

# Slidev 進階功能完全指南

探索 Slidev 的所有高級特性

從基礎到專家級功能的完整演示



高性能



可定制



互動式

開始探索 →

# 功能導覽

1. Slidev 進階功能完全指南
2. 功能導覽
3. 拖拽功能展示
4. 高級動畫系統
5. 互動式程式碼執行
6. 自定義組件生態
7. 複雜資料視覺化
8. 進階數學與科學公式
9. 多語言程式碼展示
10. 響應式設計與互動
11. 實時程式碼協作
12. 進階佈局技巧
13. 多媒體整合
14. 進階功能檢查清單

# 拖拽功能展示

Slidev 支持完全可拖拽的元素



這個元素可以自由移動

## 實現方式

```
<template>
  <!-- 在 frontmatter 定義位置 -->
  <v-drag pos="demo" class="...">
    拖拽內容
  </v-drag>

  <!-- 拖拽箭頭 -->
  <v-drag-arrow two-way />
</template>
```

```
---
dragPos:
  demo: x,y,width,height,rotate
---
```

# 高級動畫系統

使用 @vueuse/motion 創建複雜動畫

彈跳動畫

縮放動畫

滑動動畫

```
<div
  v-motion
  :initial="{ y: -100, opacity: 0, rotate: -90 }"
  :enter="{
    y: 0, opacity: 1, rotate: 0,
    transition: { duration: 800, type: 'spring' }
  }">
  動畫內容
</div>
```

# 互動式程式碼執行

Monaco Editor 整合，支持即時執行

```
// TypeScript/JavaScript 即時執行環境
interface User {
  id: number;
  name: string;
  age: number;
}

const users: User[] = [
  { id: 1, name: 'Alice', age: 25 },
  { id: 2, name: 'Bob', age: 30 },
  { id: 3, name: 'Charlie', age: 35 }
];

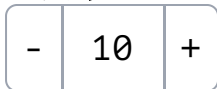
// 試試修改這些程式碼並執行！
function findAdults(users: User[], minAge: number = 18): User[] {
  return users.filter(user => user.age >= minAge);
}

function getUserStats(users: User[]) {
  const totalAge = users.reduce((sum, user) => sum + user.age, 0);
  return {
    count: users.length,
    averageAge: Math.round(totalAge / users.length),
    ages: users.map(u => u.age).sort((a, b) => a - b)
  };
}
```



# 自定義組件生態

## 計數器組件



### 進階互動示範



滑鼠位置: X: 0, Y: 0

這是一個展示 Vue 組件和響應式資料的示例



## 組件實現

```
<script setup lang="ts">
import { ref } from 'vue'
import { useToggle, useMouse, useWindowSize } from '@vueu

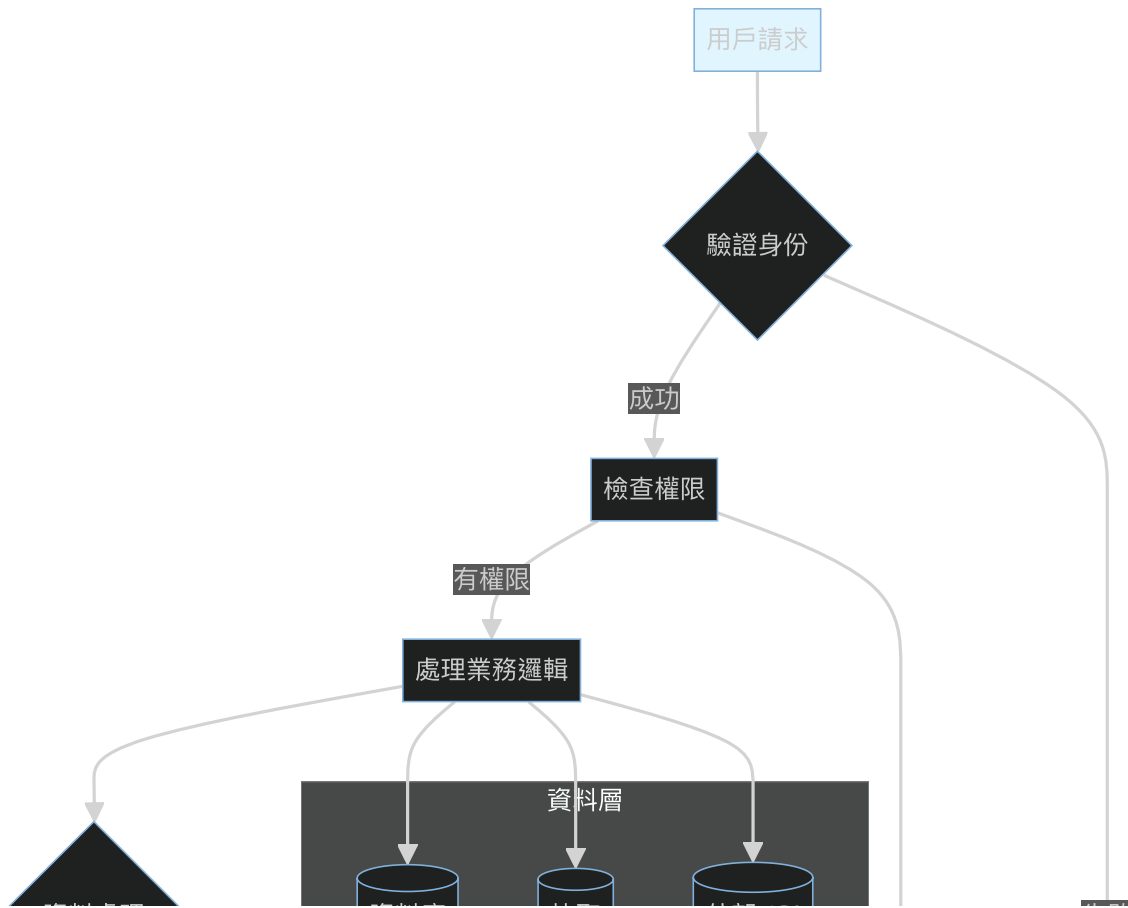
const props = defineProps<{
  count: number
}>()

const counter = ref(props.count)
const [isDark, toggle] = useToggle()

const { x, y } = useMouse()
const { width, height } = useWindowSize()
</script>

<template>
  <div class="space-y-4">
    <!-- 計數器 -->
    <div flex="~" w="min" border="~ rounded-md"
      :class="isDark ? 'border-white bg-gray-800' : 'b
    <button @click="counter -= 1" class="btn-counter">-
```

# 複雜資料視覺化



# 進階數學與科學公式

## 量子力學

薛丁格方程式：

$$i\hbar \frac{\partial}{\partial t} |\Psi(t)\rangle = \hat{H} |\Psi(t)\rangle$$

波函數歸一化：

$$\int_{-\infty}^{\infty} |\Psi(x, t)|^2 dx = 1$$

## 機器學習

損失函數（交叉熵）：

$$L = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C y_{i,j} \log(p_{i,j})$$

## 複雜矩陣運算

特徵值分解：

$$A = Q\Lambda Q^{-1}$$

其中  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$

## 傅立葉變換

連續傅立葉變換：

$$\mathcal{F}\{f(t)\} = \int_{-\infty}^{\infty} f(t) e^{-i2\pi ft} dt$$

離散傅立葉變換：

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N}$$



# 多語言程式碼展示

```
// Rust - 系統程式設計
use std::collections::HashMap;

#[derive(Debug, Clone)]
struct User {
    id: u32,
    name: String,
    email: String,
}

impl User {
    fn new(id: u32, name: &str, email: &str) -> Self {
        Self {
            id,
            name: name.to_string(),
            email: email.to_string(),
        }
    }

    fn is_valid_email(&self) -> bool {
        self.email.contains('@') && self.email.contains('.')
    }
}

fn main() {
```

```
# Python - 資料科學與機器學習
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

class DataProcessor:
    def __init__(self, data_path: str):
        self.data = pd.read_csv(data_path)
        self.model = None

    def preprocess(self):
        # 處理缺失值
        self.data.fillna(self.data.mean(), inplace=True)

        # 特徵縮放
        from sklearn.preprocessing import StandardScaler
        scaler = StandardScaler()
        return scaler.fit_transform(self.data)

    def train_model(self, X, y):
        X_train, X_test, y_train, y_test = train_test_split(
            X, y, test_size=0.2, random_state=42
        )
```

[plugin:vite:vue] Error parsing JavaScript expression: Identifier directly after number. (1:12)

[/@slidev/slides/11.md:5:1046](#)

```
3 |   <h1>實時程式碼協作</h1>
4 |   <p>Monaco Editor 支持多種進階功能 : </p>
5 |   <Monaco code-lz="PTAEmdlQhM1AVBPADgUwMoGMB0BLBAXUgehmdR6oEVWgp+YBQW/
  |
6 |   </InjectedLayout>
7 |   </template>
```

```
at createCompilerError (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+co
at emitError (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+compiler-co
at createExp (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+compiler-co
at Object.onattribend (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+co
at Tokenizer.handleInAttrValue (/Users/huai/side_project/mono-slidev/node_modules/.pnp
at Tokenizer.stateInAttrValueDoubleQuotes (/Users/huai/side_project/mono-slidev/node m
at Tokenizer.parse (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+comp
at Object.baseParse (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+comp
at Object.parse (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+compiler
at Object.parse$1 [as parse] (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/
```

Click outside, press **Esc** key, or fix the code to dismiss.

You can also disable this overlay by setting `server.hmr.overlay` to `false` in

[plugin:vite:vue] Error parsing JavaScript expression: Identifier directly after number. (1:12)

[/@slidev/slides/11.md:5:1046](#)

```
3 |   <h1>實時程式碼協作</h1>
4 |   <p>Monaco Editor 支持多種進階功能 : </p>
5 |   <Monaco code-lz="PTAEmdlQhM1AVBPADgUwMoGMB0BLBAXUgehmdR6oEVWgp+YBQW/
  |
6 |   </InjectedLayout>
7 |   </template>
```

```
at createCompilerError (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+co
at emitError (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+compiler-co
at createExp (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+compiler-co
at Object.onattribend (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+co
at Tokenizer.handleInAttrValue (/Users/huai/side_project/mono-slidev/node_modules/.pnp
at Tokenizer.stateInAttrValueDoubleQuotes (/Users/huai/side_project/mono-slidev/node_m
at Tokenizer.parse (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+comp
at Object.baseParse (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+comp
at Object.parse (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/@vue+compiler
at Object.parse$1 [as parse] (/Users/huai/side_project/mono-slidev/node_modules/.pnpm/
```

Click outside, press **Esc** key, or fix the code to dismiss.

You can also disable this overlay by setting `server.hmr.overlay` to `false` in

# 進階佈局技巧

主要功能

核心特性展示

輔助功能

增強用戶體驗

特殊功能

獨特價值提供

2021

● Slidev 發布

2022

● 社群生態建立

2023

● 企業級採用

# 多媒體整合

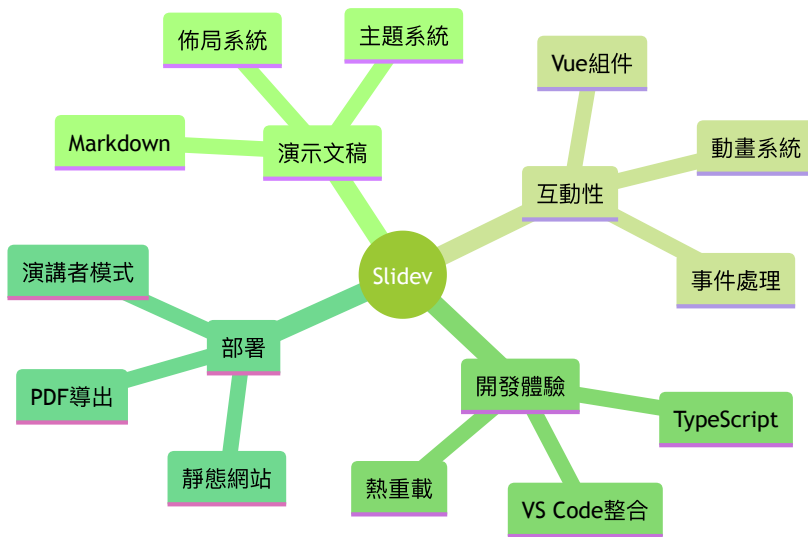
## 嵌入影片



## 社交媒體



## 進階圖表



# 進階功能檢查清單



## 核心進階功能

- ☒ 自定義組件開發
- ☒ 複雜動畫系統
- ☒ 拖拽互動元素
- ☒ 即時程式碼執行
- ☒ 響應式佈局
- ☒ 多媒體整合



## 進階工具





- ☒ Monaco Editor 整合
- ☒ Mermaid 圖表
- ☒ LaTeX 數學公式
- ☒ TypeScript 支援
- ☒ 繪圖功能
- ☒ 錄製與分享



你已掌握 Slidev 的所有進階功能！

# 謝謝觀看！

## 探索更多 Slidex 資源

-  官方文檔 - 完整的功能指南
-  主題畫廊 - 精美的主題集合
-  插件生態 - 強大的擴展功能
-  社群討論 - 獲得幫助與分享



Star on GitHub

-----/