


# Variablen

**Variablen** sind interne Spielwerte – für Spielende **unsichtbar** – mit denen du Logik, Zustände und Verzweigungen steuerst (Flags, Zähler, Punkte, Fortschritt).



## Anlegen

**Name** – frei wählbar (nur Buchstaben/Zahlen/\_), z. B. `Szene_gesehen`, `punkte`, `tresor_fehlversuche`.  
**Anzahl** – Startwert (Ganzzahl, Standard 0/1).

Verwende `0/1` als **Boolean-Flag**: `Szene_gesehen = 1` bedeutet „Intro schon gezeigt“.

## Verhalten & Reichweite

- Gilt **projektweit** für den aktuellen Durchlauf/Spielstand.
- Für Spielende **nicht sichtbar** (nur für Bedingungen & Aktionen nutzbar).
- Bleibt erhalten, bis du sie änderst oder den Spielstand zurücksetzt.

## Nutzung in Bedingungen

Du kannst Variablen an jeder Stelle abfragen, um Elemente/Wege zu steuern.  
Unterstützte Vergleiche (typisch):

- `==` gleich `!=` ungleich
- `>` größer `<` kleiner
- `>=` mindestens `<=` höchstens

## Beispiele

- Intro überspringen: nur anzeigen, **wenn** `Szene_gesehen == 0`.
  - Bonus freischalten: Button zeigen, **wenn** `punkte >= 50`.
  - Sicherheitslogik: Hinweis zeigen, **wenn** `tresor_fehlversuche >= 3`.
- 

# Variablen ändern (Aktionen)

- **Erhöhen/Verringern:** `punkte += 10`, `tresor_fehlversuche += 1`.
  - **Setzen:** `Szene_gesehen = 1` (Flag aktivieren), `timer = 0` (reset).
  - **Beispiel-Workflow:** Beim Szenenstart `Szene_gesehen = 1`; beim richtigen Code `punkte += 20`; beim falschen Code `tresor_fehlversuche += 1`.
- 

# Muster & Best Practices

- **Benennung:** klein\_schreiben\_mit\_unterstrich (eindeutig & sprechend).
  - **Flags statt Text:** nutze Zahlen (0/1) statt Strings – schneller & fehlerarm.
  - **Initialisierung:** setze Startwerte bewusst zu Beginn des Spiels/Levels.
  - **Sicherheit:** prüfe vor dem Abziehen auf Grenzwerte (z. B. nicht unter 0).
  - **Dokumentation:** kurze Tabelle deiner Variablen (Name, Bedeutung, Startwert).
- 

# Mini-Beispiele

- **Intro-Once**
  - Bedingung: „Intro-Dialog nur, wenn `intro_seen == 0`“
  - Aktion beim Anzeigen: `intro_seen = 1`
- **Punktesystem**
  - Rätsel gelöst: `punkte += 10`
  - Bonuspfad: sichtbar, wenn `punkte >= 30`
- **Fehlversuche**
  - Bei falscher Eingabe: `fails += 1`
  - Zeige Hilfe, wenn `fails >= 2`

Mit Variablen baust du vom simplen „Schon gesehen?“ bis zur komplexen Quest-Logik alles sauber und wiederverwendbar.

---

Revision #3

Created 27 September 2025 14:47:40 by David Wright

Updated 27 September 2025 14:49:52 by David Wright