HBo Tampermonkey Framework

# Datei- und Verzeichnisstruktur

Für jede Website existiert ein Unterverzeichnis in src/funke oder src/hartmut. Ein Projektverzeichnis enthält die folgenden Dateien:

* main.js  
  Der Inhalt dieser Datei wird im Tampermonkey-Plugin direkt eingefügt. Alle weiteren Dateien sind hier referenziert und werden automatisch geladen und aktualisiert.
* styles.css  
  die Styles werden automatisch in die Seite eingefügt
* UI.js  
  Hier werden alle UI Erweiterungen, wie z.B. Key-Handler oder Vorlesefunktionien registriert.
* Weitere Dateien werden nach Bedarf angelegt und in main.js referenziert.

# main.js

Bei neuen Projekten, die gelb markierten Texte ersetzen.

// ==UserScript==

// @name Projektname

// @version 1

// @description Beschreibung

// @grant GM\_getResourceURL

// @grant GM\_getResourceText

// @grant GM\_info

// @grant addStyle

// @grant unsafeWindow

// @include URL der Website

// @downloadURL http://localhost:3000//<Projektpfad>/main.js

// @updateURL http://localhost:3000//<Projektpfad>/main.js

// @resource css-common http://localhost:3000/common/styles.css

// @resource css http://localhost:3000//<Projektpfad>/styles.css

// @require http://localhost:3000/common/head.js

// @require http://localhost:3000/common/styles.js

// @require http://localhost:3000/common/KeyHandler.js

// @require http://localhost:3000/common/readout.js

// @require http://localhost:3000/<Projektpfad>/UI.js

// ==/UserScript==

console.log('initialized', 'Version '+GM\_info.script.version);

console.groupEnd();

# Globale Module

## Readout

### Registrierung

window.registerForReadOut(selectors:String);

window.registerForReadOut(selectors:String, options:Object);

window.registerForReadOut(url-pattern:RegExp, selectors:String);

window.registerForReadOut(url-pattern:RegExp, selectors:String, options:Object);

window.registerForReadOut(url-wildcard:String, selectors:String, options:Object);

### Optionen

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Name | | | Typ | | Defaultwert | | Beschreibung |
| pitch | | | float | 0,8 | | Tonhöhe | |
| rate | | | Float | 1,35 | | Sprechgeschwindigkeit | |
| volume | | | Float | 0,8 | | Lautstärke | |
| language | | | string | de-DE | | Sprache | |
| prefix | | | string | <leer> | | Vorangestellter Text | |
| suffix | | | string | <leer> | | Nachfolgender Text | |
| immediate | | | boolean | False | | Löscht die aktuelle Queue und spricht den Text sofort. | |
| readHidden | | | boolean | false | | Versteckte oder unsichtbare Elemente vorlesen | |
| exclude | | | string | Array | "" | | Selector(en) für zu ignorierende Unterelemente | |
| extract | | | function | null | | Callback-Funktion, die den Text des Elements zurückliefert. Syntax: extract(node)  Ergebnis:   * String * Array aus Strings * {text,pitch,rate,volume} * [{text,pitch,rate,volume}, …] * Node * [Node, …] * NodeList   Wenn die Funktion mit der Syntax  function(node){…}  Definiert wird, stehen innerhalb der Funktion die Methoden  this.default(node)  und die Eigenschaft  this.format  zur Verfügung. | |
| replace | | | object | Array | null | | Textersetzungen. Der Parameter enthält entweder ein Object mit den Attributen pattern und replacement, oder ein Array mit entsprechenden Objekten. Siehe String.replace() | |
|  | pattern | | string | RegularExpression |  | |
|  | replacement | | string | function |  | |
|  | | |  |  | |  | |
| childElements | | | object | null | | Optionen für bestimmte Unterelemente. Die Optionen für ein Unterelement werden mit dem CSS-Selector als Key angegeben. | |
|  | rate | | float |  | |  | |
|  | pitch | | float |  | |  | |
|  | volume | | float |  | |  | |
|  | language | | String |  | |  | |
|  | prefix | | string |  | |  | |
|  | suffix | | string |  | |  | |
|  | text | | string |  | |  | |
|  | extract | | function |  | | s. oben | |
|  | replace | | object | Array |  | | Textersetzungen. Der Parameter enthält entweder ein Object mit den Attributen pattern und replacement, oder ein Array mit ebensolchen Objekten. Siehe String.replace() | |
|  | | pattern | string | RegularExpression |  | |
|  | | replacement | string | function |  | |
|  |  | |  |  | |  | |

## Mutation Handler

### Registrieerung

Window.onMutation('selector', reaction);

Window.onMutation({'selector1':reaction1, 'selector2': reaction2, …});

### Reaction Object

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Typ | Defaultwert | Beschreibung |
| selector | String |  | CSS-Selector |
| filter | Function |  | Filterfunkiton zur weiteren Einschränkung der betroffenen Elemente |
| className | string |  | Klassenliste, wird zur bestehenden Klassenliste hinzugefügt |
| style | string |  | Styles |
| callback | Function |  | Callback(element,index,array) |
| listeners | Map |  | Event-Listener. Der Parameter enthält eine Liste mit <eventName>:<handleFunction> |
| runOnLoad | Boolean | false | Wenn true, wird die Reaktion nach dem Laden der Seite sofort ausgeführt |
| maxRecursions | Integer | 0 | Maximale Anzahl rekursiver Aufrufe |

# Styles

## Globale Style Variablen

|  |  |  |
| --- | --- | --- |
| Name | Verwendung | Bemerkungen |
| --button-overlay |  | Benötigt eine Hintergrundfarbe |
| --button-soft-overlay |  |
| --row-overlay |  |
| --row-soft-overlay |  |
| Farben |  |  |
| --ok-button |  |  |
| --ok-border |  |  |
| --cancel-button |  |  |
| --cancel-border |  |  |
| --neutral-button |  |  |
| --neutral-border |  |  |
| --neutral2-button |  |  |
| --neutral2-border |  |  |
| --attention-button |  |  |
| --attention-border |  |  |
| --danger-button |  |  |
| --danger-border |  |  |
| --disabled-button |  |  |
| --disabled-border |  |  |

## Style Mixins

Styles werden gemischt, indem sie mit der Syntax

@<style-name>

@<style-name> !important

Ineinander verschachtelt werden.

Im finalen Stylesheet werden diese Elemente expandiert und ggf. mit der Markierung !important versehen.

Doppelte Style-Anweisungen werden, entsprechend der Reihenfolge, überschrieben oder überschreiben die Anweisungen aus dem Mixin.

### Beispiel:

1. button, input[type="submit"] {
2. border: 1px solid blue;
3. }
4. .left {
5. Float: left;
6. }
7. .merged {
8. @button;
9. @.left;
10. Border-width: 2px;
11. }

### Global definierte Styles

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| .ok-button |  |  |  |
| .cancel-button |  |  |  |
| .neutral-button |  |  |  |
| .neutral2-button |  |  |  |
| .attention-button |  |  |  |
| .danger-button |  |  |  |
| .disabled-button |  |  |  |
| input-highlight |  |  |  |
| hbo-icon |  |  |  |
| hbo-icon-alert |  |  |  |
| hbo-icon-info |  |  |  |
|  |  |  |  |
|  |  |  |  |