

変数への代入とその確認方法

変数への代入と確認方法

JavaScriptでは変数に代入した値を取り扱うことが多いです。

なので、今回は変数への代入と、代入されている値が正しいのかも確認を行なっていきます。

変数への代入方法

変数への代入は変数への代入の宣言を必要があります。

```
var variable = 1234;
```

この`var`の宣言を行っており、`variable`を変数名として定義しています。

これ以降のコードで`variable`は変数として取り扱われます。

- `var`以外にも変数の宣言方法がありますが、別の課題で紹介します。

コードの最後にある`;`ですが、なくても問題ないですが一応文法としては付けることになっていますので付けるようにしましょう。

変数のデータ型

変数へ値が代入されると代入された値によってデータ型が決まります。

```
// variableNumberはNumber型になる
var variableNumber = 1234;

// variableTextはString型になる
var variableText = "Text";

// variableBooleanはBoolean型になる
var variableBoolean = true;
```

また計算した結果や、文字列を結合したものを代入することも可能です。

```
// addNumberは2になる
var addNumber = 1 + 1;

// combinationTextは"JavaScript"になる
var combinationText = "Java" + "Script";
```

計算は...

- `+`が足し算
- `-`が引き算
- `*`が掛け算
- `/`が割り算

文字列について

文字列はシングルクォーテーション（`'`）または、ダブルクォーテーション（`"`）認識されます。どちらが正解などはとくにありません。

会社や案件によって異なるので柔軟に対応できるようにしましょう。

課題の制作をマークダウン形式で書いている関係上シングルクォーテーションがコードブロックとして認識されるのでダブルクォーテーションで書いています。

代入された変数の確認方法

プログラミングを行なっている際、正しい値が代入できているか確認したい時があります。

その時に使用するのが、`console.log()`です。

作成した変数を（`log`）の中に書き、そのコードがブラウザで読み込まれて処理されるとブラウザに表示されます。

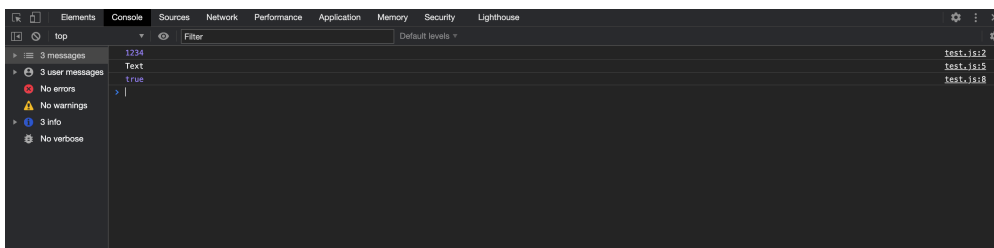
下記のコード書いて、確認したい場合は...

```
var variableNumber = 1234;
console.log(variableNumber);

var variableText = "Text";
console.log(variableText);

var variableBoolean = true;
console.log(variableBoolean);
```

これを読み込んだ際はこのようになります。



代入された内容が表示され、データ型によっても色が変わっています。

`console.log()`を書く時の注意点

`console.log()`は変数が作られた後でないと表示しないので必ず変数を作成した後に書いてください。

課題

1. 変数にNumber型を代入して`console.log()`で表示させてください。
2. 変数にString型を代入して`console.log()`で表示させてください。
3. 変数にBoolean型を代入して`console.log()`で表示させてください。
4. 10と20を足した結果を変数に代入して`console.log()`で表示させてください。
5. 20から10を引いた結果を変数に代入して`console.log()`で表示させてください。
6. 50へ2を掛けた結果を変数に代入して`console.log()`で表示させてください。
7. 20を2で割った結果を変数に代入して`console.log()`で表示させてください。
8. 12と34を結合させた結果を変数に代入して`console.log()`で表示させてください。