

22.練習 (四) - Todo List

一、HTML

```
<header>
|   <h1>Todo List</h1>
</header>

<form>
|   <input type="text">
|   <input type="number" min="2021" max="2050" placeholder="Y" required>
|   <input type="number" min="1" max="12" placeholder="M" required>
|   <input type="number" min="1" max="31" placeholder="D" required>
|   <button type="submit">Add into List</button>
</form>

<div class="sort">
|   <button>依預定時間排序</button>
</div>

<section></section>
<script src="./app.js"></script>
```

Todo List

二、CSS

1.背景與 header。

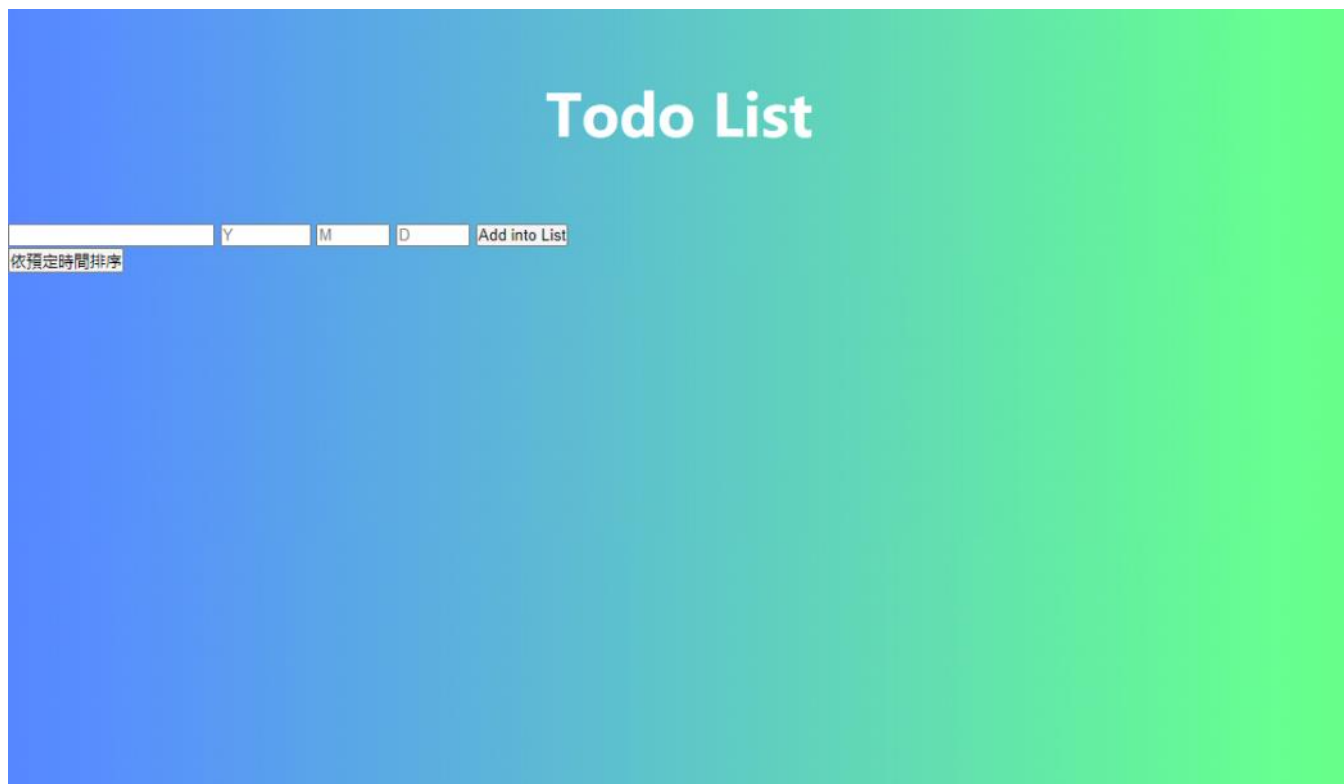
```

* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body {
  background: linear-gradient(90deg, rgb(84, 130, 255), rgb(100, 255, 134));
}

header {
  height: 20vh;
  display: flex;
  justify-content: center;
  align-items: center;
  h1 {
    color: white;
    font-size: 3rem;
  }
}

```



2.表單物件。

```

form {
  display: flex;
  justify-content: center;
  align-items: center;

  input,
  button {
    border: none;
    padding: 0.25rem 0.5rem;
    font-size: 1.75rem;
    margin: 0.1rem;
  }

  input[type="number"] {
    width: 100px;
    text-align: center;
  }

  button {
    cursor: pointer;
    background-color: #008b8b;
    color: white;
  }
}

```

Todo List

Y

M

D

Add into List

依預定時間排序

3.隱藏數字上下按鈕。

```

/* Chrome, Safari, Edge, Opera */
input::-webkit-outer-spin-button,
input::-webkit-inner-spin-button {
  -webkit-appearance: none;
  margin: 0;
}

/* Firefox */
input[type="number"] {
  -moz-appearance: textfield;
}

```

Todo List

依預定時間排序

三、JavaScript

1. 設定表單按鈕行為。

```

let add = document.querySelector("form button");
add.addEventListener("click", e => {
  e.preventDefault();
});

```

2. 取得表單資料，並新增至表單下方。

```

let section = document.querySelector("section");

```

```

let form = e.target.parentElement;
let todoText = form.children[0].value;
let todoYear = form.children[1].value;
let todoMonth = form.children[2].value;
let todoDate = form.children[3].value;

let todo = document.createElement("div");
todo.classList.add("todo");
let text = document.createElement("p");
text.classList.add("todo-text");
text.innerText = todoText;
let time = document.createElement("p");
time.classList.add("todo-time");
time.innerText = todoYear + "/" + todoMonth + " / " + todoDate;
todo.appendChild(text);
todo.appendChild(time);

section.appendChild(todo);

```

Todo List

test	2021	1	1	<input type="button" value="Add into List"/>
------	------	---	---	--

依預定時間排序

test
2021/1 / 1

3.調整資料區的樣式。

```

section {
  display: flex;
  flex-direction: column;
  align-items: center;

  div.todo {
    width: 50vw;
    display: flex;
    align-items: center;
    margin: 0.2rem 0rem;

    p {
      background-color: white;
      color: black;
      padding: 0.25rem 0.5rem;
      font-size: 1.75rem;
    }

    p.todo-text {
      width: 65%;
    }
    p.todo-time {
      width: 35%;
      text-align: center;
    }
  }
}

```

Todo List

test

2021

12

31

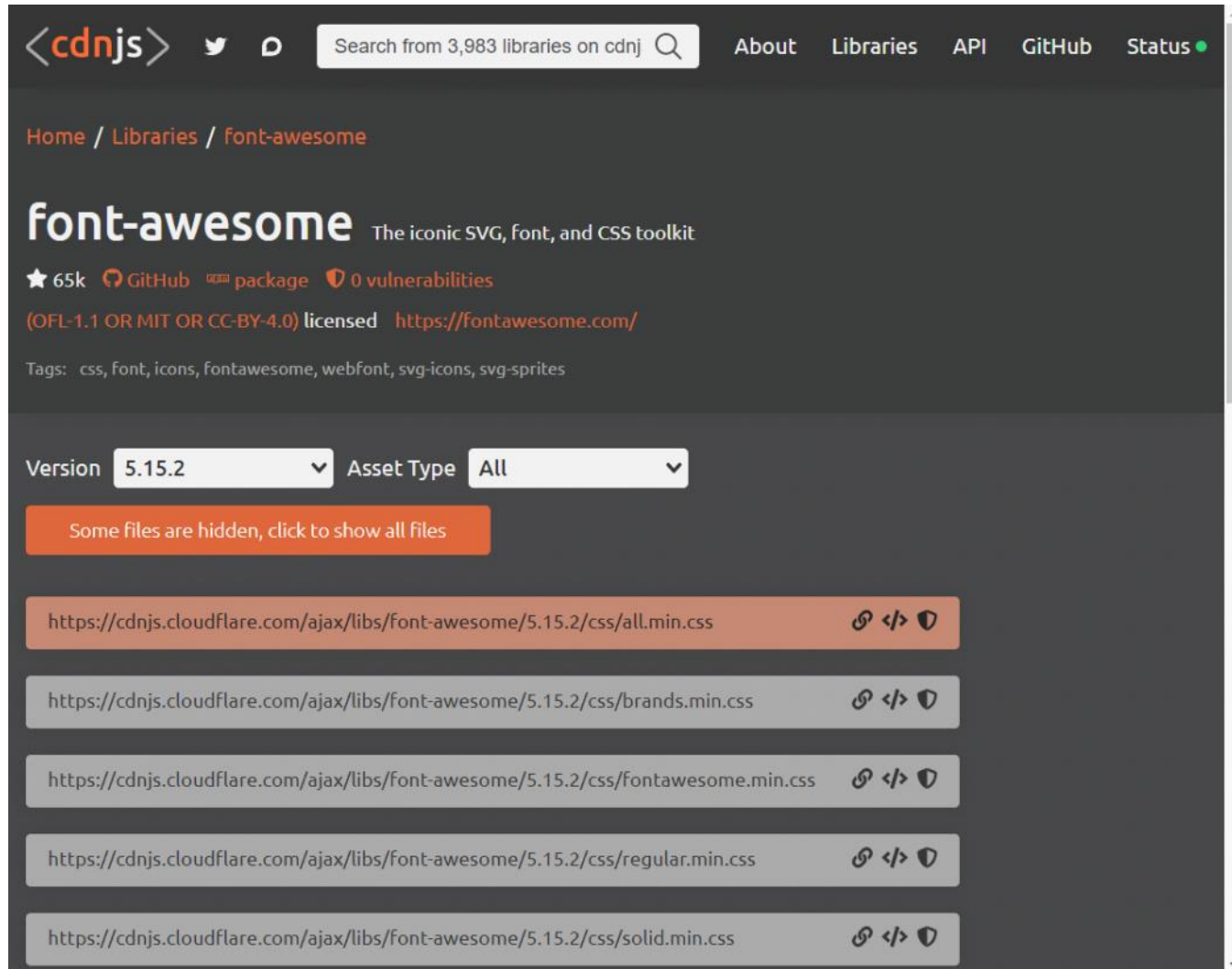
Add into List

依預定時間排序

test	2021/12 / 31
test	2021/12 / 31
test	2021/12 / 31

四、加入按鈕圖示

1.先到 [cdnjs](https://cdnjs.com) 官網複製 CSS 連結到 index.html。



2.設定按鈕圖示

```
let completeButton = document.createElement("button");
completeButton.classList.add("complete");
completeButton.innerHTML = '<i class="fas fa-check"></i>';

let trashButton = document.createElement("button");
trashButton.classList.add("trash");
trashButton.innerHTML = '<i class="fas fa-trash"></i>';

todo.appendChild(completeButton);
todo.appendChild(trashButton);
```



3.設定按鈕樣式

```
button {  
  padding: 0.25rem 0.5rem;  
  font-size: 1.75rem;  
  border: none;  
  cursor: pointer;  
  color: white;  
}  
  
button.complete {  
  background-color: rgb(50, 255, 50);  
}  
  
button.trash {  
  background-color: red;  
}
```




五、設定事項動畫特效與事件

1. 新增事項時彈出動畫。

```
@keyframes scaleUp {  
  from {  
    transform: scale(0);  
  }  
  to {  
    transform: scale(1);  
  }  
}
```

```
todo.style.animation = "scaleUp 0.3s forwards";
```



2.事項完成時標示記號。

```
completeButton.addEventListener("click", e => {
  let todoItem = e.target.parentElement;
  todoItem.classList.add("done");
})
```

```
.done {
  opacity: 0.5;
  text-decoration: line-through;
  text-decoration-thickness: 0.2rem;
}
```





```
transition: all 0.1s ease;
```

```
i {
  pointer-events: none;
}
```

Todo List

test	2021	1	1	Add into List
------	------	---	---	---------------

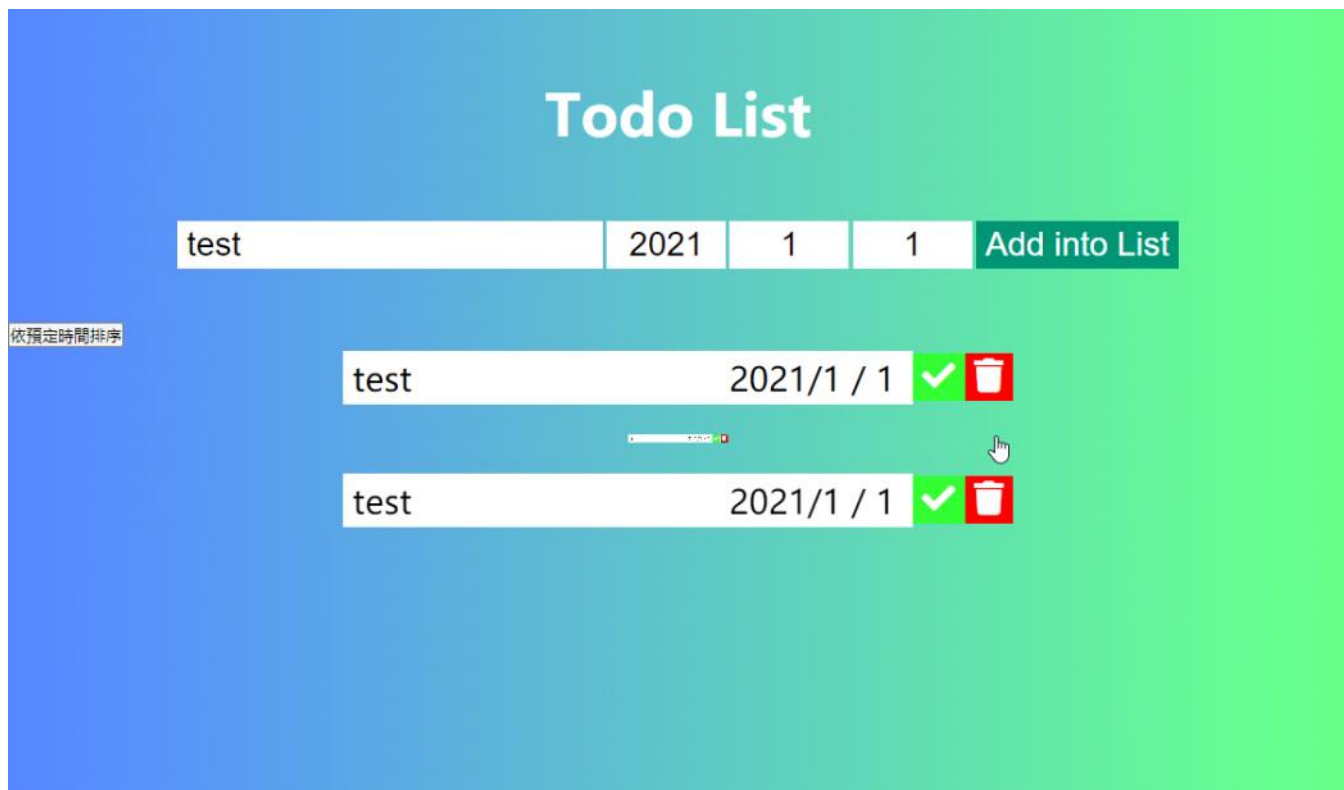
依預定時間排序

test	2021/1 / 1	✓	
test	2021/1 / 1	✓	
test	2021/1 / 1	✓	
test	2021/1 / 1	✓	

3.設定移除動畫與特效。

```
trashButton.addEventListener("click", e => {  
  let todoItem = e.target.parentElement;  
  todoItem.addEventListener("animationend", () => {  
    todoItem.remove();  
  })  
  todoItem.style.animation = "scaleDown 0.3s forwards";  
})
```

```
@keyframes scaleDown {  
  to {  
    transform: scale(0);  
  }  
}
```

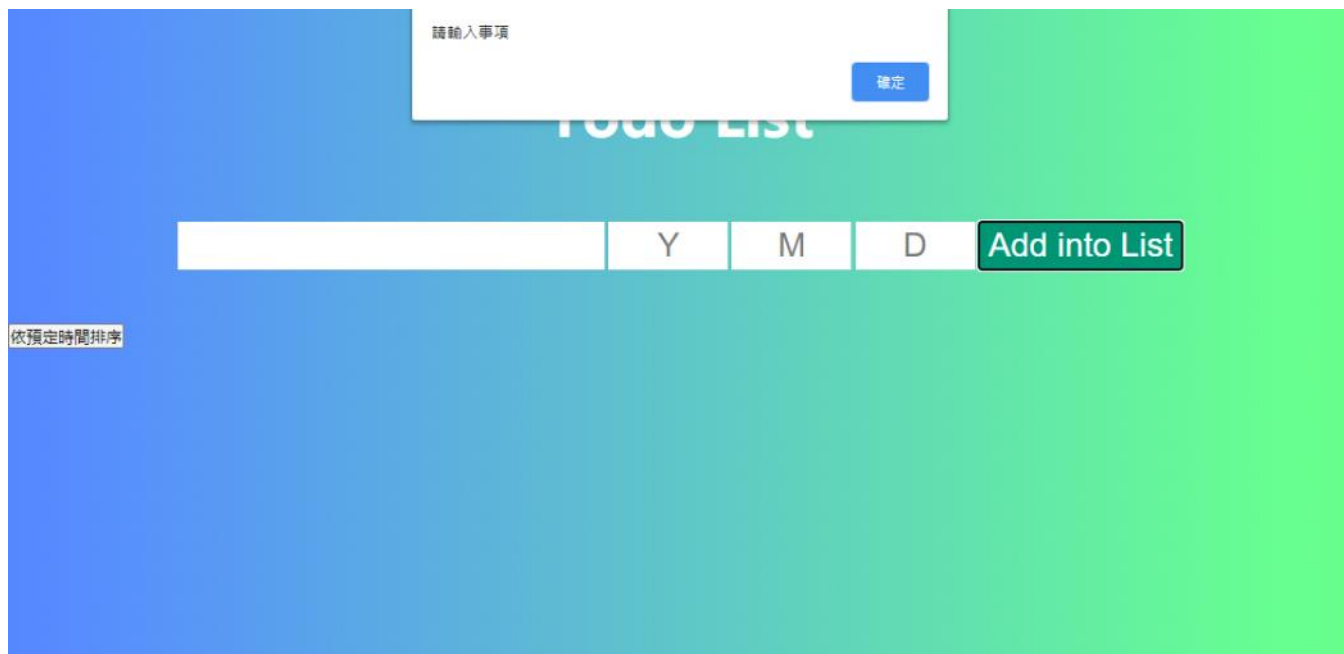


4.按下新增事項按鈕後清空輸入框

```
form.children[0].value = "";
```

5.提示未輸入資料

```
if (todoText === "") {
    alert("請輸入事項");
    return;
}
```



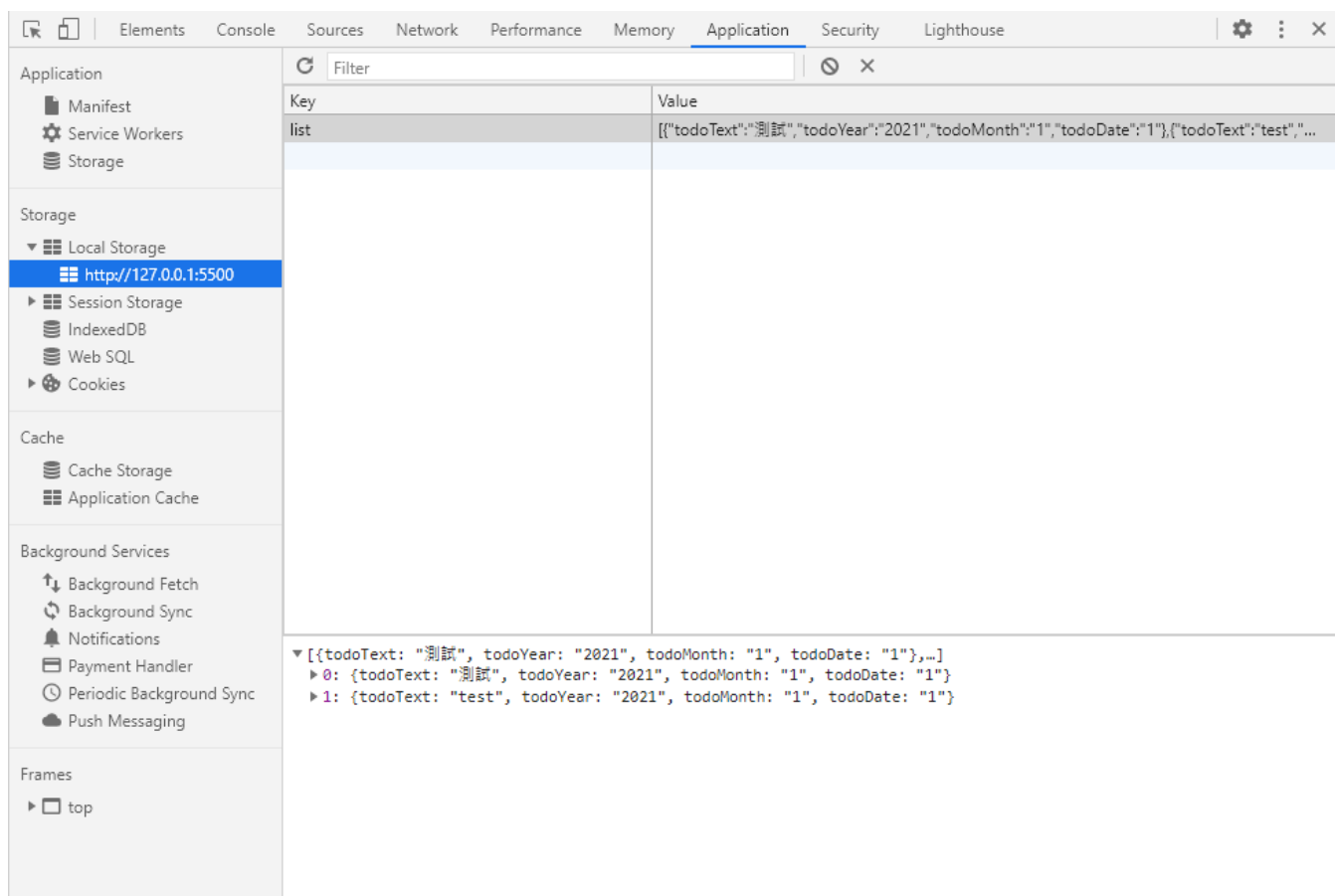
六、localStorage

1.將資料以陣列方式存入localStorage。

```
let myTodo = {
  todoText: todoText,
  todoYear: todoYear,
  todoMonth: todoMonth,
  todoDate: todoDate
};

let myList = localStorage.getItem("list");
if (myList == null) {
  localStorage.setItem("list", JSON.stringify([myTodo]));
} else {
  let myListArray = JSON.parse(myList);
  myListArray.push(myTodo);
  localStorage.setItem("list", JSON.stringify(myListArray));
}

console.log(JSON.parse(localStorage.getItem("list")));
```



2.重新開啟網頁時讀取localStorage資料。

```

let myList = localStorage.getItem("list");
if (myList !== null) {
  let myListArray = JSON.parse(myList);
  myListArray.forEach(item => {

    let todo = document.createElement("div");
    todo.classList.add("todo");
    let text = document.createElement("p");
    text.classList.add("todo-text");
    text.innerText = item.todoText;
    let time = document.createElement("p");
    time.classList.add("todo-time");
    time.innerText = item.todoYear + "/" + item.todoMonth + " / " + item.todoDate;
    todo.appendChild(text);
    todo.appendChild(time);
  });
}

```

```

let completeButton = document.createElement("button");
completeButton.classList.add("complete");
completeButton.innerHTML = '<i class="fas fa-check"></i>';
completeButton.addEventListener("click", e => {
  let todoItem = e.target.parentElement;
  todoItem.classList.toggle("done");
})

let trashButton = document.createElement("button");
trashButton.classList.add("trash");
trashButton.innerHTML = '<i class="fas fa-trash"></i>';
trashButton.addEventListener("click", e => {
  let todoItem = e.target.parentElement;
  todoItem.addEventListener("animationend", () => {
    todoItem.remove();
  })
  todoItem.style.animation = "scaleDown 0.3s forwards";
})

todo.appendChild(completeButton);
todo.appendChild(trashButton);

todo.style.animation = "scaleUp 0.3s forwards";

section.appendChild(todo);
})
}

```

3. 刪除localStorage資料。

```

let text = todoItem.children[0].innerText;
let myListArray = JSON.parse(localStorage.getItem("list"));
myListArray.forEach((item, index) => {
  if (item.todoText == text) {
    myListArray.splice(index, 1);
    localStorage.setItem("list", JSON.stringify(myListArray));
  }
})

```

七、依時間排序

1. 按鈕樣式

```

div.sort {
  display: flex;
  justify-content: center;
  align-items: center;
  margin: 1rem 0rem;

  button {
    cursor: pointer;
    background-color: #00728f;
    color: white;
    border: none;
    padding: 0.25rem 0.5rem;
    font-size: 1.75rem;
  }
}

```



2. 建立比較函式。

```

function mergeTime(arr1, arr2) {
  let result = [];
  let i = 0;
  let j = 0;
  while (i < arr1.length && j < arr2.length) {
    if (Number(arr1[i].todoYear) > Number(arr2[j].todoYear)) {
      result.push(arr2[j]);
      j++;
    } else if (Number(arr1[i].todoYear) < Number(arr2[j].todoYear)) {
      result.push(arr1[i]);
      i++;
    } else if (Number(arr1[i].todoYear) == Number(arr2[j].todoYear)) {
      if (Number(arr1[i].todoMonth) > Number(arr2[j].todoMonth)) {
        result.push(arr2[j]);
        j++;
      } else if (Number(arr1[i].todoMonth) < Number(arr2[j].todoMonth)) {
        result.push(arr1[i]);
        i++;
      } else if (Number(arr1[i].todoMonth) == Number(arr2[j].todoMonth)) {
        if (Number(arr1[i].todoDate) > Number(arr2[j].todoDate)) {
          result.push(arr2[j]);
          j++;
        } else {
          result.push(arr1[i]);
          i++;
        }
      }
    }
  }
}

while (i < arr1.length) {
  result.push(arr1[i]);
  i++;
}

while (j < arr2.length) {
  result.push(arr2[j]);
}

```



```

    j++;
  }
  return result;
}

function mergeSort(arr) {
  if (arr.length === 1) {
    return arr;
  } else {
    let middle = Math.floor(arr.length / 2);
    let right = arr.slice(0, middle);
    let left = arr.slice(middle, arr.length);
    return mergeTime(mergeSort(right), mergeSort(left));
  }
}

```

3.建立按鈕功能

```

let sortButton = document.querySelector("div.sort button");
sortButton.addEventListener("click", () => {

  let sortedArray = mergeSort(JSON.parse(localStorage.getItem("list")));
  localStorage.setItem("list", JSON.stringify(sortedArray));

  let len = section.children.length;
  for (let i = 0; i < len; i++) {
    section.children[0].remove();
  }

  loadData();
})

```

```

loadData();

function loadData() {
  let myList = localStorage.getItem("list");

```

Todo List

<input type="text"/>	Y	M	D	Add into List
----------------------	---	---	---	---------------

依預定時間排序

測試	2021/1 / 1	✓	🗑
html	2021/1 / 1	✓	🗑
test	2021/1 / 2	✓	🗑
JS	2022/1 / 1	✓	🗑