

# Grid基礎觀念

# 基礎用法1

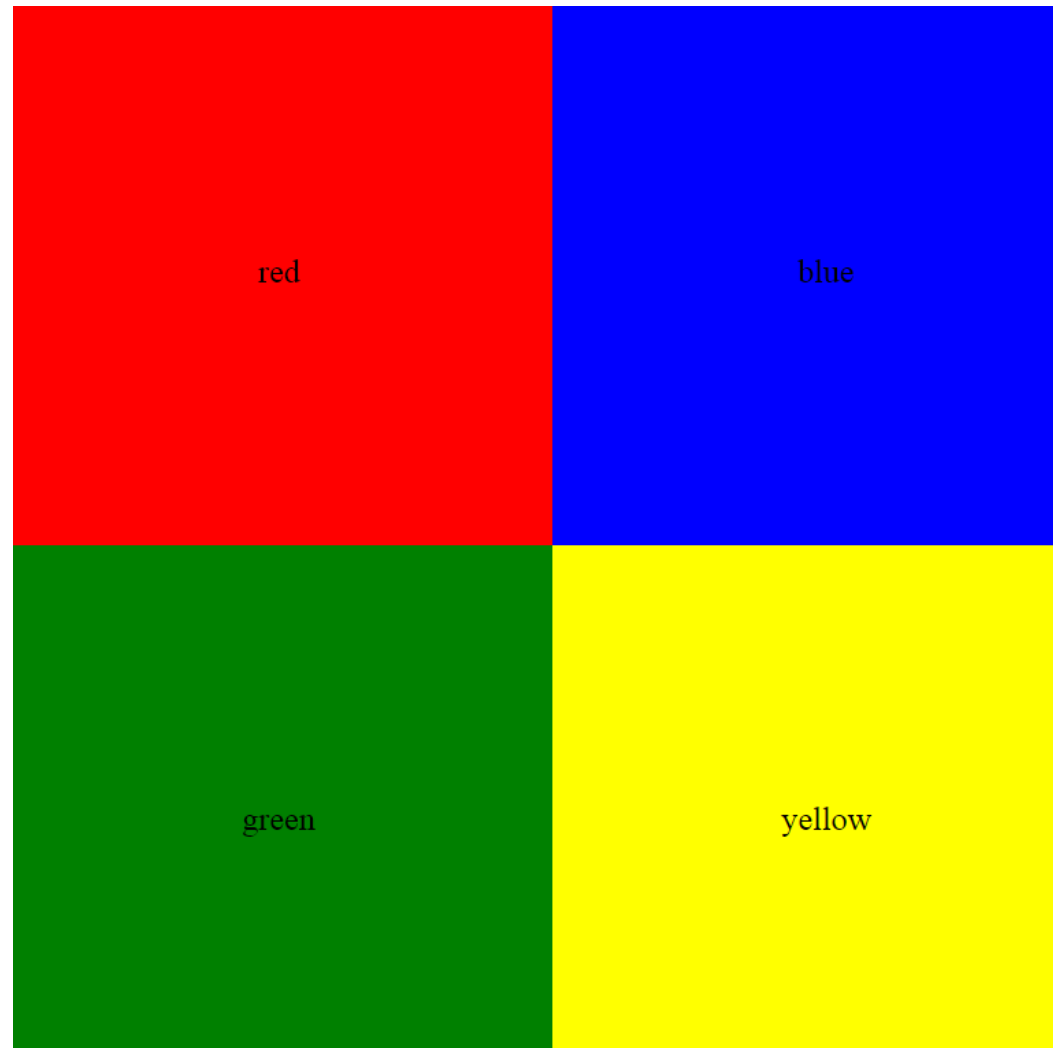
```
.grid {  
  width: 800px;  
  height: 800px;  
  display: grid;  
  grid-template-columns: 1fr 1fr ;  
  grid-template-rows: 1fr 1fr ;  
}
```

```
.grid {  
  width: 800px;  
  height: 800px;  
  display: grid;  
  grid-template-columns: repeat(2, 1fr);  
  grid-template-rows: repeat(2, 1fr);  
}
```

```
.red {  
  background-color: red;  
  grid-area: red;  
}  
  
.blue {  
  background-color: blue;  
  grid-area: blue;  
}  
  
.green {  
  background-color: green;  
  grid-area: green;  
}  
  
.yellow {  
  background-color: yellow;  
  grid-area: yellow;  
}
```

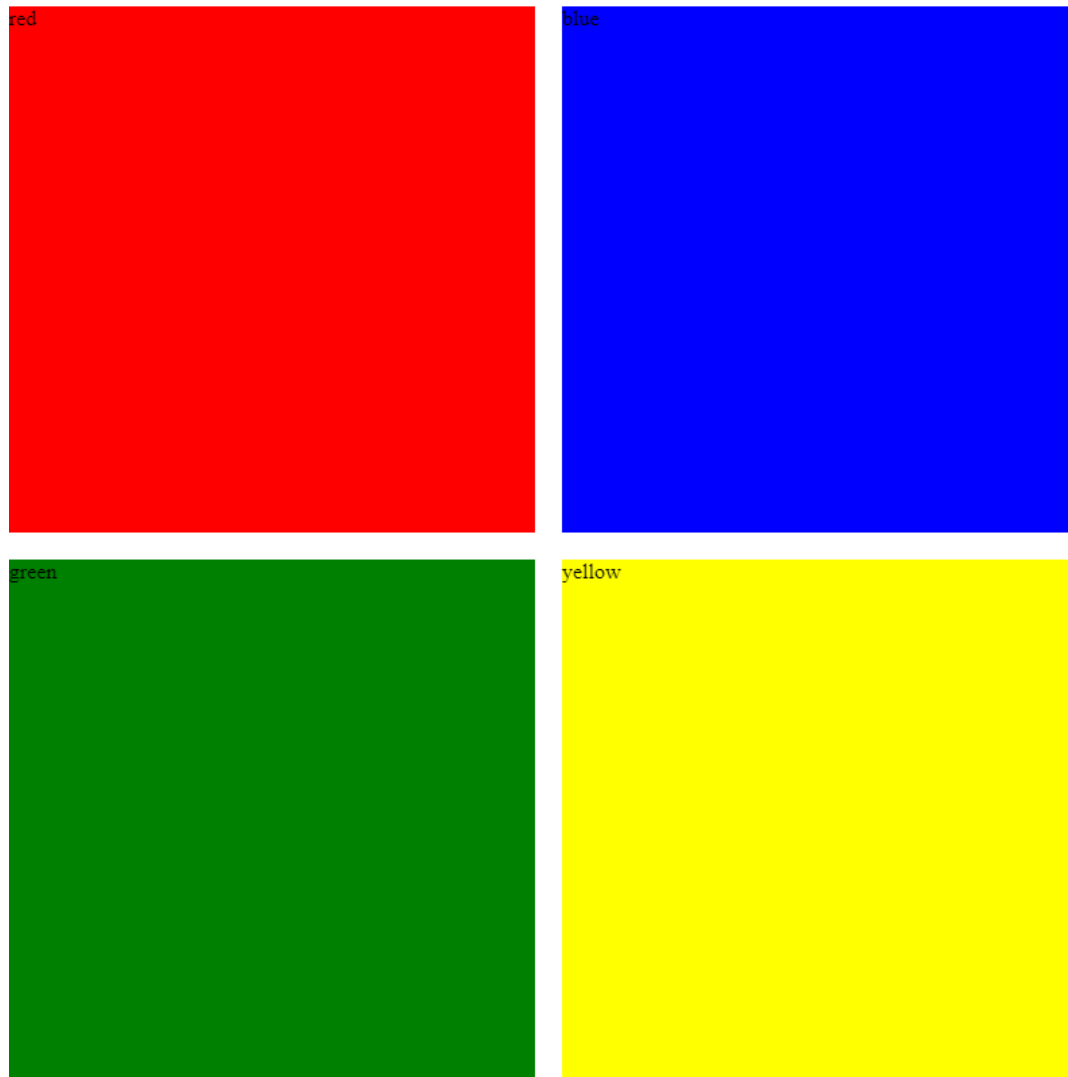
# 排版1

```
.grid {  
  width: 800px;  
  height: 800px;  
  display: grid;  
  grid-template-columns: repeat(2, 1fr);  
  grid-template-rows: repeat(2, 1fr);  
  grid-template-areas:  
    "red blue"  
    "green yellow"  
  ;  
}
```



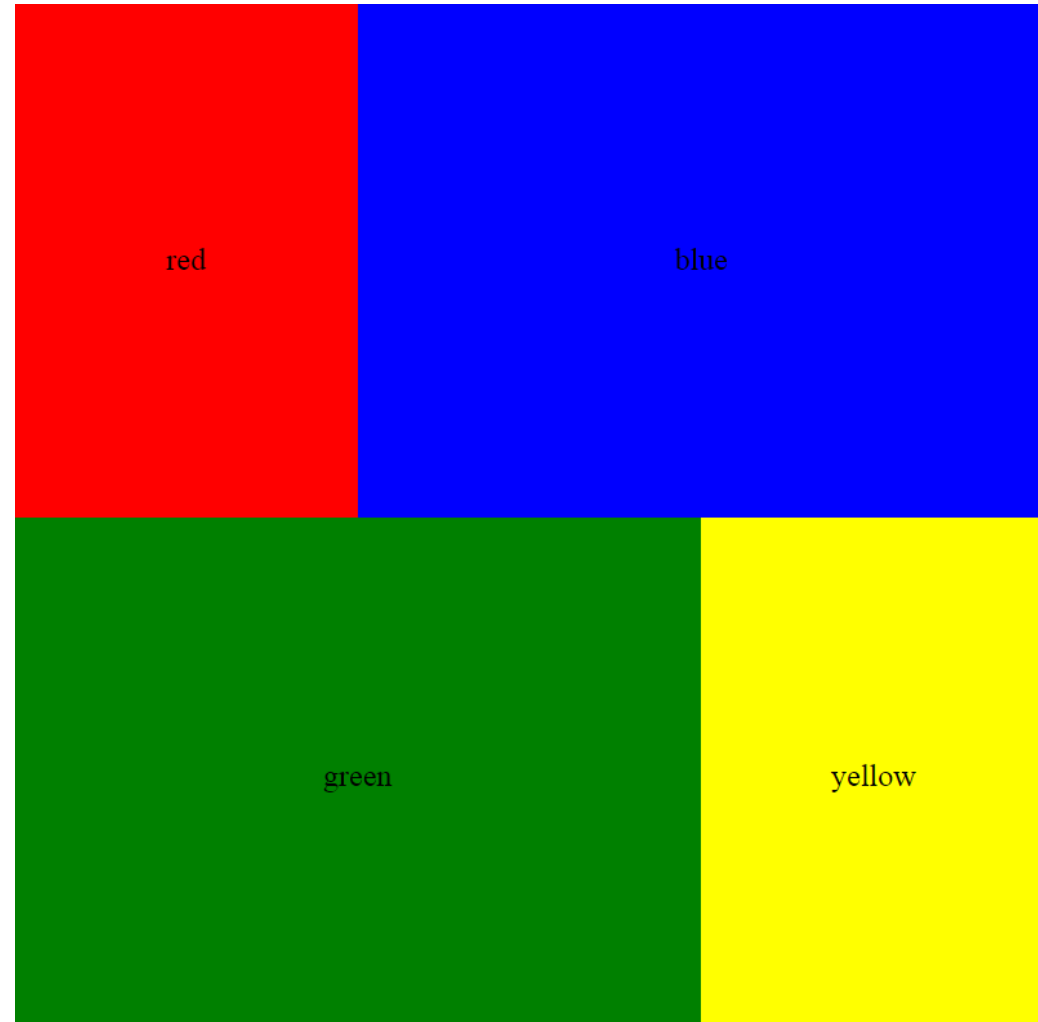
# 產生間距

```
.grid {  
  width: 800px;  
  height: 800px;  
  display: grid;  
  grid-template-columns: repeat(2, 1fr);  
  grid-template-rows: repeat(2, 1fr);  
  gap: 20px;  
  grid-template-areas:  
    "red blue"  
    "green yellow"  
  ;  
}
```

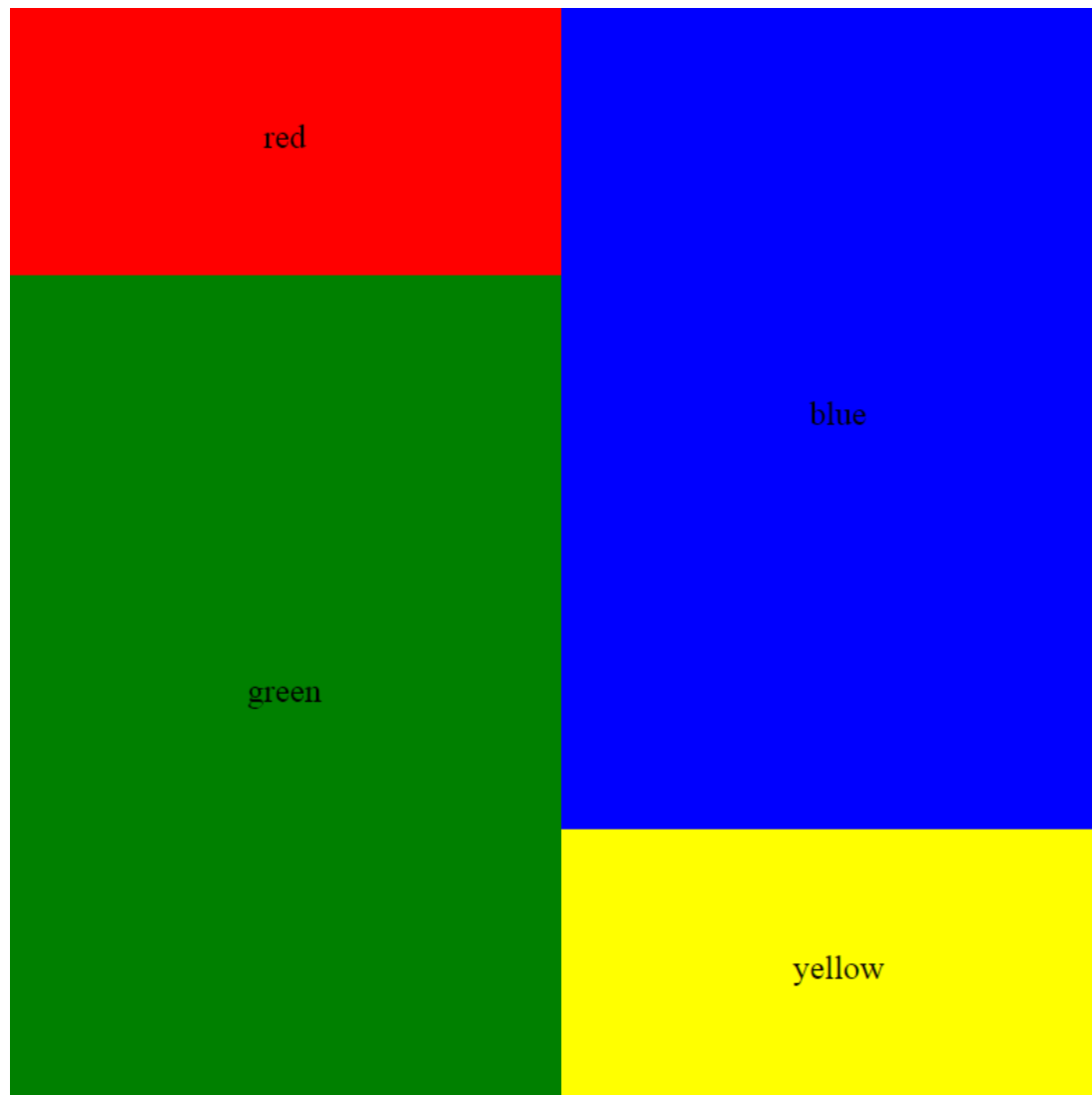


# 排版2

```
.grid {  
  width: 800px;  
  height: 800px;  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-template-rows: repeat(2, 1fr);  
  grid-template-areas:  
    "red blue blue"  
    "green green yellow"  
  ;  
}
```



# 排版3

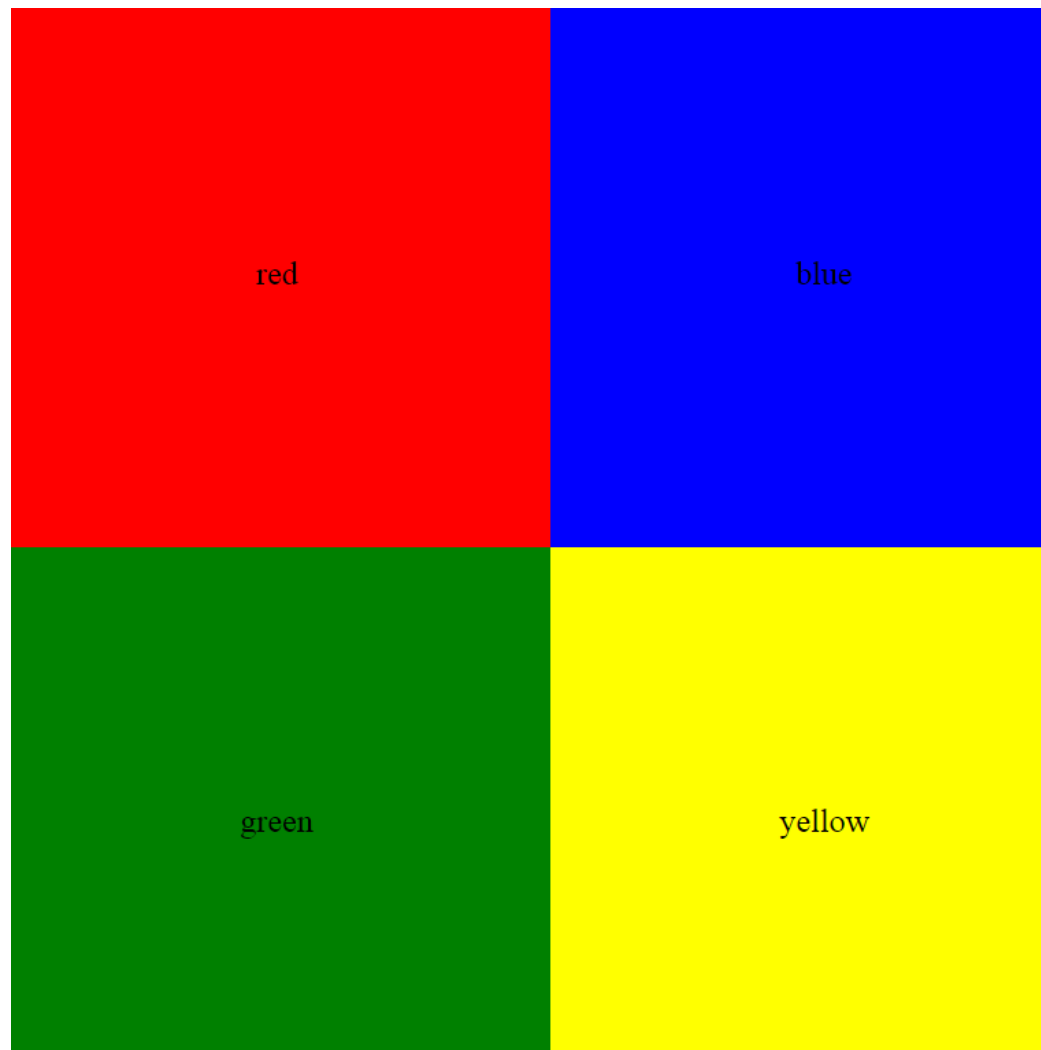


# 排版3

```
.grid {  
  width: 800px;  
  height: 800px;  
  display: grid;  
  grid-template-columns: repeat(4, 1fr);  
  grid-template-rows: repeat(4, 1fr);  
  grid-template-areas:  
    "red red blue blue"  
    "green green blue blue"  
    "green green blue blue"  
    "green green yellow yellow"  
  ;  
}
```

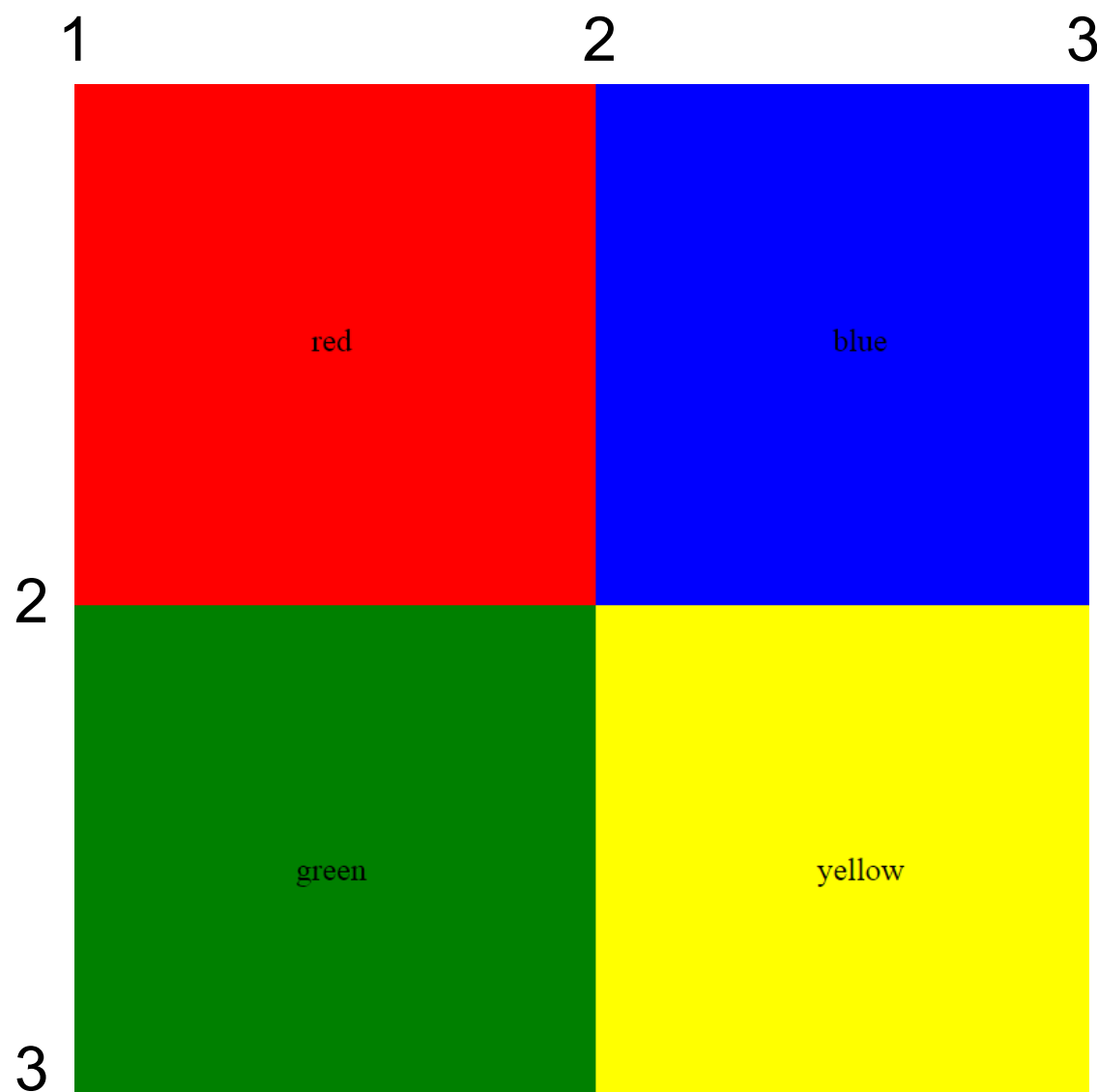
## 基礎用法2

```
.grid {  
  width: 800px;  
  height: 800px;  
  display: grid;  
  grid-template-columns: 1fr 1fr 2fr;  
  grid-template-rows: 2fr 2fr;  
  grid-template-areas:  
    "red red blue"  
    "green green yellow"  
  ;  
}
```





# 基礎用法3 / 子層指定



## 基礎用法3 / 子層指定

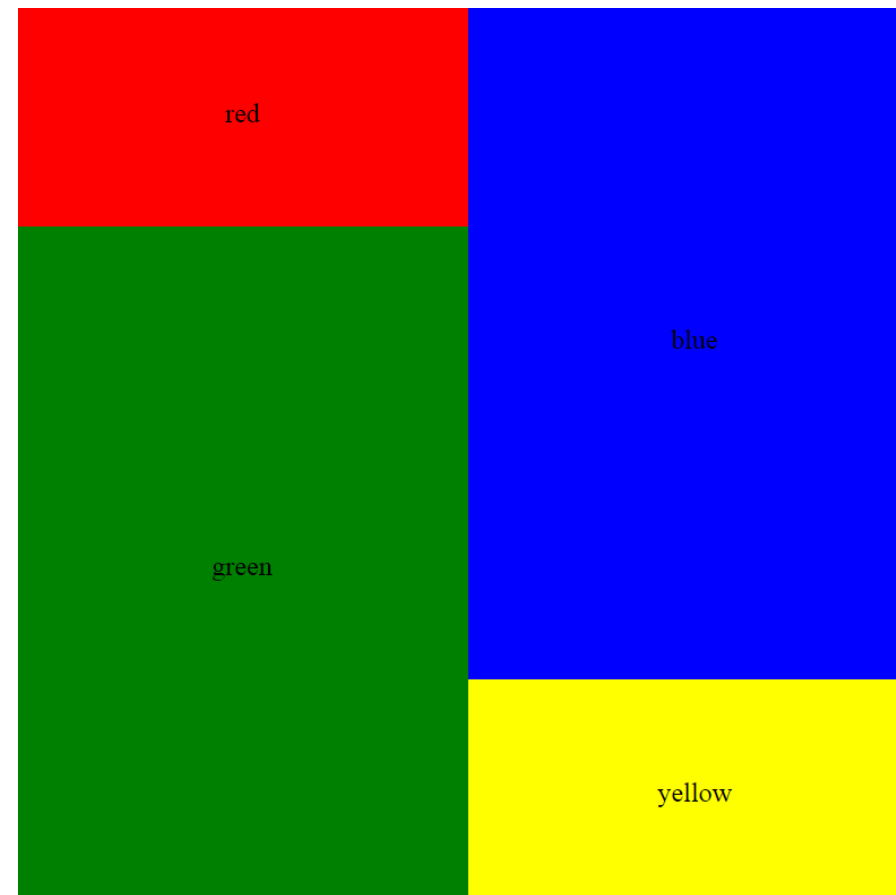
```
.grid {  
  width: 800px;  
  height: 800px;  
  display: grid;  
  grid-template-columns: repeat(2,1fr);  
  grid-template-rows: repeat(2,1fr);  
}
```

```
.red {  
  background-color: red;  
  grid-area: red;  
  grid-column: 1/2;  
  grid-row: 1/2;  
}  
  
.blue {  
  background-color: blue;  
  grid-area: blue;  
  grid-column: 2/3;  
  grid-row: 1/2;  
}  
  
.green {  
  background-color: green;  
  grid-area: green;  
  grid-column: 1/2;  
  grid-row: 2/3;  
}  
  
.yellow {  
  background-color: yellow;  
  grid-area: yellow;  
  grid-column: 2/3;  
  grid-row: 2/3;  
}
```

## 基礎用法3 / 子層指定

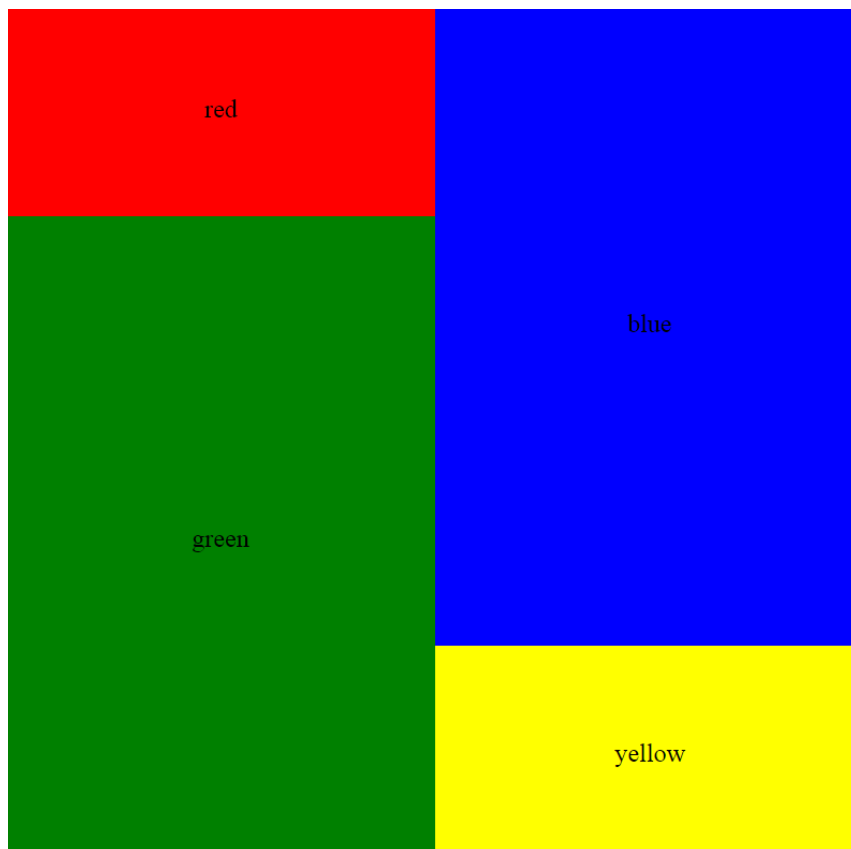
```
.red {  
  background-color: red;  
  grid-area: red;  
  grid-column: 1/3;  
  grid-row: 1/2;  
}  
  
.blue {  
  background-color: blue;  
  grid-area: blue;  
  grid-column: 3/5;  
  grid-row: 1/4;  
}
```

```
.green {  
  background-color: green;  
  grid-area: green;  
  grid-column: 1/3;  
  grid-row: 2/5;  
}  
  
.yellow {  
  background-color: yellow;  
  grid-area: yellow;  
  grid-column: 3/5;  
  grid-row: 4/5;  
}
```

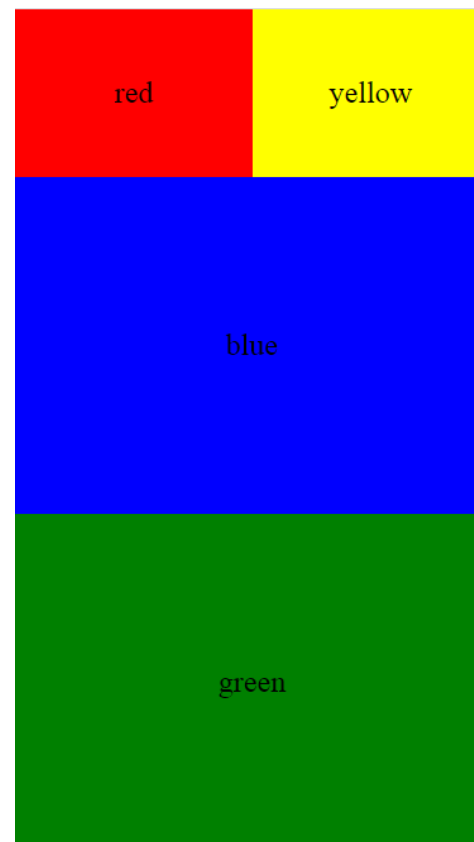


# 響應式排版

Desktop



Mobile



# 響應式排版

```
@media (max-width:768px) {  
  .grid {  
    width: 100%;  
    height: auto;  
    grid-auto-columns: repeat(2, 1fr);  
    grid-auto-rows: repeat(5, 1fr);  
    gap: 0;  
  }  
  .red {  
    grid-column: 1/2;  
    grid-row: 1/2;  
  }  
  
  .blue {  
    grid-column: 1/3;  
    grid-row: 2/4;  
  }  
}
```

```
.green {  
  grid-column: 1/3;  
  grid-row: 4/6;  
}  
  
.yellow {  
  grid-column: 2/3;  
  grid-row: 1/2;  
}  
div {  
  min-height: calc(100vh / 5);  
}  
}
```