

setTimeout with this

1. Issues
2. Solution: use a wrapper function
3. Solution: use bind()

1. Issues

- Không thể bind this cho setTimeout.
- this trong callback function của setTimeout default là global object (window)

```
const student = {
  name: 'Long Do',
  sayHello() {
    console.log('NAME', this.name);
    console.log('THIS', this);
  },
}
student.sayHello(); // 'Long Do'

setTimeout(student.sayHello); // this is window, not student

setTimeout.call(student, student.sayHello); // Uncaught TypeError: Illegal
invocation

const a = setTimeout.bind(student)
a(student.sayHello) // Uncaught TypeError: Illegal invocation
```

2. Solution: use a wrapper function

```
const student = {
  name: 'Long Do',
  sayHello() {
    console.log('NAME', this.name);
    console.log('THIS', this);
  },
}
setTimeout(() => {
  student.sayHello(); // this is student --> 'Long Do'
});
```

3. Solution: use bind()

```
const student = {
  name: 'Long Do',
```

```
    sayHello() {
      console.log('NAME', this.name);
      console.log('THIS', this);
    },
  }
  setTimeout(student.sayHello.bind(student)); // 'Long Do'
```

Thêm ví dụ

```
const student = {
  name: "Long Do",
  normal() {
    console.log("NAME", this.name);
    console.log("THIS", this);
  },

  arrow: () => {
    console.log("NAME", this.name);
    console.log("THIS", this);
  },

  timeoutNormal() {
    console.log("OUTER THIS", this);
    setTimeout(function () {
      console.log("NAME", this.name);
      console.log("THIS", this);
    });
  },

  timeoutArrow() {
    console.log("OUTER THIS", this);
    setTimeout(() => {
      console.log("NAME", this.name);
      console.log("THIS", this);
    });
  },
};
```