

■ 教育実施記録帳

管理No.	FND27	Day6	Page1
実施内容	複数の条件分岐、デバッグ入門、復習		
場所	オンライン	設備・工数	
実施日	2024/5/9	実施時間	9:00 ~ 12:00
実施者	Uraraさん	受講者	
	DJさん		
作成者	本田	作成日(開始)	2024/5/9
		作成日(完了)	2024/5/9
<Time>	<Contents>		
	★複数の条件分岐（論理演算）		
	・複雑な条件を組み立てたい時に「論理演算子」と「比較演算子」を組合わせて利用する		
	・ネスト(入れ子)が深いと処理が見難くなるので注意		
	// 論理演算子の左右式はどちらも有効な式である必要がある		
	function sayIfComfortable(temp) {		
	if (temp >= 20 && temp <= 25) { // 引数が抜けると無効の式となり		
	return "It's comfortable."; // エラーが表示される		
	} else if (temp < 18 temp >28) { // ネスト(入れ子)		
	return "It's not comfortable.";		
	} else {		
	return "It's ok.";		
	}		
	}		
	console.log(sayIfComfortable(22));		
	★論理演算子		
	演算子	読み	概要
	& &	アンバサンド2個	AND(論理積/かつ)
		パイプライン2個	OR(論理和/または)
	!	バング	NOT(否定/～でない)
	★論理演算子の評価結果		
	左式	右式	&& (AND)
			(OR)
	true	true	true
	true	false	false
	false	true	true
	false	false	false
	!true(false)	!true(false)	false
	!true(false)	!false(true)	true
	!false(true)	!true(false)	true
	!false(true)	!false(true)	true
	!true(false)	true	true
	!true(false)	false	false
	!false(true)	true	true
	!false(true)	false	true
	true	!true(false)	true
	true	!false(true)	true
	false	!true(false)	false
	false	!false(true)	true

■教育実施記録帳

管理No.	FND27	Day6	Page2
実施内容	複数の条件分岐、デバッグ入門、復習		
場所	ライオン	図書・工室	
実施日	2024/5/9	実施時間	9:00 ~ 12:00
実施者	Uraraさん	受講者	
	DJさん		
作成者	本田	作成日(開始)	2024/5/9
		作成日(完了)	2024/5/9
<Time>	<Contents>		
	★疑似コーディング(人間の言葉でロジックを組み立てる)		
	・効率的にどう書けば良いかを考える手法		・簡単なところからJSードに置き換えていく
	・疑似コードでもインデントは大事		・return文はわかるようにしておく
	// 疑似コードを書く		
	もし疲れてなかったら		
	return "勉強しよう"		
	そうでなければ		
	もしお金があれば		
	return "コーヒーを買おう"		
	そうでなければ		
	return "昼寝しよう"		
	// 関数宣言を加える		
	function whatToDo(isTired, hasMoney) {		
	もし疲れてなかったら		
	return "勉強しよう"		
	そうでなければ		
	もしお金があれば		
	return "コーヒーを買おう"		
	そうでなければ		
	return "昼寝しよう"		
	}		// 忘れないよう注意！！
	// 一つずつ置き換えていく		
	function whatToDo(isTired, hasMoney) {		
	// もし疲れてなかったら		
	if (!isTired) {		
	return "勉強しよう"		
	} else {		
	もしお金があれば		
	return "コーヒーを買おう"		
	そうでなければ		
	return "昼寝しよう"		
	}		// 忘れないよう注意！！
	}		
	// 一つずつ置き換えていく		
	function whatToDo(isTired, hasMoney) {		
	//もし疲れてなかったら		
	if (!isTired) {		
	return "勉強しよう"		
	} else {		
	// もしお金があれば		
	if (hasMoney) {		// ネストした条件分岐のため !「else if」
	return "コーヒーを買おう"		
	} else {		
	return "昼寝しよう"		
	}		// 忘れないよう注意！！
	}		
	}		

■教育実施記録帳

管理No.	FND27	Day6	Page3
実施内容	複数の条件分岐、デバッグ入門、復習		
場所	ライオン	図書・工室	
実施日	2024/5/9	実施時間	9:00 ~ 12:00
実施者	Uraraさん DJさん	受講者	
作成者	本田	作成日(開始)	2024/5/9
		作成日(完了)	2024/5/9
<Time>	<Contents>		
	// 不要なコメントを消す、テストコードを加える		
	function whatToDo(isTired, hasMoney) {		
	if (!isTired) {		
	return "勉強しよう"		
	} else {		
	if (hasMoney) {		
	return "コーヒーを買おう"		
	} else {		
	return "昼寝しよう"		
	}		
	}		
	}		
	whatToDo(true, false); // テストコードの追加		
	☆分岐の深さがピンとこないときはフローチャートを書いてみると疑似コードが書きやすくなる		
	こういった動きをコードで作ってみよう		
	<pre> graph TD A{疲れてる} -- はい --> B[はい] A -- いいえ --> C[いいえ] B --> D{お金ある} D -- はい --> E[はい] D -- いいえ --> F[いいえ] E --> G[コーヒーを買おう] F --> H[昼寝しよう] C --> I[いいえ] I --> J[勉強しよう] </pre>		
	★デバッグからの学び		
	・コードの動作や構造を確認できるようになる		
	・プログラムは「決定論的」でありデバッグを通して論理的な思考を養う		
	・コードを読みプログラムを解析することで、設計やロジックの問題点を見つけやすくなる		
	★デバッグの手法		
	//console.log()を利用する		
	function describeTheWeather(season, temp) {		
	let isNormal;		
	if (season === "Spring") {		
	console.log("Season is spring!"); // ここまで正常にブロックが実行されている		
	isNormal = temp === "warm"; // ことを確認するために追加する		
	}		
	// ...		
	console.log("isNormal:", isNormal); // コードを実行すれば「isNormal」に代入		
	//される値を確認できる		
	if (isNormal) {		
	return "The temperature is normal for the season.";		
	}		
	return "The temperature is unusual for the season.";		
	}		
	☆変数や引数の値がわからない時はconsole.log()で表示すればいい		

■教育実施記録帳

管理No.	FND27	Day6	Page4
実施内容	複数の条件分岐、デバッグ入門、復習		
場所	オンライン	設備・工数	
実施日	2024/5/9	実施時間	9:00 ~ 12:00
実施者	Uraraさん	受講者	
	DJさん		
作成者	本田	作成日(開始)	2024/5/9
		作成日(完了)	2024/5/9
<Time>	<Contents>		
	★デバッグメッセージの書き方		
	// 良いデバッグメッセージの書き方		
	console.log(number);	OKAY	
	//変数の名前を一緒に表示		
	console.log("number is:", number);	BETTER	
	//変数の名前と期待値と一緒に表示		
	console.log("number should be 3:", number);	BEST	
	★関数・比較・条件分岐レビュー		
	・仮引数 (parameter)	・関数の中身 (function body)	
	・function キーワード	・関数呼び出し	
	・return文	・実引数 (arguments)	
	・返り値	・関数宣言	
	// 構成と名称		
	function createGreeting(greeting, firstName, lastName) {		
	return greeting + firstName + lastname;		
	}		
	creategreeting("Konnichiwa","Tsubasa", "Kondo");		
	> "KonnichiwaTsubasaKondo"		
	★Pop Quiz!		
	・ブーリアンとは？		
	データの異なるカテゴリの一つ。真偽値を返す。		
	・"true" === true の結果は？		
	false		
	・typeof は何をするもの？使用方法は？		
	オペランドのデータ型を取得		
	console.log(typeof Nan); // => number		
	・expression とは？		
	JavaScript の式		
	・.length は何をするもの？使用方法は？		
	文字列の長さ（文字数）を取得する		
	console.log("NaN".length); // => 3		
	・if文の丸かっこの中はどんなデータ型の値？		
	boolean型のture/false		
	・! は何をするもの？		
	バングは否定するもの		
	・& &は何をするもの？		
	アンパサンド2個は論理積するもの		