

Attribute · class · style binding

attribute binding

[attr.colspan]="2"

[attr.rowspan]="2"

class binding

[class]="myClassVar"

[class]=" ' title active ' "

} 取代原有 class 設定

[class.active]="true" → 新增 / 移除 單一 class

style binding

[style.color]="myColor"

[style.color]=" ' red ' "

[style.background-color]="myColor"

→ 小駝峰 camelCase

[style.backgroundColor]=" ' red ' "

→ 大駝峰 PascalCase

[style.font-size.**em**]="mySize"

[style.fontSize.**em**]="3"

[style.width.**px**]="300"

} 有單位(em px % ...)

ngStyle 、 *ngClass*

ngClass 同時操作多個 class

```
ngClass = "active"
```

```
[ ngClass ] = " ' active ' "
```

```
[ ngClass ] = "myClassStringVar"
```

```
[ ngClass ] = "myClassArray"
```

```
[ ngClass ] = "myClassObject"
```

注意減號 (-)

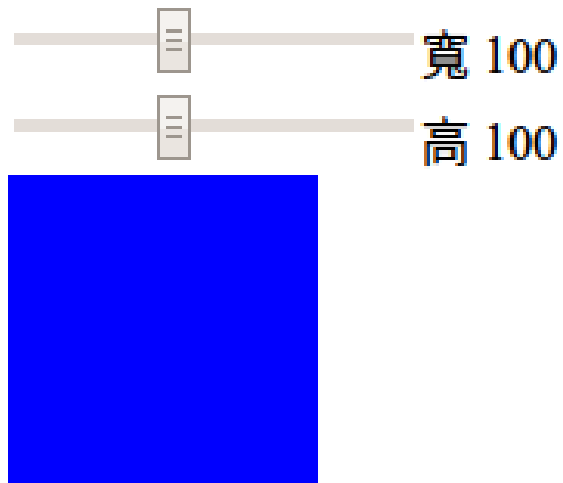
```
myClassObject = {  
  'font-size': 3em;  
  fontSize: 3em;  
};
```

ngStyle 同時操作多個 property

[ngStyle]="myStyleObject"

```
myClassObject = {  
  'font-size': '3em',    注意減號 (-)  
  color: 'red',  
  backgroundColor: '#00ff00'  
};
```

練習 1：動態調整大小



練習 1：動態調整大小

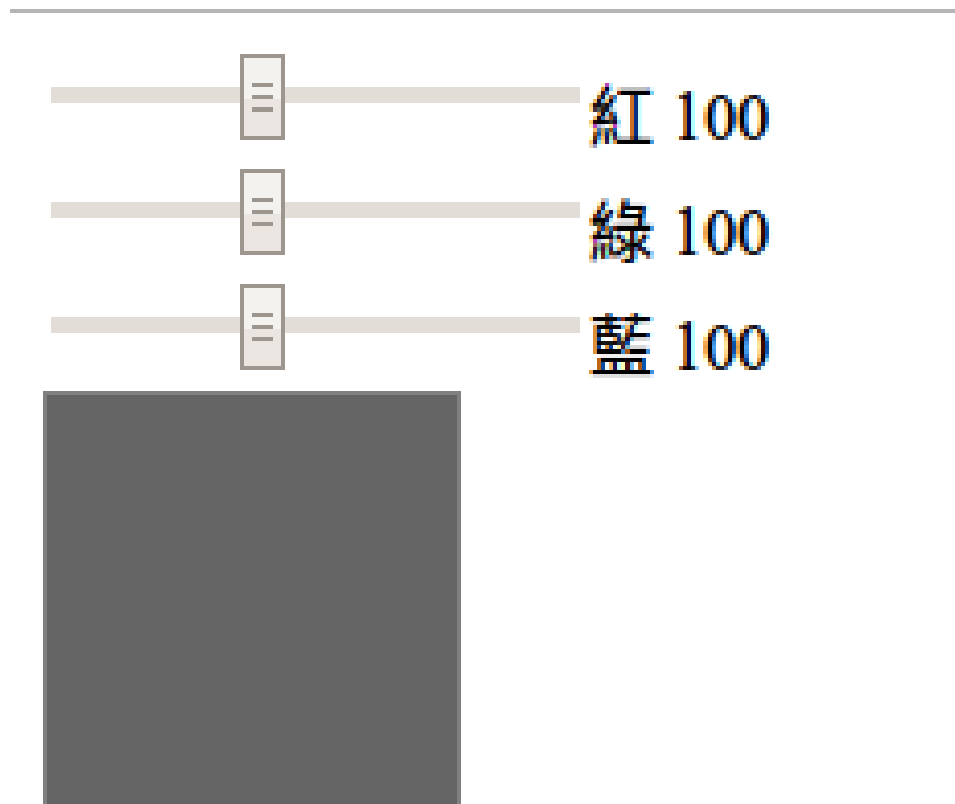
```
<input type="range" min="100" max="200"
      [(ngModel)]="w">寬 {{ w }}
<br />
<input type="range" min="100" max="200"
      [(ngModel)]="h">高 {{ h }}

<div style="background-color: blue"
      [style.width.px]="w"
      [style.height.px]="h" ></div>
```

練習 1：動態調整大小

```
export class AppComponent {  
    w = 100;  
    h = 100;  
}
```

練習 2：調色盤



練習 2：調色盤

```
<input type="range" min="0" max="255"  
  [(ngModel)]="r"  
  (input)="changeColor()">紅 {{ r }}
```

```
<br />
```

```
<input type="range" min="0" max="255"  
  [(ngModel)]="g"  
  (input)="changeColor()">綠 {{ g }}
```

```
<br />
```

```
<input type="range" min="0" max="255"  
  [(ngModel)]="b"  
  (input)="changeColor()">藍 {{ b }}
```



```
<div style="width: 100px; height: 100px; border: gray 1px solid;"  
  [ngStyle]="myColor"></div>
```

練習 2：調色盤

```
export class AppComponent {  
  r = 100;  
  g = 100;  
  b = 100;  
  myColor = {};  
  
  constructor() {  
    this.changeColor();  
  }  
  
  changeColor() {  
    this.myColor = {  
      backgroundColor: `rgb(${this.r}, ${this.g}, ${this.b})`  
    };  
  }  
}
```

END