



身為前端工程師

你有想過用  做簡報嗎？

WINNIE WU



而且還連續做了30天

- 2023 iThome 鐵人賽

影片教學

在大腦升級後，我想要學 TypeScript 了 系列

鐵人鍊成 | 共 30 篇文章 | 23 人訂閱 | 團隊 不知道我能不能頂住

- 2023 iThome 鐵人賽

影片教學

在大腦升級後，我想要學 TypeScript 了 系列

鐵人鍊成 | 共 30 篇文章 | 23 人訂閱 | 團隊 不知道我能不能頂住

- 2023 iThome 鐵人賽

影片教學

在大腦升級後，我想要學 TypeScript 了 系列

鐵人鍊成 | 共 30 篇文章 | 23 人訂閱

團隊

不知道我能不能頂住



完了...要變殭屍了





RecordRTC.js

A decorative graphic on the left side of the slide featuring three overlapping shapes: a large teal circle, a teal rounded square, and a yellow rounded triangle.

為什麼 **Slidev** ?

Componet

```
<Counter :count="10" />
```

-	10	+
---	----	---

CODE<>

HeightLight

突顯某部分重要程式碼片段時，可使用 Markdown 中的三個反引號（``）來指定程式碼區塊，同時在後面指定程式碼的語言。

```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}  
  
//... 略  
  
function updateUser(id: number, update: User) {  
  const user = getUser(id)  
  const newUser = { ...user, ...update }  
  saveUser(id, newUser)  
}
```

HeightLight

突顯某部分重要程式碼片段時，可使用 Markdown 中的三個反引號（``）來指定程式碼區塊，同時在後面指定程式碼的語言。

```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}  
  
//... 略  
  
function updateUser(id: number, update: User) {  
  const user = getUser(id)  
  const newUser = { ...user, ...update }  
  saveUser(id, newUser)  
}
```

HeightLight

突顯某部分重要程式碼片段時，可使用 Markdown 中的三個反引號（``）來指定程式碼區塊，同時在後面指定程式碼的語言。

```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}  
  
//... 略  
  
function updateUser(id: number, update: User) {  
  const user = getUser(id)  
  const newUser = { ...user, ...update }  
  saveUser(id, newUser)  
}
```


HeightLight

突顯某部分重要程式碼片段時，可使用 Markdown 中的三個反引號（``）來指定程式碼區塊，同時在後面指定程式碼的語言。

```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}  
  
//...略  
  
function updateUser(id: number, update: User) {  
  const user = getUser(id)  
  const newUser = { ...user, ...update }  
  saveUser(id, newUser)  
}
```



HeightLight

突顯某部分重要程式碼片段時，可使用 Markdown 中的三個反引號（``）來指定程式碼區塊，同時在後面指定程式碼的語言。

```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}  
  
//... 略  
  
function updateUser(id: number, update: User) {  
  const user = getUser(id)  
  const newUser = { ...user, ...update }  
  saveUser(id, newUser)  
}
```

HeightLight

突顯某部分重要程式碼片段時，可使用 Markdown 中的三個反引號（``）來指定程式碼區塊，同時在後面指定程式碼的語言。

```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}  
  
//... 略  
  
function updateUser(id: number, update: User) {  
  const user = getUser(id)  
  const newUser = { ...user, ...update }  
  saveUser(id, newUser)  
}
```

Monaco Editor & Monaco Diff

```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}  
  
//...略  
  
function updateUser(id: number, update: User) {  
  const user = getUser(id)  
  const newUser = { ...user, ...update }  
  saveUser(id, newUser)  
}
```

Monaco Editor & Monaco Diff

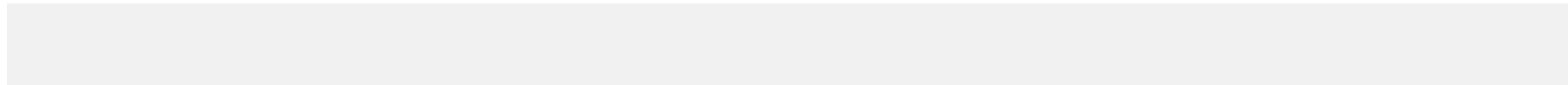
```
interface User {  
  id: number  
  firstName: string  
  lastName: string  
  role: string  
}  
  
//...略  
  
function updateUser(id: number, update: User) {  
  const user = getUser(id)  
  const newUser = { ...user, ...update }  
  saveUser(id, newUser)  
}
```

```
removed on the right.  
just some text  
abcd  
efgh  
Some more text  
~~~~  
just some text  
abcz  
zzzzefgh  
Some more text.
```

Animations

v-click

DEMO:



程式碼:

```
<div class="border my-10 tracking-wide relative">
  <div class="text-[#696969] bg-[#f1f1f1] p-2" >
    <span v-click="[1,2]">我.. <span class="text-2xl">{{ $clicks }}</span> ..出現的</span>
    <span v-click="2">我是點擊第 <span class="text-2xl">2</span> 次 出現的</span>
    <span class="transition-all duration-500 ease-linear" :class="$clicks === 3? 'opacity-100': 'opacity-0'" >...{{ $clicks
  </div>
</div>
```

v-click

DEMO:

我是點擊第 1 次 出現的

程式碼:

```
<div class="border my-10 tracking-wide relative">
  <div class="text-[#696969] bg-[#f1f1f1] p-2" >
    <span v-click="[1,2]">我.. <span class="text-2xl">{{ $clicks }}</span> ..出現的</span>
    <span v-click="2">我是點擊第 <span class="text-2xl">2</span> 次 出現的</span>
    <span class="transition-all duration-500 ease-linear" :class="$clicks === 3? 'opacity-100': 'opacity-0'" >...{{ $clicks
  </div>
</div>
```


v-click

DEMO:

我是點擊第 2 次 出現的

程式碼:

```
<div class="border my-10 tracking-wide relative">
  <div class="text-[#696969] bg-[#f1f1f1] p-2" >
    <span v-click="[1,2]">我.. <span class="text-2xl">{{ $clicks }}</span> ..出現的</span>
    <span v-click="2">我是點擊第 <span class="text-2xl">2</span> 次 出現的</span>
    <span class="transition-all duration-500 ease-linear" :class="$clicks === 3? 'opacity-100': 'opacity-0'" >...{{ $clicks
  </div>
</div>
```

v-click

DEMO:

我是點擊第 2 次 出現的

我是點擊第 3 次 出現的

程式碼:

```
<div class="border my-10 tracking-wide relative">
  <div class="text-[#696969] bg-[#f1f1f1] p-2" >
    <span v-click="[1,2]">我.. <span class="text-2xl">{{ $clicks }}</span> ..出現的</span>
    <span v-click="2">我是點擊第 <span class="text-2xl">2</span> 次 出現的</span>
    <span class="transition-all duration-500 ease-linear" :class="$clicks === 3? 'opacity-100': 'opacity-0'" >...{{ $clicks
  </div>
</div>
```

v-motion

```

```

v-motion

```

```

v-motion

```

```

v-motion

```

```

v-motion

```

```



Recording



Export 

PDF



更多功能

[Documentations](#) · [GitHub](#) · [Showcases](#)



相較其他『簡報工具』優勢

DEMO

最後

『先做做看就知道了』

焦慮有一天會變成規律

