

### 3. 製作todos lists 工具

功能說明:

1. 輸入代辦事項點擊新增或enter鍵後會新增資料並顯示於下方list-group
2. 完成事項check勾選後移至進行中或已完成
3. 點擊右方刪除鈕可刪除資料
4. 快速點擊兩下可以編輯資料，enter完成編輯，esc放棄並放棄編輯
5. 點擊nav "全部" "進行中" "已完成" 會顯示不同代辦事項
6. 即時顯示未完成任務的比數
7. 清除所有任務

代辦事項

準備要做的任務

新增

全部

進行中

已完成

☐ 梨泰院

×

☐ 李屍朝鮮

×

☐ 讀書

×

☐ 做專題

×

還有 4 筆任務未完成

清除所有任務

#### 1. 基本架構

代辦事項

準備要做的任務

新增

全部

進行中

已完成

☐ 代辦事項

×

還有 99 筆任務未完成

清除所有任務

```

<div id="app" class="container mt-5">
  <div class="row">
    <div class="col-md-8 offset-2">
      <div class="input-group mb-3">...
    </div>
    <div class="card">
      <div class="card-header bg-light">...
    </div>
    <div class="card-body">...
    </div>
    <div class="card-footer d-flex justify-content-between bg-light">...
    </div>
  </div>
</div>
</div>

```

### input-group

```

<div class="input-group mb-3">
  <div class="input-group-prepend">
    <span class="input-group-text">代辦事項</span>
  </div>
  <input type="text" class="form-control" placeholder="準備要做的任務">
  <div class="input-group-append">
    <button class="btn btn-outline-primary" type="button" id="button-addon2">新增</button>
  </div>
</div>

```

### card-header

```

<div class="card-header bg-light">
  <ul class="nav nav-tabs">
    <li class="nav-item">
      <a class="nav-link active">全部</a>
    </li>
    <li class="nav-item">
      <a class="nav-link">進行中</a>
    </li>
    <li class="nav-item">
      <a class="nav-link">已完成</a>
    </li>
  </ul>
</div>

```

### card-body

```

<div class="card-body">
  <ul class="list-group">
    <li class="list-group-item">
      <div class="d-flex">
        <div class="form-check">
          <input class="form-check-input" type="checkbox">
          <label class="form-check-label" >
            代辦事項
          </label>
        </div>
        <button type="button" class="close ml-auto" aria-label="Close">
          <span aria-hidden="true">&times;</span></button>
        </div>
        <input type="text" class="form-control">
      </li>
    </ul>
  </div>

```

### card-footer

```

<div class="card-footer d-flex justify-content-between bg-light">
  <span>還有 99 筆任務未完成</span>
  <a href="#">清除所有任務</a>
</div>

```

## 2. Vue 架構 - 綁定 input

```

<script>
  const App = {
    data() {
      return {
        newTodo: '預算審查會議!',
        visibility: 'all',
        cacheTitle: '',
        cacheItem: [],
        todos: [
          {
            id: '001',
            title: '線上學習',
            completed: false,
          }
        ]
      }
    },
    methods: { ...
    computed: { ...
  }

  Vue.createApp(App).mount('#app');
</script>

```

### 綁定 input

```

<div class="input-group mb-3">
  <div class="input-group-prepend">
    <span class="input-group-text">代辦事項</span>
  </div>
  <input type="text" class="form-control" v-model="newTodo" @keyup.enter="addTodo">
  <div class="input-group-append">
    <button class="btn btn-outline-primary" @click="addTodo">新增</button>
  </div>
</div>

```

代辦事項 預算審查會議! 新增

全部 進行中 已完成

☐ 線上學習 x

還有1比任務 清楚所有任務

Find components...

<Root> 33 ms

<Root> Filter state...

- data
  - cacheItem: Array[0]
  - cacheTitle: ""
  - newTodo: "預算審查會議!"
  - todos: Array[1]
    - 0: Object
    - visibility: "all"
- computed
  - countActive: 1
  - filterTodo: Array[1]
    - 0: Object

3. 新增至todos 並以v-for將資料顯示出

```

methods: {
  addTodo() {
    const timestamp = Math.floor(Date.now());
    const value = this.newTodo.trim();
    if(!value){
      return;
    }
    this.todos.push({
      id: timestamp,
      title: value,
      completed: false
    });
    this.newTodo = '';
  },
  removeTodo(todo) { ...
  editTodo(item) { ...
  cancelTodo() { ...
  doneTodo(item) { ...
  destory() { ...
},

```

### 監聽觸發事件

```

<div class="input-group mb-3">
  <div class="input-group-prepend">
    <span class="input-group-text">代辦事項</span>
  </div>
  <input type="text" class="form-control" v-model="newTodo" @keyup.enter="addTodo">
  <div class="input-group-append">
    <button class="btn btn-outline-primary" @click="addTodo">新增</button>
  </div>
</div>

```

### v-for將資料顯示出

```

<ul class="list-group">
  <li class="list-group-item" v-for="(item, key) in filterTodo">
    <div class="d-flex" v-if="item.id !== cacheItem.id">
      <div class="form-check">
        <input type="checkbox" class="form-check-input" v-model="item.completed">
        <label class="form-check-label" :for="item.id"
          :class="{ 'completed': item.completed}" @dblclick="editTodo(item)">
          {{ item.title }}
        </label>
      </div>
      <button type="button" class="close ml-auto" data-dismiss="modal" aria-label="Close"
        @click="removeTodo(item)">...
      </button>
    </div>
    <input type="text" name="" id="" class="form-control" ...
      @keyup.enter="doneTodo(item)">
  </li>
</ul>

```

代辦事項

新增

全部

進行中

已完成

☐ 線上學習

×

☐ 預算審查會議!

×

☐ 專題會議

×

還有3比任務

清楚所有任務

#### 4. 刪除事件

```

removeTodo(todo) {
  const vm = this;
  var newIndex = vm.todos.findIndex((item, index) => {
    return todo.id === item.id
  });

  vm.todos.splice(newIndex, 1);
},

```

```

<button type="button" class="close ml-auto" data-dismiss="modal" aria-label="Close"
  @click="removeTodo(item)">
  <span aria-hidden="true">&times;</span>
</button>

```



## 5. 刪除線

判斷checkbox是否有選取來決定是否套用class

```
<style>
  .completed{
    text-decoration: line-through;
    color: ■ crimson;
  }
</style>
```

必須綁定checkbox 與item.completed

```
<div class="form-check">
  <input type="checkbox" class="form-check-input" v-model="item.completed">
  <label class="form-check-label" :for="item.id"
    :class="{ 'completed' : item.completed}" @dblclick="editTodo(item)">
    {{ item.title }}
  </label>
</div>
```

代辦事項

新增

全部

進行中

已完成

☒ 線上學習

×

☐ 預算審查會議!

×

☒ 專題會議

×

還有1比任務

清楚所有任務

## 6. 切換nav與顯示的資料



代辦事項 李屍朝鮮 新增

全部 進行中 已完成

☐ 梨泰院

還有 99 筆任務未完成 清除所有任務

```
<div class="card-header">
  <ul class="nav nav-tabs">
    <li class="nav-item">
      <a class="nav-link" href="#" :class="{ 'active' : visibility==='all'}"
        @click="visibility='all'">全部</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#" :class="{ 'active' : visibility==='active'}"
        @click="visibility='active'">進行中</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="#" :class="{ 'active' : visibility==='completed'}"
        @click="visibility='completed'">已完成</a>
    </li>
  </ul>
</div>
```

需搭配 visibility 來計入目前顯示的類型!

```
data() {
  return {
    newTodo: '預算審查會議!',
    visibility: 'all',
    cacheTitle: '',
    cacheItem: [],
    todos: [
      {
        id: '001',
        title: '線上學習',
        completed: false,
      }
    ]
  }
},
```

使用computed 建立 filterTodo 回傳過濾的資料結果!

```
computed: {  
  filterTodo() {  
    const vm = this;  
    if(vm.visibility === 'all'){  
      return vm.todos;  
    }else if(vm.visibility === 'active'){  
      const activeTodo = [];  
      vm.todos.forEach((item, key) => {  
        if(!item.completed){  
          activeTodo.push(item);  
        }  
      });  
      return activeTodo;  
    }else if (vm.visibility === 'completed') {  
      const activeTodo = [];  
      vm.todos.forEach((item, key) => {  
        if (item.completed) {  
          activeTodo.push(item);  
        }  
      });  
      return activeTodo;  
    }  
  },  
  countActive() { ...  
}
```

更新v-for 資料

```

<ul class="list-group">
  <li class="list-group-item" v-for="item in filterTodo">
    <div class="d-flex">
      <div class="form-check">
        <input class="form-check-input" type="checkbox" :id="item.title" v-model="item.completed">
        <label class="form-check-label" :for="item.title" :class="{ 'completed': item.completed }">
          {{ item.title }}
        </label>
      </div>
      <div>
        <button type="button" class="close ml-auto" aria-label="Close" @click="removeTodo(item)">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
    </div>
    <input type="text" class="form-control">
  </li>
</ul>

```

代辦事項
準備要做的任務
新增

全部
進行中
已完成

☒ 梨泰院

☒ 李屍朝鮮

☐ 做專題

還有 99 筆任務未完成
清除所有任務

代辦事項
準備要做的任務
新增

全部
進行中
已完成

☐ 做專題

還有 99 筆任務未完成
清除所有任務

代辦事項 準備要做的任務 新增

全部 進行中 已完成

☒ 梨泰院

☒ 李屍朝鮮

還有 99 筆任務未完成 清除所有任務

## 7. 編輯代辦事項

利用`cacheTitle`和`cacheItem`來製作編輯的狀態與功能

```
data: {
  newTodo: "李屍朝鮮",
  todos: [
    {
      id: '001',
      title: '梨泰院',
      completed: false,
    }
  ],
  visibility: 'all',
  cacheTitle: '',
  cacheItem: [],
},
```

```
editTodo(item) {
  this.cacheItem = item;
  this.cacheTitle = item.title;
},
cancelTodo() {
  this.cacheTitle = '';
  this.cacheItem = '';
},
doneTodo(item) {
  item.title = this.cacheTitle;
  this.cacheTitle = '';
  this.cacheItem = '';
},
```

## 監聽與判斷事件

```
<ul class="list-group">
  <li class="list-group-item" v-for="(item, key) in filterTodo">
    <div class="d-flex" v-if="item.id !== cacheItem.id">
      <div class="form-check">
        <input type="checkbox" class="form-check-input" v-model="item.completed">
        <label class="form-check-label" :for="item.id"
          :class="{ 'completed' : item.completed}" @dblclick="editTodo(item)">
          {{ item.title }}
        </label>
      </div>
      <button type="button" class="close ml-auto" data-dismiss="modal" aria-label="Close"
        @click="removeTodo(item)">
        <span aria-hidden="true">&times;</span>
      </button>
    </div>
    <input type="text" name="" id="" class="form-control"
      v-if="item.id === cacheItem.id"
      @keyup.esc="cancelTodo"
      v-model="cacheTitle"
      @keyup.enter="doneTodo(item)">
    </li>
</ul>
```

## 8. 計算代辦事項

```
countActive() {
  let active = 0;
  this.todos.forEach((item, key) => {
    if(!item.completed){
      active ++;
    }
  });
  return active;
}
```

```
<div class="card-footer d-flex justify-content-between">
  <span>還有{{ countActive }}筆任務</span>
  <a href="#" @click="destory">清楚所有任務</a>
</div>
```

代辦事項

準備要做的任務

新增

全部

進行中

已完成

☒ 梨泰院

×

☐ 李屍朝鮮

×

☐ 做專題

×

還有 2 筆任務未完成

清除所有任務

## 9. 清除所有任務

```
<div class="card-footer d-flex justify-content-between bg-light">  
  <span>還有 {{filterActive}} 筆任務未完成</span>  
  <a href="#" @click="destory">清除所有任務</a>  
</div>
```

```
destory() {  
  this.todos = [];  
},
```