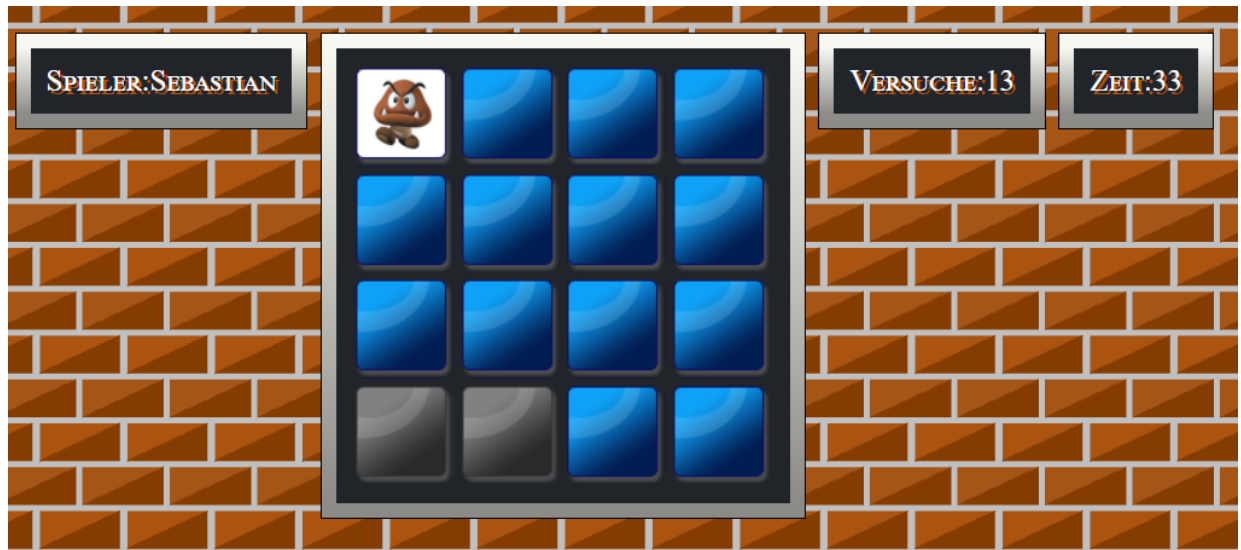


Hausübung 2 – JavaScript – Memory Spiel

Ziel dieser Aufgabe ist es ein simples Memory Spiel zu programmieren

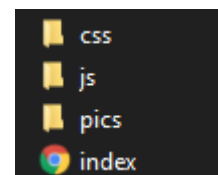
Grundlegende Spielidee: Sie sehen 16 Karten, die verdeckt sind (blaues Bild wird angezeigt). Sobald Sie auf eine Karte klicken, wird diese „umgedreht“ (das jeweils zugewiesene Bild wird angezeigt (card1.png- card16.png). Es können maximal zwei Karten gleichzeitig aufgedeckt werden. Wenn die zweite Karte geöffnet wird, wird Sie auf jeden Fall für eine gewisse Zeit angezeigt, bevor beide Karten wieder verdeckt werden falls das Paar nicht zusammenpasst. Falls das Paar zusammenpasst werden die Hintergrundbilder des Paares auf „memoryBG1.png“ gesetzt und die Javascript Funktionalität (dass sie anklickbar sind) wird entfernt.



1. Laden Sie sich das Template von Moodle herunter (beinhaltet HTML Datei und CSS Datei)
(Optional können Sie Ihr eigenes Design implementieren und das Template nicht verwenden)

Sie finden folgende Ordnerstruktur:

- Im Ordner „pics“ befinden sich Bilder für das Spiel, diese können bei Bedarf ausgetauscht werden
 - Im Ordner „css“ befindet sich ein CSS File, dieses kann bei Bedarf verändert/erweitert werden
 - Der Ordner „js“ ist leer, hier muss eine Javascript Datei erstellt werden, die in weiterer Folge die vollständige Programmlogik abbildet (kein Javascript in der index.html Datei)
2. Lesen Sie sich den Code in der „index.html“ durch und erweitern Sie gegebenenfalls mehrere „div“ Elemente um „ids“, um für Javascript einfache Referenzen zur Verfügung zu haben

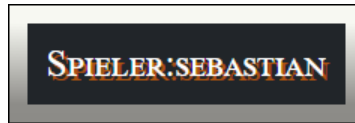


```
<div class="infobox">
  <div class="labels">
    Spieler:
  </div>
  <div class="labels">

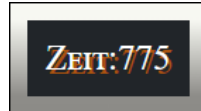
  </div>
</div>
```

Hausübung 2 – JavaScript – Memory Spiel

3. Fordern Sie den User beim Laden der Seite zur Eingabe seines Namens auf und stellen Sie den Namen in dem dafür vorbereiteten „div“ Element dar.



4. Implementieren Sie eine Funktion, die die Sekunden solange mitzählt, bis das Spiel erfolgreich beendet wird und stellen Sie die Sekunden in dem dafür vorbereiteten „div“ Element dar. (Einmal pro Sekunde soll der Wert höhergezählt werden)



5. Implementieren Sie eine Funktion, die die Versuche mitzählt (ein Versuch entspricht dem Aufdecken von zwei Karten) und stellen sie die Versuche in dem dafür vorbereiteten „div“ Element dar



6. Erstellen sie 16 Spielkarten via Javascript (neue Knoten generieren) und definieren Sie notwendige Attribute (z.B.: Klassenname, onclick Event, usw.) per Javascript
7. Sammeln Sie die erstellten Knoten in einem Array
8. Mischen Sie das Array damit die Karten bei jedem Versuch an einer anderen Position dargestellt werden. (deaktivieren Sie die Mischfunktion während des Testens, um Zeit zu sparen)
9. Fügen Sie die Karten in den dafür vorgesehenen „div“ Bereich (Spielbereich) ein. (Wenn Sie das Layout mit „float“ erstellen wollen, muss bei jeder 4ten Karte per Javascript das floaten „gecleart“ werden, damit die nächste Reihe mit 4 Karten erstellt werden kann) Sie können natürlich auch andere Varianten verwenden um das Layout zu gestalten.
10. Achten Sie darauf, dass bereits gefunden Karten nicht erneut angeklickt werden können.