Codewars Challenge-Katalog

1. **Challenge Name:** Remove the Noise

**Description:**  
You are given a string that may contain random "noise" words: "uh", "erm", and "like". Your job is to remove all of these noise words from the string, but preserve the spacing and punctuation of the remaining content.

The function should return the cleaned-up version of the string with extra spaces removed (i.e., no double spaces), and trim leading/trailing spaces.

Ein Bild, das Text, Screenshot, Software, Multimedia-Software enthält.

KI-generierte Inhalte können fehlerhaft sein.

2. **Challenge:** Is the string uppercase?

**Aufgabe:**  
Schreibe eine Funktion isUpperCase, die prüft, ob **alle Buchstaben** in einem gegebenen String **Großbuchstaben** sind.  
Nicht-Buchstaben (z. B. Zahlen, Leerzeichen, Satzzeichen) sollen **ignoriert** werden.

Ein Bild, das Text, Screenshot, Schrift, Display enthält.

KI-generierte Inhalte können fehlerhaft sein.

3. **Challenge:** Fake Binary

**📋 Aufgabe:**

Schreibe eine Funktion fakeBin(x), die einen String aus Ziffern (0–9) entgegennimmt. Ersetze jede Ziffer wie folgt:

* Wenn die Ziffer kleiner als 5 ist → ersetze sie durch "0"
* Wenn die Ziffer 5 oder größer ist → ersetze sie durch "1"

🔁 Gib den neuen String zurück.

Ein Bild, das Text, Screenshot, Schrift enthält.

KI-generierte Inhalte können fehlerhaft sein.

4. Challenge: Mumbling

**Beschreibung:**

Dieser Kata verlangt von dir, einen bestimmten Text umzuwandeln.

Beispiel:  
Eingabe: "abcd"  
Ausgabe: "A-Bb-Ccc-Dddd"

Jeder Buchstabe der Eingabe wird so transformiert:

* Der erste Buchstabe jeder Gruppe wird **großgeschrieben**.
* Danach folgen mehrere **kleingeschriebene Wiederholungen** desselben Buchstabens – so oft, wie seine Position im String ist (beginnend bei 1).
* Die Gruppen werden durch **Bindestriche** getrennt.



5. Challenge: Vowel Count

**Aufgabe:**

Schreibe eine Funktion, die die Anzahl der Vokale in einem gegebenen String zählt.  
Für diese Aufgabe gelten nur **"a", "e", "i", "o", "u"** als Vokale (nur Kleinbuchstaben).  
Der Eingabestring besteht **nur aus Kleinbuchstaben und/oder Leerzeichen**.

Beispiel:

getCount("hello world") // => 3

getCount("abracadabra") // => 5

getCount("") // => 0

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

KI-generierte Inhalte können fehlerhaft sein.

6. Challenge: Alternating Case Splitter

**📝 Beschreibung:**

Schreibe eine Funktion alternateCaseSplit(str), die einen String annimmt und ihn in ein Array aufteilt, wobei jeder Teil abwechselnd **in Klein- und Großbuchstaben** formatiert ist.

Der ursprüngliche String besteht nur aus Buchstaben und Leerzeichen. Die Funktion soll den String anhand der Leerzeichen aufteilen, und dann:

* **Jedes ungerade Wort** (1., 3., 5., ...) komplett in **Kleinbuchstaben**
* **Jedes gerade Wort** (2., 4., 6., ...) komplett in **Großbuchstaben**

**📥 Eingabe:**

* Ein string bestehend aus alphabetischen Zeichen und Leerzeichen, keine Sonderzeichen oder Ziffern.
* Es gibt **keine** führenden oder mehrfachen Leerzeichen.

**📤 Ausgabe:**

* Ein Array von Strings mit abwechselnder Formatierung (klein, groß, klein, ...)

**🔁 Beispiele:**

alternateCaseSplit("Hello world from codewars")

// ➞ ["hello", "WORLD", "from", "CODEWARS"]

alternateCaseSplit("JavaScript is awesome")

// ➞ ["javascript", "IS", "awesome"]

alternateCaseSplit("test")

// ➞ ["test"]

Ein Bild, das Text, Screenshot, Schrift enthält.

KI-generierte Inhalte können fehlerhaft sein.