

Praktikumsaufgaben für VW7

Programmierung interaktiver Systeme im Wintersemester 2023/24
Prof. Dr. Martin Weigel

[Gotta Catch 'Em All!](#) • [Molekül-Fabrik mit Semaphoren](#)

1 Gotta Catch 'Em All!

Der folgende Processing-Sketch lädt alle 151-Pokemon von der [Pokemon.com](#) Webseite und stellt diese in einem Grid dar. Alle Bilder werden dafür sequentiell (also nacheinander) in der `setup()`-Methode geladen. Dies dauert sehr lange, weswegen der Processing-Sketch meist erst nach mehreren Sekunden korrekt gezeichnet wird. Schreiben Sie den Sketch um, sodass die Anwendung sofort starten kann.

Starten Sie dafür **mehrere Threads** in der `setup()`-Methode ohne die `setup()`-Methode zu blockieren. Jeder Thread soll einen Start- und einen Endindex bekommen und nur die Bilder in diesem Bereich laden. Nutzen Sie eine Konstante, um die Anzahl der Threads festzulegen. Alle Threads sollen in etwa die gleiche Anzahl an Bildern laden.

Prüfen Sie in der `draw()`-Methode, welche der Bilder bereits geladen wurden. Die geladenen Bilder sollen in der Processing App dargestellt werden. Noch nicht geladene Bilder sollen durch einen gezeichneten Platzhalter dargestellt werden (z.B. ein X).

Basissketch: <https://gist.github.com/MartinWeigel/bf56c8546a231c90dd812e692f32499c>



Erweitern Sie den Sketch. Sie müssen hierfür keine MVC-Anwendung schreiben.

2 Molekül-Fabrik mit Semaphoren

Semaphoren können als Signal-Mechanismus zwischen mehreren Threads genutzt werden. Die Methoden `release()` und `acquire()` müssen dafür nicht zwangsläufig in der gleichen Methode/Klasse aufgerufen werden. Ihre Aufgabe ist es vier Thread-Klassen zu erstellen:

1. `WasserstoffGenerator` generiert in einem zufälligen Abstand ein Wasserstoff-Atom (H). Der maximale Abstand wird durch den Kontruktor gesetzt.
2. `SauerstoffGenerator` generiert in einem zufälligen Abstand ein Sauerstoff-Atom (O). Der maximale Abstand wird durch den Kontruktor gesetzt.
3. `WasserGenerator` generiert in einem — im Konstruktor übergebenen — festgelegten Abstand ein Wasser-Molekül (H₂O). Dazu benötigt der Generator aber zwei Wasserstoff-Atome und ein Sauerstoff-Atom. Ansonsten wartet er, bis diese Ressourcen existieren.
4. `OzonGenerator` generiert in einem — im Konstruktor übergebenen — festgelegten Abstand ein Ozon-Molekül (O₃). Dazu benötigt der Generator drei Sauerstoff-Atome. Ansonsten wartet er, bis diese Ressourcen existieren.

Nutzen Sie eine Semaphore für jedes Atom/Molekül, um einen Überblick über die generierten Ressourcen zu bekommen. Starten Sie jeweils einen Thread für die Generatoren.

Schreiben Sie einen Processing-Sketch mit dem Sie die von den Generatoren erstellten Mengen an Atomen und Molekülen anzeigen.

Lesen Sie die [Dokumentation zu Semaphoren](#), um weitere Methoden von Semaphoren und deren Parameter kennenzulernen.