# WEB ACADEMY

Front-end за Напреднали

Какво е API?

- API - съкратено от **A**pplication **P**rogram **I**nterface

- набор от правила, протоколи и методи, за изграждане на софтуерни. Чрез API различни програми могат да "комуникират" и обменят данни помежду си

HTML5 API:

- Geolocation
- Local storage
- App Cache
- Web Workers
- Server-Sent Events

Други известни API:

- Google Maps API
- FaceBook API
- YouTube API
- EPay API
- PayPal API
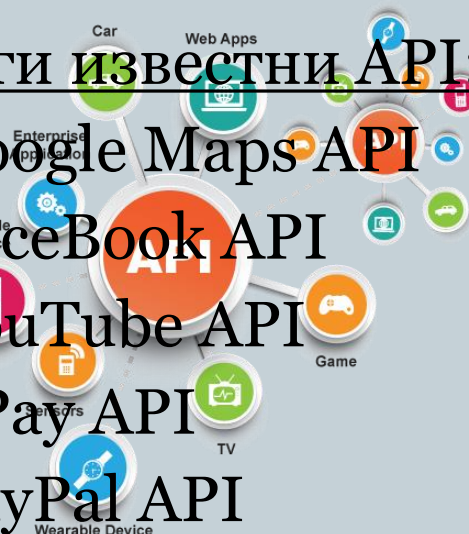
HTML5 API

HTML5 Geolocation:
* връща географското местоположение на потребителя
* не е достъпно, докато потребителя не даде позволение

Интересни методи:

* watchPosition() - следи движението (като GPS)
* clearWatch() - спира watchPosition() метода

Примери за използване на Geolocation:
* използване без проверка за грешки
* прихващане на съобщения за грешки
* показване на резултата чрез Google Maps
* използване на watchPosition

HTML5 Local Storage:

- по-добрият еквивалент на "бисквитките"
- по-сигурен вариант спрямо "бисквитките"
- позволява съхранение на по-големи данни
- съществуват 2 обекта за съхранение на данни:
  - window.localStorage - за съхранение на данни без срок
  - window.sessionStorage - съхранява данни само 1 сесия
- проверете дали се поддържа от браузъра преди ползване
- внимание: запазените данни винаги са в низов формат.

```
<script>                                // Проверка за поддръжка от браузъра
if(typeof(Storage) !== "undefined") {
    localStorage.setItem("lastname", "Smith");  // Съхраняваме информацията
} else { // Съжелявам! Браузъра не поддържа local storage! :-( }
</script>
```

Методи за работа с Local Storage:

- setItem(key, value) - поставя информация в обекта
- getItem(key) - връща стойността на съответния ключ
- removeItem(key) - премахва стойност от local storage
- възможно е да се ползва и директно обръщение, т.е. localStorage.key и localStorage.getItem(key) са еднакви

```
// Пример за работа с localStorage - броим, колко пъти сме изпълнили кода :-)
if (localStorage.clickcount) {
    localStorage.clickcount = Number(localStorage.clickcount) + 1;
} else {
    localStorage.clickcount = 1;
}
document.getElementById("result").innerHTML = "You have clicked the button " +
localStorage.clickcount + " time(s).";
```

HTML5 API

Server-Sent Events (често срещано като SSE):

- позволява уеб страницата да бъде обновявана от сървъра
- не забравяйте, да проверите дали баузъра поддържа SSE

```
if(typeof(EventSource) !== "undefined") {   // Проверка за поддръжка на SSE
    // Yes! Server-sent events support!
    // Some code.....
} else {
    // Sorry! No server-sent events support..
}
```

- Събития при работа със SSE:

Events	Description
onopen	When a connection to the server is opened
onmessage	When a message is received
onerror	When an error occurs

HTML5 API

```php
<?php                                              # Пример за SSE
   header('Content-Type: text/event-stream');
   header('Cache-Control: no-cache');
   $time = date('r');
   echo "data: The server time is: {$time}\n\n"; // задължително започва с data:
   flush();
?>
<div id="result"></div>
<script>
 if(typeof(EventSource) !== "undefined") {
    var source = new EventSource("demo_sse.php");
    source.onmessage = function(event) {
       document.getElementById("result").innerHTML += event.data + "<br>";
    };
 } else { document.getElementById("result").innerHTML = "Sorry"; }
</script>
```

Събития

Събитията (events) възникват, когато нещо се случи, а HTML от своя страна реагира на тези събития. HTML5 предоставя много нови събития, които могат да бъдат обособени в следните няколко категории:

- събития възникващи от браузъра
- събития възникващи от уеб форма
- събития възникващи от клавиатурата
- събития възникващи от мишката
- събития възникващи от мултимедия
- събития възникващи от Clipboard паметта
- събития възникващи от други източници

Нека ги разгледаме по подробно...
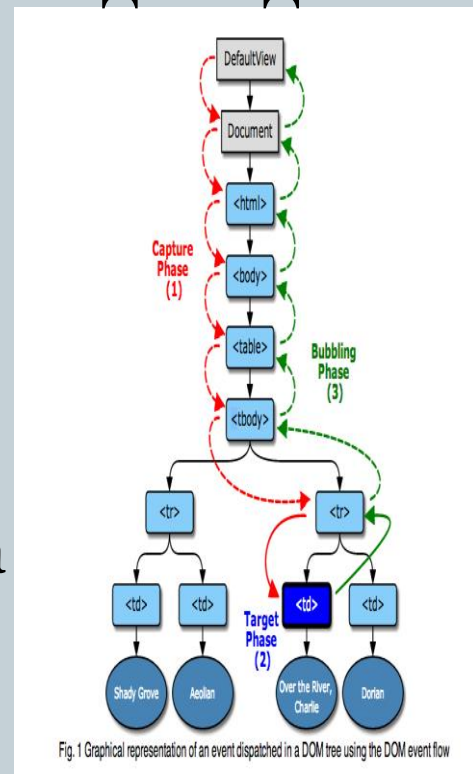
Fig. 1 Graphical representation of an event dispatched in a DOM tree using the DOM event flow

Събития

- Събития възникващи от браузъра:

Attribute		Value	Description
onafterprint	5	*script*	Script to be run after the document is printed
onbeforeprint	5	*script*	Script to be run before the document is printed
onbeforeunload	5	*script*	Script to be run when the document is about to be unloaded
onerror	5	*script*	Script to be run when an error occurs
onhashchange	5	*script*	Script to be run when there has been changes to the anchor part of the a URL
onload		*script*	Fires after the page is finished loading
onmessage	5	*script*	Script to be run when the message is triggered
onoffline	5	*script*	Script to be run when the browser starts to work offline
ononline	5	*script*	Script to be run when the browser starts to work online
onpagehide	5	*script*	Script to be run when a user navigates away from a page
onpageshow	5	*script*	Script to be run when a user navigates to a page
onpopstate	5	*script*	Script to be run when the window's history changes
onresize	5	*script*	Fires when the browser window is resized
onstorage	5	*script*	Script to be run when a Web Storage area is updated
onunload		*script*	Fires once a page has unloaded (or the browser window has been closed)

Събития

Събития възникващи от уеб форма:

Attribute		Value	Description
onblur		*script*	Fires the moment that the element loses focus
onchange		*script*	Fires the moment when the value of the element is changed
oncontextmenu	5	*script*	Script to be run when a context menu is triggered
onfocus		*script*	Fires the moment when the element gets focus
oninput	5	*script*	Script to be run when an element gets user input
oninvalid	5	*script*	Script to be run when an element is invalid
onreset	5	*script*	Fires when the Reset button in a form is clicked
onsearch		*script*	Fires when the user writes something in a search field (for <input="search">)
onselect		*script*	Fires after some text has been selected in an element
onsubmit		*script*	Fires when a form is submitted

Събития

Събития възникващи от мишката:

Attribute		Value	Description
onclick		*script*	Fires on a mouse click on the element
ondblclick		*script*	Fires on a mouse double-click on the element
ondrag	5	*script*	Script to be run when an element is dragged
ondragend	5	*script*	Script to be run at the end of a drag operation
ondragenter	5	*script*	Script to be run when an element has been dragged to a valid drop target
ondragleave	5	*script*	Script to be run when an element leaves a valid drop target
ondragover	5	*script*	Script to be run when an element is being dragged over a valid drop target
ondragstart	5	*script*	Script to be run at the start of a drag operation
ondrop	5	*script*	Script to be run when dragged element is being dropped
onmousedown		*script*	Fires when a mouse button is pressed down on an element
onmousemove		*script*	Fires when the mouse pointer is moving while it is over an element
onmouseout		*script*	Fires when the mouse pointer moves out of an element
onmouseover		*script*	Fires when the mouse pointer moves over an element
onmouseup		*script*	Fires when a mouse button is released over an element
onscroll	5	*script*	Script to be run when an element's scrollbar is being scrolled
onwheel	5	*script*	Fires when the mouse wheel rolls up or down over an element

Събития

Събития възникващи от мултимедия:

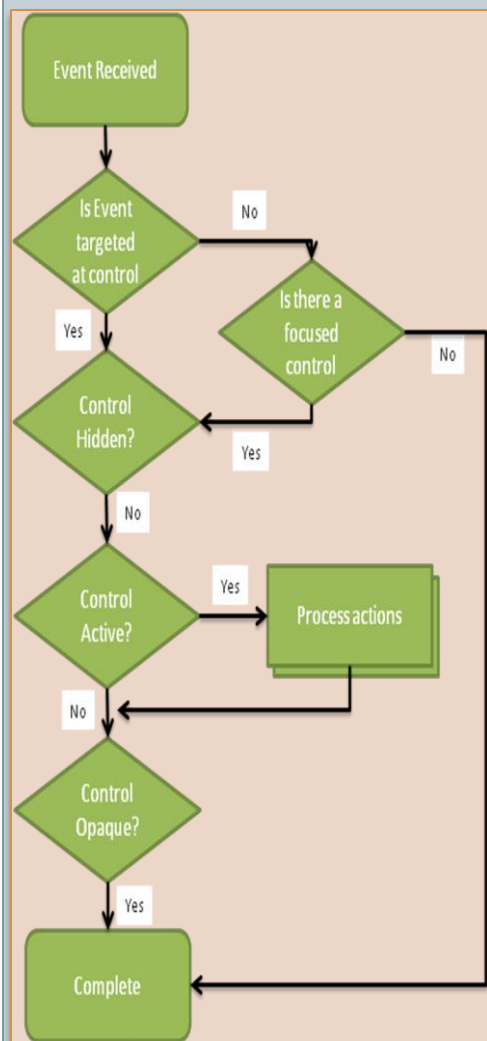| Attribute		Value	Description
onabort		*script*	Script to be run on abort
oncanplay	5	*script*	Script to be run when a file is ready to start playing (when it has buffered enough to begin)
oncanplaythrough	5	*script*	Script to be run when a file can be played all the way to the end without pausing for buffering
ondurationchange	5	*script*	Script to be run when the length of the media changes
onemptied	5	*script*	Script to be run when something bad happens and the file is suddenly unavailable (like unexpectedly disconnects)
onended	5	*script*	Script to be run when the media has reach the end (a useful event for messages like "thanks for listening")
onerror	5	*script*	Script to be run when an error occurs when the file is being loaded
onloadeddata	5	*script*	Script to be run when media data is loaded
onloadedmetadata	5	*script*	Script to be run when meta data (like dimensions and duration) are loaded
onloadstart	5	*script*	Script to be run just as the file begins to load before anything is actually loaded
onpause	5	*script*	Script to be run when the media is paused either by the user or programmatically

Събития

Събития възникващи от мултимедия:

Attribute	Value	Description
onplay	script	Script to be run when the media is ready to start playing
onplaying	script	Script to be run when the media actually has started playing
onprogress	script	Script to be run when the browser is in the process of getting the media data
onratechange	script	Script to be run each time the playback rate changes (when a user switches to a slow motion or fast forward mode)
onseeked	script	Script to be run when the seeking attribute is set to false indicating that seeking has ended
onseeking	script	Script to be run when the seeking attribute is set to true indicating that seeking is active
onstalled	script	Script to be run when the browser is unable to fetch the media data for whatever reason
onsuspend	script	Script to be run when fetching the media data is stopped before it is completely loaded for whatever reason
ontimeupdate	script	Script to be run when the playing position has changed (when the user fast forwards to a different point in the media)
onvolumechange	script	Script to be run each time the volume is changed which (includes setting the volume to "mute")
onwaiting	script	Script to be run when the media has paused but is expected to resume (when the media pauses to buffer more data)

Събития

Събития възникващи от клавиатурата:

Attribute	Value	Description
onkeydown	script	Fires when a user is pressing a key
onkeypress	script	Fires when a user presses a key
onkeyup	script	Fires when a user releases a key

Събития възникващи от Clipboard паметта:

Attribute	Value	Description
oncopy	script	Fires when the user copies the content of an element
oncut	script	Fires when the user cuts the content of an element
onpaste	script	Fires when the user pastes some content in an element

Събития възникващи от други източници:

Attribute		Value	Description
onerror	5	script	Fires when an error occurs while loading an external file
onshow	5	script	Fires when a <menu> element is shown as a context menu
ontoggle	5	script	Fires when the user opens or closes the <details> element

ВЪПРОСИ