

## ■教育実施記録帳

管理No.	FND27	Day14	Page1
実施内容	メソッド、DOM		
場所	ライオン	設備・工数	
実施日	2024/5/21	実施時間	9:00 ~ 12:00
実施者	Masaん DJさん	受講者	
作成者	本田	作成日(開始)	2024/5/21
		作成日(完了)	2024/5/21
<Time>	<Contents>		
	★メソッド (Method)		
	・オブジェクトプロパティ(キーと値のペア)の中で、 <u>値が関数</u> のものの呼称		
	[. js] オブジェクトプロパティへのメソッドの書き方		
	const dog = {		
	name: "Hana",		
	age: 12,		
	greet: function(friend) {		
	return "Bow-wow, " + friend + "!";		
	},		
	}		
	[. js] オブジェクトプロパティのメソッドの実行		
	console.log(dog.greet("Keisuke"));		
	[DevTool Console]		
	Bow-wow,Keisuke!		
	[. js] 例) dogに runメソッドを実装(コンソールに"run!"を出力)		
	const dog = {		
	name: "Hana",		
	age: 12,		
	greet: function(friend) {		
	return "Bow-wow, " + friend + "!";		
	run: function() {		
	console.log("run!");		
	},		
	}		
	[. js] オブジェクトプロパティのメソッドの実行		
	dog.run();		
	[DevTool Console]		
	run!		
	★DOM		
	・DocumentObjectModel (ドキュメントオブジェクトモデル) の略		
	(亜) <呼称にObjectって既に謳われてるんやね		
	・ドキュメント中の各要素は、親子関係や兄弟関係を持っている		
	(亜) <HTMLドキュメントは家系図みたいにツリー構造しとるんやね		
	・ブラウザはHTMLを読み込むと、ツリーのノードをJavaScriptのオブジェクトに変換して		
	メモリー中に格納する		
	(亜) <ここ大事なことやな...格納する際は他のデータと区別してるんやって		
	<div> <div> <div>HTML</div> </div> <div> <div>Loading...</div> </div> <div> <div>JavaScript オブジェクト</div> </div> </div> <div>このオブジェクトツリーをドキュメントオブジェクトモデルと呼び「DOM」や「DOMツリー」とも言う</div>		

## ■教育実施記録帳

管理No.	FND27	Day14	Page2
実施内容	メソッド、DOM		
場所	ライオン	設備・工数	
実施日	2024/5/21	実施時間	9:00 ~ 12:00
実施者	Masaん	受講者	
	DJさん		
作成者	本田	作成日(開始)	2024/5/21
		作成日(完了)	2024/5/21
<Time>	<Contents>		
	★documentオブジェクト		
	・DOMツリーのルートを指しており、つまりはHTML全体に相当する		
	・documentオブジェクトを介してJavaScriptでDOMにアクセスし、変更を加えることが出来る		
	★Document.getElementsByTagName()メソッド		
	・要素オブジェクトの中にある特定のタグを持つ全ての要素を取得する		
	☆戻り値：HTMLコレクション（配列のようなデータ）		
	[DevTool Console] toyota.jpでの実施		
	> const h2 =		
	document.getElementsByTagName("h2");		
	<・ undefined		
	[DevTool Console] 要素の中身の取得		
	> h2[0].innerHTML;		
	<・ “重要なお知らせ”		
	★.innerText		
	・要素オブジェクトが持つテキストコンテンツを調べる、変更する		
	[DevTool Console] 要素の中身の変更		
	> h2[0].innerText = “ちょっとだけ重要なお知らせ”;		
	<・ “ちょっとだけ重要なお知らせ”		
	★Document.createElement()メソッド		
	・JavaScriptのオブジェクトとして表されたHTMLの要素を作成する		
	[.js] 要素の作成		
	document.createElement("p");		
	[DevTool Console]		
	> document.createElement("p")		
	<・ <p></p> //新しくパラグラフ要素が作られる		
	> typeof document.createElement("p")		
	<・ "object" //要素のデータ型はオブジェクト		
	[.js] 新しく作った要素（オブジェクト）を変数に入れておけば後で使える		
	const paragraph = document.createElement("p");		
	★.appendChild()		
	・作成した要素を、既に表示されているHTML要素に追加する		
	[.js] 作成した要素		
	const paragraph = document.createElement("p");		
	[.js] 新しく作った要素にテキストを追加		
	paragraph.textContent = "I was born in JS, moulded by it"; //(亞) <哲学的やね...		
	[.js] 新しく作った要素にテキストを追加		
	document.body.appendChild(paragraph);		
	☆document.bodyが既に存在している要素だから追加出来る		