

5 Differences between Arrow & Regular Functions

{.js}

JavaScript

VS

```
const greet = function(who) {  
  return `Hello, ${who}`;  
}
```

```
const greet = (who) => {  
  return `Hello, ${who}!`;  
}
```



01

this value

Regular Function

```
// indirect invocation
function myFunction() {
  console.log(this);
}
const myContext = { value: 'A' };
myFunction.call(myContext); // logs { value: 'A' }
myFunction.apply(myContext); // logs { value: 'A' }
```

Arrow Function

```
const myObject = {
  myMethod(items) {
    console.log(this); // logs myObject
    const callback = () => {
      console.log(this); // logs myObject
    };
    items.forEach(callback);
  }
};
myObject.myMethod([1, 2, 3]);
```



02

Constructors

Regular Function

```
function Car(color) {  
  this.color = color;  
}  
const redCar = new Car('red');  
redCar instanceof Car; // => true
```

Arrow Function

```
const Car = (color) => {  
  this.color = color;  
};  
const redCar = new Car('red');  
// TypeError: Car is not a constructor
```



03

Arguments Object

Regular Function

```
function myFunction() {  
  console.log(arguments);  
}  
myFunction('a', 'b');  
// logs { 0: 'a', 1: 'b', length: 2 }
```

Arrow Function

```
function myRegularFunction() {  
  const myArrowFunction = () => {  
    console.log(arguments);  
  }  
  myArrowFunction('c', 'd');  
}  
myRegularFunction('a', 'b');  
// logs { 0: 'a', 1: 'b', length: 2 }
```



04

Implicit return

Regular Function

```
function myFunction() {  
  return 42;  
}  
myFunction(); // ⇒ 42
```

Arrow Function

```
const increment = (num) ⇒ num + 1;  
increment(41); // ⇒ 42
```



04

Methods

Regular Function

```
class Hero {  
  constructor(heroName) {  
    this.heroName = heroName;  
  }  
  logName() {  
    console.log(this.heroName);  
  }  
}  
const batman = new Hero('Batman');
```

Arrow Function

```
class Hero {  
  constructor(heroName) {  
    this.heroName = heroName;  
  }  
  logName = () => {  
    console.log(this.heroName);  
  }  
}  
const batman = new Hero('Batman');
```



Thanks for reading !

Follow my profile for more
coding related contents

Like and Share the post
with your network

@Anandhraj Mudaliyar