

Quiz: Understanding `this` in JavaScript

Please answer those questions without running the code.

1. Question 1

Consider the following code snippet:

```
"use strict";

function myFunction() {
  console.log(this);
}

const obj = {
  prop: "Hello",
  method: function() {
    myFunction();
  }
};

obj.method();
```

What will be the output when `obj.method()` is called?

- a) `undefined` b) `null` c) `ReferenceError: myFunction is not defined` d) `Window`

2. Question 2

Given the following code:

```
const calculator = {
  result: 0,
  add: function(num) {
    this.result += num;
  }
};

const addFunction = calculator.add;

calculator.add(5);
addFunction(3);

console.log(calculator.result);
```

What will be the output of the `console.log` statement?

- a) 5 b) 3 c) 8 d) NaN

3. Question 3

Consider this constructor function:

```
function Dog(name) {
  this.name = name;
}

const dog = Dog("Buddy");
console.log(dog.name);
```

What will be the output of the `console.log` statement?

- a) undefined b) "Buddy" c) ReferenceError: name is not defined d) TypeError: Cannot read properties of undefined (reading 'name')

4. Question 4

Given the following code:

```
"use strict";

const obj1 = {
  name: "Alice",
  printName: function() {
    console.log(this.name);
  }
};

const obj2 = {
  name: "Bob"
};

obj2.printName = obj1.printName;

obj2.printName();
```

What will be the output when `func()` is called?

a) `undefined` b) `null` c) `"Alice"` d) `"Bob"`

5. Question 5

Consider the following code:

```
const obj = {
  name: "John",
  regularFunction: function() {
    console.log(this.name);
  },
  arrowFunction: () => {
    console.log(this.name);
  }
};

obj.regularFunction();
obj.arrowFunction();
```

What will be the output when `obj.regularFunction()` and `obj.arrowFunction()` are called?

a) "John" for both b) "John" for `obj.regularFunction()` and `undefined` for `obj.arrowFunction()` c) `undefined` for both d)

`TypeError: Cannot read property 'name' of undefined`