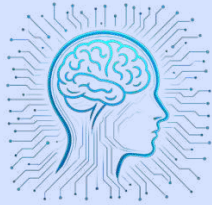


9 – while-Schleife in PHP

Übersicht



Was lerne ich hier?

- Wie funktioniert eine while-Schleife in PHP?
- Was ist der Unterschied zwischen while- und for-Schleife?
- Was ist die Abbruchbedingung?

Was du können musst:

- Du erkennst, wann du eine while-Schleife verwenden musst.
- Du kannst while-Schleifen programmieren.
- Du kannst erklären, was eine Abbruchbedingung ist.

Du musst nach der Bearbeitung die Aufgabe unten selbstständig lösen können!

Informiere dich mit einem der folgenden Lernwege. Du kannst die Lernwege auch wechseln oder kombinieren. Wichtig ist, dass du die oben genannten Themen selbstständig anwenden kannst – **siehe Aufgabenstellungen ganz unten im Dokument!**

Erklärvideo	Informationen	Lehrer	KI
https://tinyurl.com/winif13php - Kapitel 9 – while-Schleife in PHP	siehe Infoblatt: 09-while-schleife-2-info.pdf	Frag deinen Lehrer	Frag die KI – Hinweise siehe unten!

Frag die KI

Die KI (z.B. duck.ai) kann dir wie immer all das erklären. Ziel ist jedoch, dass du dir die obigen Themen so erklären lässt, dass du in der Lage bist, while-Schleifen zu programmieren und sie anzuwenden.

Aufgaben

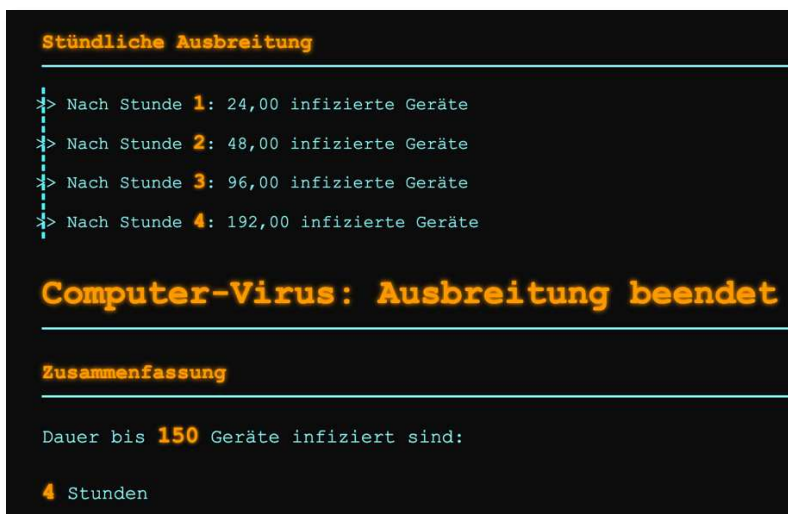
Dein Job für Thema 9 (while-Schleifen)

Du hast das Thema dann vollständig bearbeitet, wenn du die Aufgaben selbstständig lösen kannst!

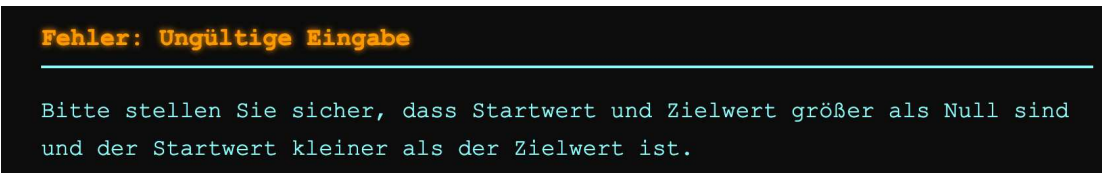
1. Ein neuer Computer-Virus breitet sich weltweit aus. Pro Stunde *verdoppelt* sich die Zahl der infizierten Geräte. Du sollst diese Entwicklung mit PHP simulieren.

Ein Formular (*1-virus.html*) liegt vor, ebenso das Grundgerüst für die Auswertungsseite (*1-virus-auswertung.php*).

Erstelle in *1-virus-auswertung.php* ein Programm in PHP, dass berechnet, wie lange es dauert, bis eine gesuchte Anzahl an Geräten infiziert ist.



- 1b. Erweitere das Programm aus 1 so, dass eine Fehlermeldung ausgegeben wird, falls eine sinnlose Eingabe gemacht wird, nämlich wenn ...
... die Anzahl der infizierten Geräte größer oder gleich ist wie die Anzahl der gesuchten Geräte;
... die Anzahl der gesuchten Geräte 0 oder kleiner ist;
... die Anzahl der infizierten Geräte 0 oder kleiner ist.



2.

Exkurs: Geo-Engineering

Geo-Engineering sind große menschliche Versuche, das Klima zu beeinflussen. Hier sind zwei Hauptarten:

- **Stratosphärische Aerosol-Injektionen:** Wir könnten winzige Teilchen in die obere Atmosphäre schießen, um Sonnenlicht zu blockieren und die Erde kühler zu machen.
- **Direkte Luftabscheidung:** Wir könnten spezielle Technologien verwenden, um CO₂ direkt aus der Luft zu holen.

Hier ist der Haken: Diese Ideen haben viele Risiken, und wir wissen nicht genau, wie sie sich auf die Natur auswirken könnten. Deshalb ist es immer noch viel wichtiger, weniger Treibhausgase zu erzeugen und auf saubere Energie umzusteigen, um den Klimawandel zu stoppen. Geoengineering ist sozusagen der "Plan B", falls wir andere Lösungen nicht rechtzeitig umsetzen können.

Mit einer neuartigen Technik aus dem Bereich des Geo-Engineering kann die durchschnittliche Temperatur der Erde jährlich um 1% gesenkt werden. Entwirf eine Simulation, die berechnet, wie lange es dauert, bis durch die Senkung eine bestimmte Zieltemperatur erreicht ist.



Geoengineering Simulator

Bitte geben Sie die gewünschten Parameter für die Temperaturmodellierung ein:

Aktuelle Durchschnittstemperatur:

Absenkung der Temperatur in %:

Ziel-Temperatur:

BERECHNEN

Simulationsergebnisse: Temperaturtrend

Start: 14 °C, Reduktion pro Jahr: 1 % (ca. 0,140 °C), Ziel: 13 °C

Errechneter Verlauf

- »Nach Jahr 1: 13,86 Grad Celsius
- »Nach Jahr 2: 13,72 Grad Celsius
- »Nach Jahr 3: 13,58 Grad Celsius
- »Nach Jahr 4: 13,44 Grad Celsius
- »Nach Jahr 5: 13,30 Grad Celsius
- »Nach Jahr 6: 13,16 Grad Celsius
- »Nach Jahr 7: 13,02 Grad Celsius
- »Nach Jahr 8: 12,88 Grad Celsius

Die Ziel-Temperatur von 13 °C wird nach 8 Jahren erreicht.

Zurück

»Zur Simulation

Vervollständige die Dateien *2-geo.html* und *2-geo-auswertung.php*

- 3 Ein Server wird mit einer Klimaanlage gekühlt. Durch ein Leck tritt aus der Klimaanlage Kühlmittel aus. Berechne, wie lange es dauert, bis ein kritischer Füllstand in der Klimaanlage erreicht ist.

Vervollständige die Dateien 3-klimaanlage.html und 3-klimaanlage-auswertung.php entsprechend der folgenden Screenshots:

Klimaanlage



Kühlmittel in Liter:

Verlust in %:

Kritischer Füllstand in Liter:

Klimaanlage - Ergebnis

Verlust über die Zeit

Der kritische Füllstand beträgt: 35,00 Liter.

Stunde	Kühlmittel (Liter)
1	47,50
2	45,13
3	42,87
4	40,73
5	38,69
6	36,75
7	34,92

Nach 7 Stunden wurde der kritische Füllstand unterschritten.

[Neue Berechnung starten](#)

Das **Ergebnis wird als Tabelle ausgegeben**. Letztlich funktioniert das genau so wie eine Liste:

- Drumrum die `<table>`- und `</table>`-Tags
- Um jede Zeile die `<tr>`- und `</tr>`-Tags
- Um jede Zelle ein `<td>...</td>`
- Wenn Sie stark sind: Die Kopfzeile bekommt keine `<td>`, sondern `<th>`

Erstelle außerdem ein ordentliches CSS, das sieht ja fürchterlich aus!

Übungsaufgaben

Mit diesen Aufgaben kannst du das Thema weiter üben und vertiefen.

- 2b.** Erweitere das Programm aus 2 („Geo-Engineering“) so:
- Im Formular wird ein Startjahr eingegeben (etwa: 2026). In der Ausgabe werden „echte“ Jahreszahlen ausgegeben, also „Nach Jahr 2028: xy Grad Celsius“
 - Wenn die Ergebnisliste 50 Jahre überschreitet, wird die Ausgabe abgebrochen und die Meldung »Prognose über weitere Jahre zu ungenau.« ausgegeben.

- 3b** Erweitere das Programm aus 3 („Klimaanlage“) so:

Unter einem Füllstand von **40 Litern halbiert** sich der ursprüngliche **Verlust** an Kühlmittel (wenn also z.B. 5% in das Textfeld eingegeben wurde, sinkt der Verlust auf 2,5%). Der aktuelle Verlust soll in einer eigenen Tabellenspalte ausgegeben werden:

Verlust über die Zeit

Stunde	Akt. Verlust (%)	Kühlmittel (Liter)
--------	------------------	--------------------

1	5,00	47,50
2	5,00	45,13
3	5,00	42,87
4	5,00	40,73
5	5,00	38,69
6	2,50	37,72
7	2,50	36,78
8	2,50	35,86
9	2,50	34,96

Nach 9 Stunden wurde der kritische Füllstand von 35,00 Litern unterschritten.

4

Erstelle ein einfaches **Würfelspiel**, das die folgenden Regeln in PHP umsetzt:

- Der Spieler würfelt so lange, bis er eine 6 würfelt. Das Spiel endet, sobald eine 6 gewürfelt wird.
- Die gewürfelten Augen werden addiert.
- Der Spieler gewinnt, wenn die Gesamtpunktzahl am Ende über 20 liegt.

Das Programm soll die einzelnen Würfelergebnisse und die Endpunktzahl in einer Tabelle ausgeben.

Hinweis: Einen zufälligen Würfelwurf erzeugen Sie mit

```
$letzter_wurf = rand(1, 6);
```

Das Formular zum Starten liegt bereits vor als *4-wuerfelzocker.html*:

Glücksspiel: Würfelglück

Würfle so lange, bis eine 6 kommt. Ziel: Mehr als 20 Punkte!

WÜRFEL WERFEN!

Ein Teil der Auswertungsseite liegt ebenfalls vor als *4-wuerfelzocker-ergebnis.php*.

Spiel beendet!

Ergebnisse

Wurf-Nr.	Gewürfelte Augen	Aktuelle Gesamtpunktzahl
1	4	4
2	4	8
3	6	14

✗ ****VERLOREN!**** Nur 14 Punkte erreicht. Das reicht nicht für den Gewinn.

NOCHMAL WÜRFELN!

Hinweis:

- Der Ergebnistext im unteren Kasten ist ein `<p>` mit der Klasse *ergebnis*.