Melden.

Drei Fachbegriffe

Machtgefälle	Betroffene können sich nicht gleich wirksam wehren.
Reichweite	Kopien vergrößern Publikum und Dauer eines Angriffs.
Hilfeweg	Er nennt sichere Schritte und verantwortliche Ansprechpersonen.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Erstelle einen Hilfeplan für einen verletzenden Gruppenchat mit sicheren Sofortschritten und zwei Ansprechstellen.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen Klassenkodex mit konkreten Beispielen, Meldeweg, Unterstützerrollen und einer Übung für den Ernstfall.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

33 · Klassenregeln für digitale Räume

Name: _____

Datum: _____

Kurz erklärt

Klassenregeln für digitale Räume Wirksame Regeln benennen erlaubtes Verhalten, Grenzen, Hilfewege und nachvollziehbare Folgen. Sie werden gemeinsam geübt und regelmäßig überprüft.

Beispiel: Die Klasse vereinbart: keine Screenshots ohne Zustimmung, Konflikte nicht weiterleiten, Vorfälle sichern und an zwei benannte Ansprechpersonen melden.

Drei Fachbegriffe

Machtgefälle	Betroffene können sich nicht gleich wirksam wehren.
Reichweite	Kopien vergrößern Publikum und Dauer eines Angriffs.
Hilfeweg	Er nennt sichere Schritte und verantwortliche Ansprechpersonen.

1. Erkläre das Thema in eigenen Worten. Nutze mindestens zwei Fachbegriffe aus der Tabelle.

2. Zerlege das Beispiel in Ausgangslage, Verarbeitung und überprüfbares Ergebnis.

3. Erstelle einen Hilfeplan für einen verletzenden Gruppenchat mit sicheren Sofortschritten und zwei Ansprechstellen.

4. Notiere einen Rand- oder Fehlerfall. Sage zuerst voraus, was passieren soll, und begründe danach deine Erwartung.

Mini-Projekt

Entwickelt einen Klassenkodex mit konkreten Beispielen, Meldeweg, Unterstützerrollen und einer Übung für den Ernstfall.

Dokumentiere Ziel, Plan, drei Tests, einen gefundenen Fehler und eine Verbesserung.

Lösungen und Bewertungshinweise

Die Hinweise zeigen mögliche fachliche Kerne. Offene Aufgaben dürfen anders gelöst sein, wenn Begriffe, Begründung und Test nachvollziehbar stimmen.

Lösungshinweise

01 · Scratch

1: Scratch stellt Programmbefehle als zusammensteckbare Blöcke dar. So werden Abläufe, Ereignisse, Schleifen, Bedingungen und Variablen direkt sichtbar.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Programm startet mit der grünen Flagge, bewegt eine Figur und reagiert auf eine gedrückte Taste. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Der Plan braucht mindestens ein Ereignis, eine wiederholte Prüfung, eine Bedingung, eine Variable und ein überprüfbares Ziel.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

02 · Scratch vertiefen

1: Größere Scratch-Projekte brauchen Nachrichten, Listen, Klone und eigene Blöcke. Diese Werkzeuge teilen das Programm in verständliche Zuständigkeiten.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Eine Nachricht startet gleichzeitig Musik und Gegner; ein eigener Block bündelt die wiederholte Bewegung eines Hindernisses. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Der Plan braucht mindestens ein Ereignis, eine wiederholte Prüfung, eine Bedingung, eine Variable und ein überprüfbares Ziel.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

03 · Nachrichten zwischen Figuren

1: Mit Senden und Empfangen reagieren Figuren auf gemeinsame Ereignisse, ohne direkt das Skript einer anderen Figur zu verändern.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Die Spielfigur sendet 'Runde vorbei'; Bühne, Gegner und Punkteanzeige führen daraufhin jeweils ihr eigenes Endskript aus. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Der Plan braucht mindestens ein Ereignis, eine wiederholte Prüfung, eine Bedingung, eine Variable und ein überprüfbares Ziel.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

04 · Klone und Listen

1: Klone erzeugen viele gleichartige Figuren aus einer Vorlage. Listen speichern mehrere Werte in fester Reihenfolge und können während des Spiels verändert werden.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Fangspiel erzeugt Münzen als Klone und speichert die Namen der erreichten Level in einer Liste. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Der Plan braucht mindestens ein Ereignis, eine wiederholte Prüfung, eine Bedingung, eine Variable und ein überprüfbares Ziel.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

05 · Eigene Blöcke in Scratch

1: Eigene Blöcke fassen zusammengehörige Befehle unter einem sprechenden Namen zusammen. Parameter machen denselben Ablauf mit unterschiedlichen Werten nutzbar.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Der Block 'zeichne Vieleck mit Seitenlänge' erhält Seitenzahl und Länge und ersetzt mehrere fast gleiche Skripte. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Der Plan braucht mindestens ein Ereignis, eine wiederholte Prüfung, eine Bedingung, eine Variable und ein überprüfbares Ziel.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

06 · Ein Lernspiel entwickeln

1: Ein Lernspiel verbindet fachlich richtige Aufgaben mit verständlicher Rückmeldung, steigendem Anspruch und einer fairen Spielregel.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Vokabelspiel erklärt nach jeder Antwort den Fehler, mischt die Reihenfolge und beendet eine Runde erst nach zehn geprüften Aufgaben. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Der Plan braucht mindestens ein Ereignis, eine wiederholte Prüfung, eine Bedingung, eine Variable und ein überprüfbares Ziel.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

07 · Programmieren: Grundlagen

1: Programmieren übersetzt eine Problemlösung in eindeutige Daten und Anweisungen. Planung, kleine Programmteile, Tests und Überarbeitung gehören zum gesamten Prozess.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Fahrkartenprogramm wird zuerst in Eingaben, Tarifregeln, Berechnung und Ausgabe zerlegt und danach schrittweise umgesetzt. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

08 · Vom Problem zum Programm

1: Vor dem Code werden Ziel, Eingaben, Ausgaben, Regeln und Sonderfälle verstanden. Erst dann entsteht ein Algorithmus und schließlich ein Programm.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Für einen Notenrechner werden zulässige Punkte, Maximalpunktzahl, Rundung und Fehlermeldungen festgelegt, bevor eine Formel geschrieben wird. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

09 · Probleme zerlegen

1: Dekomposition teilt eine große Aufgabe in kleine Teilprobleme mit klaren Schnittstellen. Jedes Teil lässt sich einzeln verstehen und testen.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Quiz zerfällt in Frage anzeigen, Eingabe lesen, Antwort prüfen, Punkte ändern und Ergebnis ausgeben. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

10 · Anforderungen und Testfälle

1: Anforderungen sagen überprüfbar, was ein Programm leisten soll. Testfälle verbinden Eingaben mit erwarteten Ergebnissen und decken Normal-, Rand- und Fehlerfälle ab.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Altersprüfer wird mit 17, 18, 19, leerer Eingabe und Text statt Zahl getestet. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

11 · Pseudocode schreiben

- 1:** Pseudocode beschreibt Programmlogik mit Einrückung und klaren Schlüsselwörtern, ohne an die genaue Schreibweise einer Programmiersprache gebunden zu sein.
- 2:** Ausgangslage und Daten stammen aus diesem Beispiel: WIEDERHOLE für jede Zahl, WENN Zahl gerade, DANN addiere sie zur Summe beschreibt den Kern unabhängig von Python oder Scratch. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.
- 3:** Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.
- 4:** Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

12 · Daten und Steuerung

- 1:** Programme speichern Werte und verändern ihren Ablauf durch Sequenzen, Verzweigungen und Schleifen. Datenzustand und Steuerfluss müssen gemeinsam betrachtet werden.
- 2:** Ausgangslage und Daten stammen aus diesem Beispiel: Ein Kontostand wird als Zahl gespeichert; eine Bedingung verhindert eine Auszahlung, wenn der Betrag nicht ausreicht. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.
- 3:** Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.
- 4:** Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

13 · Variablen und Datentypen

- 1:** Variablen benennen gespeicherte Werte. Datentypen legen fest, welche Werte und Operationen sinnvoll sind, etwa Zahl, Text oder Wahrheitswert.
- 2:** Ausgangslage und Daten stammen aus diesem Beispiel: Die Zeichenfolge '12' wird an '3' zu '123' angehängt, während die Zahlen 12 und 3 zu 15 addiert werden. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.
- 3:** Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.
- 4:** Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

14 · Eingabe und Ausgabe

- 1:** Eingaben liefern Daten von Menschen, Sensoren oder Dateien. Ausgaben machen Ergebnisse sichtbar oder steuern andere Systeme; Validierung trennt gültige von ungültigen Werten.
- 2:** Ausgangslage und Daten stammen aus diesem Beispiel: Ein Temperaturprogramm liest einen Sensorwert, prüft den Messbereich und zeigt Wert oder verständliche Fehlermeldung. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.
- 3:** Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.
- 4:** Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

15 · Bedingungen programmieren

1: Bedingungen ergeben wahr oder falsch und steuern alternative Programmwege. Reihenfolge und vollständige Fälle verhindern widersprüchliche Ergebnisse.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Bei Punkten ab 90 gilt Gold, ab 70 Silber, sonst Bronze; die höchste Grenze wird zuerst geprüft. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

16 · Schleifen auswählen

1: Eine Zählschleife passt zu bekannter Wiederholungszahl, eine Bedingungsschleife zu offenem Ende. Jede Schleife braucht veränderlichen Zustand und erreichbaren Abbruch.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Fünf Messwerte werden mit einer Zählschleife gelesen; eine Passworteingabe läuft bis zur korrekten Eingabe oder maximal drei Versuchen. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

17 · Programme strukturieren

1: Sprechende Namen, kleine Funktionen und klar getrennte Zuständigkeiten machen Programme verständlich, testbar und wiederverwendbar.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Statt einer langen Folge übernehmen `preis_berechnen`, `rabatt_pruefen` und `beleg_ausgeben` jeweils eine Aufgabe. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

18 · Funktionen und Parameter

1: Eine Funktion bündelt einen benannten Ablauf. Parameter liefern veränderliche Eingaben, sodass derselbe Code unterschiedliche Fälle bearbeiten kann.

2: Ausgangslage und Daten stammen aus diesem Beispiel: `flaeche(breite, hoehe)` berechnet Rechtecke verschiedener Größe, ohne die Formel zu kopieren. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

19 · Rückgabewerte verwenden

1: Ein Rückgabewert übermittelt das Ergebnis einer Funktion an die aufrufende Stelle. Ausgabe auf dem Bildschirm und Rückgabe sind unterschiedliche Aufgaben.

2: Ausgangslage und Daten stammen aus diesem Beispiel: `ist_gerade(14)` gibt wahr zurück; erst der aufrufende Programmteil entscheidet, ob daraus eine Meldung entsteht. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

20 · Fehler systematisch suchen

1: Debugging reproduziert einen Fehler, grenzt seine Stelle ein, prüft Annahmen und verändert jeweils nur eine Ursache. Danach laufen alte und neue Tests erneut.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein falscher Punktestand wird mit protokollierten Werten vor und nach der Trefferbedingung auf eine doppelte Erhöhung eingegrenzt. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Zum Beispiel: Punkte und Maximalpunktzahl als Zahlen; Ausgabe Prozentwert; Maximalpunktzahl muss positiv sein; Punkte liegen zwischen 0 und Maximum. Tests: normaler Wert, 0 Punkte, ungültiger Wert über Maximum.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

21 · Robotik

1: Roboter verbinden Mechanik, Elektronik und Programme. Sensoren erfassen die Umwelt, ein Programm entscheidet und Aktoren verändern Bewegung, Licht oder Ton.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Fahrroboter misst den Abstand, bremst unter 15 Zentimetern und dreht die Motoren anschließend in verschiedene Richtungen. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Ein möglicher Ablauf misst fortlaufend, stoppt unter 15 cm, fährt kurz zurück, dreht mit entgegengesetzten Motorrichtungen und misst erneut.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

22 · Sensoren, Verarbeitung und Aktoren

1: Sensoren wandeln physikalische Größen in Daten um. Die Verarbeitung vergleicht Werte mit Regeln; Aktoren setzen Entscheidungen in Handlungen um.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Lichtsensor liefert einen Zahlenwert, das Programm prüft einen Grenzwert und schaltet als Aktor eine LED ein. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Ein möglicher Ablauf misst fortlaufend, stoppt unter 15 cm, fährt kurz zurück, dreht mit entgegengesetzten Motorrichtungen und misst erneut.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

23 · Ozobot mit Farbcodes steuern

1: Ozobot folgt Linien und erkennt definierte Farbfolgen als Befehle. Saubere Linien, passende Abstände und kontrollierte Geschwindigkeit sind Teil des Programms.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Eine Farbfolge vor einer Kreuzung kann die Fahrtrichtung festlegen; ein Testparcours zeigt, ob Farbe und Position zuverlässig gelesen werden. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Ein möglicher Ablauf misst fortlaufend, stoppt unter 15 cm, fährt kurz zurück, dreht mit entgegengesetzten Motorrichtungen und misst erneut.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

24 · Calliope mini programmieren

1: Der Calliope mini kombiniert Tasten, Sensoren, LEDs, Funk und Anschlüsse auf einem Mikrocontroller. Programme reagieren auf Ereignisse und Messwerte.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Beim Drücken von Taste A zeigt die LED-Matrix ein Symbol; bei hoher Lautstärke wechselt es zu einer Warnanzeige. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Ein möglicher Ablauf misst fortlaufend, stoppt unter 15 cm, fährt kurz zurück, dreht mit entgegengesetzten Motorrichtungen und misst erneut.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

25 · LEGO-Roboter programmieren

1: Ein LEGO-Roboter steuert Motoren über Leistung, Richtung und Dauer und nutzt Sensorwerte für Rückmeldung. Bauweise und Programm beeinflussen gemeinsam das Fahrverhalten.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Abstandssensor löst Bremsen aus; ungleiche Räder oder Motorleistungen müssen beim Geradeausfahren berücksichtigt werden. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Ein möglicher Ablauf misst fortlaufend, stoppt unter 15 cm, fährt kurz zurück, dreht mit entgegengesetzten Motorrichtungen und misst erneut.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

26 · Open Roberta im Simulator

1: Open Roberta stellt Programme mit NEPO-Blöcken dar und kann viele Abläufe ohne Hardware simulieren. Simulation prüft Logik, ersetzt aber nicht jeden realen Sensorfehler.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein virtueller Roboter folgt einer Linie; danach wird geprüft, welche Werte und Geschwindigkeiten am echten Gerät angepasst werden müssen. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Ein möglicher Ablauf misst fortlaufend, stoppt unter 15 cm, fährt kurz zurück, dreht mit entgegengesetzten Motorrichtungen und misst erneut.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

27 · Ein Robotikprojekt planen

1: Ein Robotikprojekt braucht Aufgabe, Umweltannahmen, Sensoren, Aktoren, Zustände, Sicherheitsgrenzen und messbare Tests.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein Rettungsroboter soll einen markierten Bereich finden; Testfälle prüfen Kurven, verlorene Linie, Hindernis und leere Batterie. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Ein möglicher Ablauf misst fortlaufend, stoppt unter 15 cm, fährt kurz zurück, dreht mit entgegengesetzten Motorrichtungen und misst erneut.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

28 · Cybermobbing

1: Cybermobbing ist wiederholte digitale Herabsetzung, Ausgrenzung oder Bedrohung mit Machtungleichgewicht. Reichweite, Kopierbarkeit und ständige Erreichbarkeit verstärken die Wirkung.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein peinliches Bild wird trotz Bitte weiterverbreitet und mit abwertenden Kommentaren in mehreren Gruppen geteilt. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Nicht mitmachen, relevante Belege sichern, betroffene Person unterstützen, blockieren/melden und Lehrkraft, Schulsozialarbeit oder Eltern einbeziehen; bei Bedrohung sofort Erwachsene und gegebenenfalls Polizei.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

29 · Cybermobbing erkennen

1: Unterschieden werden einmaliger Streit, Konflikt und systematisches Mobbing. Wiederholung, Absicht, Wirkung, Öffentlichkeit und Machtverhältnis liefern Hinweise.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Ein einzelner unhöflicher Kommentar ist ernst zu nehmen, eine wochenlange koordinierte Ausgrenzung erfüllt zusätzlich typische Mobbingmerkmale. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Nicht mitmachen, relevante Belege sichern, betroffene Person unterstützen, blockieren/melden und Lehrkraft, Schulsozialarbeit oder Eltern einbeziehen; bei Bedrohung sofort Erwachsene und gegebenenfalls Polizei.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

30 · Rollen und Gruppendynamik

1: Neben betroffener und angreifender Person beeinflussen Mitmachende, Zuschauende, Unterstützende und Moderierende den Verlauf. Schweigen kann als Zustimmung wirken.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Eine Person leitet nicht weiter, widerspricht sachlich, sichert Hilfe und verändert damit die Gruppennorm. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Nicht mitmachen, relevante Belege sichern, betroffene Person unterstützen, blockieren/melden und Lehrkraft, Schulsozialarbeit oder Eltern einbeziehen; bei Bedrohung sofort Erwachsene und gegebenenfalls Polizei.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

31 · Beweise sichern und Vorfälle melden

1: Vor dem Blockieren oder Löschen werden relevante Inhalte mit Datum, Kontext, Nutzernamen und URL gesichert. Meldung erfolgt an Plattform und vertrauenswürdige Erwachsene.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Screenshots zeigen nicht nur die Beleidigung, sondern auch Verlauf und Zeitpunkt; Passwörter und private Daten werden dabei nicht weiterverteilt. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Nicht mitmachen, relevante Belege sichern, betroffene Person unterstützen, blockieren/melden und Lehrkraft, Schulsozialarbeit oder Eltern einbeziehen; bei Bedrohung sofort Erwachsene und gegebenenfalls Polizei.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

32 · Betroffene wirksam unterstützen

1: Unterstützung nimmt die Erfahrung ernst, vermeidet Schuldzuweisung, stellt Sicherheit her und bezieht geeignete Vertrauenspersonen ein.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Statt 'geh einfach offline' fragt eine Mitschülerin, was gebraucht wird, begleitet zur Lehrkraft und hilft beim Melden. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Nicht mitmachen, relevante Belege sichern, betroffene Person unterstützen, blockieren/melden und Lehrkraft, Schulsozialarbeit oder Eltern einbeziehen; bei Bedrohung sofort Erwachsene und gegebenenfalls Polizei.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Lösungshinweise

33 · Klassenregeln für digitale Räume

1: Wirksame Regeln benennen erlaubtes Verhalten, Grenzen, Hilfewege und nachvollziehbare Folgen. Sie werden gemeinsam geübt und regelmäßig überprüft.

2: Ausgangslage und Daten stammen aus diesem Beispiel: Die Klasse vereinbart: keine Screenshots ohne Zustimmung, Konflikte nicht weiterleiten, Vorfälle sichern und an zwei benannte Ansprechpersonen melden. Die Verarbeitung wird mit dem Kerngedanken der Seite begründet; das Ergebnis muss beobachtbar sein.

3: Nicht mitmachen, relevante Belege sichern, betroffene Person unterstützen, blockieren/melden und Lehrkraft, Schulsozialarbeit oder Eltern einbeziehen; bei Bedrohung sofort Erwachsene und gegebenenfalls Polizei.

4: Individuelle Lösung. Bewertbar sind eine einzige gezielt veränderte Voraussetzung, eine begründete Vorhersage und ein reproduzierbarer Test.

Quellen und Rechtliches

Alle Erklärtexpte, Aufgaben, Projekte und Lösungshinweise dieses Hefts: SchlauStart Eigenmaterial, 2026.

Weiterführende offene Materialien werden auf den zugehörigen SchlauStart-Themenseiten mit Urheber, Quelle und Lizenz einzeln ausgewiesen. Verlinkte Angebote sind nicht Bestandteil dieses Hefts.

Scratch ist eine Marke der Scratch Foundation. Calliope mini, LEGO, Ozobot, Python, Open Roberta und MDN werden ausschließlich zur sachlichen Benennung der jeweiligen Lernumgebung verwendet.