

文法エラー (syntax error) Javascriptの文法ルールに違反した時に出るエラー

- Javascriptの文法ルールに違反した時に出るエラー
- 実行前にチェックできるエラーです。
- Uncaught SyntaxError...

1 カッコの閉じ忘れ

```
[ ] if (true {  
    console.log("hello");  
}
```

2 最後のカッコの閉じ忘れ

```
[ ] for (let k = 0; k < 3; i++){  
    console.log(k);
```

3 文字列の閉じ忘れ

```
[ ] console.log("hello);
```

4 ダブルクォートとシングルクォートの混乱

```
🕒 console.log('I'm mario'); -> console.log("I `m mario"); それか console.log("I'm mario");  
  
console.log('I'm mario'); -> console.log('I\'m mario');
```

5 予約語の誤り

```
[ ] let for = 10;
```

6 不正な変数名 先に番号を入れるとエラーになります

```
let 123num = "mario";❌ let num123 = "mario";
```

7 カンマが足りないエラー

```
const array = [1,2 3] -> const array = [1,2,3];
```

8 関数の書き間違い

```
function hello (  
    console.log("hi")  
)
```

9 セミコロン (特に波括弧の前にNG)

```
if (true);  
else{  
    console.log("hello");  
}
```

10 埋め込みタグ

```
let name = "maru";  
console.log("hello ${name}") -> (`hello, ${name}`)
```

参照エラー(reference error)

1. 存在しない変数や関数を使おうとした時に出るエラーです。
2. その名前は知らなよ！って言ってくる。
3. Uncaught ReferenceError: ...

1 変数が宣言されていない

```
console.log(x) ->変数を書く
```

2 スペルミス

```
let count = 20;  
console.log(cout);
```

3関数の中に初期の変数を書いてしまう

```
function test(){  
    let value =100;  
}  
console.log(value);
```

4関数が未定義 この場合は呼び出し先が決まっていますが、関数自体は宣言されていない。

```
sayHello();
```

型エラー (type error)

1. データの種類 (型) が合わない時に出るエラー
2. そのデータ型がそんなことできないよ！と言ってくれる
3. Uncaught TypeError: ...

1 関数じゃないものを呼び出す

```
let num = 123;  
num();
```

2 undedfined/nullにプロパティを使う

```
let obj;  
console.log(obj.name);  
  
let data = null;  
console.log(data.value);
```

3 存在しないメソッドえお呼びます

```
let num = 10;  
console.log(num.toUpperCase())
```

範囲エラー (range error)

1. 値がや許されている範囲の外にあるとき出るエラー
2. その範囲が無効です！と教えてくれる
3. Uncaught RangeError:

1 配列の長さはマイナスで指定する

```
let arr =[1,2,3]  
arr.length = -1;
```

2 数字メソッドの範囲外

```
let num = 1.230000;  
console.log(num.toFixed(200))
```

3 再帰の呼びすぎ

```
//❌NG  
function num(){  
    return num()  
}  
  
num();
```

4 関数の中に正しく書いた処理命令を入れる

```
function num(n){  
  
    if (n === 0){  
        console.log("終わり")  
        return;  
    }  
    console.log(n);  
    num(n -1);  
  
}
```

```
num(5);
```