

FELADAT 1 (1 PONT)

Illesszen be egy képernyőképet néhány hozzáadott tennivalóról!



FELADAT 2 (1 PONT)

Illesszen be egy-egy képernyőképet a tennivalók állapotainak változtatásáról, a különböző oldalakon történő megjelenésükről!

The image displays four sequential screenshots of a web application titled "To-Do | Mobil- és webes szoftverek".

- First screenshot:** Shows the initial state with three tasks: "Kutya", "Macska", and "Alma". All tasks are under the "All" filter (3 items). Each task has a green checkmark, a plus sign, a minus sign, and a red trash icon.
- Second screenshot:** The "Macska" task's checkmark is now green, and a tooltip "Mark as done" is visible over the plus sign.
- Third screenshot:** A modal dialog box is shown with the text "127.0.0.1:8080 says Are you sure you want to delete the todo titled 'Kutya?'". The dialog has "Cancel" and "OK" buttons. The "Kutya" task's trash icon is highlighted in red.
- Fourth screenshot:** After deletion, the "All" filter now shows 2 items ("Macska" and "Alma"). The "Done" filter now shows 2 items ("Kutya" and "Macska").

FELADAT 3 (1.5 PONT)

Illesszen be egy-egy képernyőképet néhány tennivalóról sorrendezés előtt és után!

Illessze be a releváns kódrészleteket!

TO-DO | MOBIL- ÉS WEBES SZOFTVEREK

All 3

Active 3

Inactive

Done

Kutya	↑	↓	✓	+	-	🗑
Macska	↑	↓	✓	+	-	🗑
Alma	↑	↓	✓	+	-	🗑

All 3

Active 3

Inactive

Done

Macska	↑	↓	✓	+	-	🗑
Kutya	↑	↓	✓	+	-	🗑
Alma	↑	↓	✓	+	-	🗑

All 3

Active 3

Inactive

Done

Macska	↑	↓	✓	+	-	🗑
Alma	↑	↓	✓	+	-	🗑
Kutya	↑	↓	✓	+	-	🗑

```

buttons.forEach(function (button) { // a gomb modellek alapján legvártjuk a DOM gombokat
  if((button.action === "move-up" || button.action === "move-down") && currentTab !== "all"){
    return;
  }
  var btn = document.createElement("button"); // <button>
  btn.className = "btn btn-outline-" + button.type + " fas fa-" + button.icon;
  btn.title = button.title;
  if (todo.state === button.action) // azt a gombot letiltjuk, amelyen állapotban van egy elem
    btn.disabled = true;
  if(todos.indexOf(todo) === 0 && button.action === "move-up"){
    btn.disabled = true;
  }
  if(todos.indexOf(todo) === todos.length-1 && button.action === "move-down"){
    btn.disabled = true;
  }
  btn.onclick = button.action === "remove"
    ? function () { // klikk eseményre megerősítés után eltávolítjuk a to-do-t
        if (confirm("Are you sure you want to delete the todo titled '" + todo.name + "'?")) {
          todos.splice(todos.indexOf(todo), 1); // kiveszünk a 'todo'-adik elemről 1 elemet a todos tömbből
          renderTodos();
        }
      }
    : button.action === "move-up" ? _ => {
        const index = todos.indexOf(todo);
        const element = todo;
        todos.splice(index, 1);
        todos.splice(index-1, 0, element);
        renderTodos();
      }
    : button.action === "move-down" ? _ => {
        const index = todos.indexOf(todo);
        const element = todo;
        todos.splice(index, 1);
        todos.splice(index+1, 0, element);
        renderTodos();
      }
    : function () { // klikk eseményre beállítjuk a to-do állapotát a gomb által reprezentált állapotra
        todo.state = button.action;
        renderTodos();
      }
  buttonContainer.appendChild(btn); // a <div>-be tesszük a gombot
});

```

Csak az all fűn való kirenderelés:

```

if((button.action === "move-up" || button.action === "move-down") && currentTab !== "all"){
  return;
}

```

A gombok letiltása:

```

if(todos.indexOf(todo) === 0 && button.action === "move-up"){
  btn.disabled = true;
}
if(todos.indexOf(todo) === todos.length-1 && button.action === "move-down"){
  btn.disabled = true;
}

```

A sorrendcsere:

```

: button.action === "move-up" ? _ => {
  const index = todos.indexOf(todo);
  const element = todo;
  todos.splice(index, 1);
  todos.splice(index-1, 0, element);
  renderTodos();
}
: button.action === "move-down" ? _ => {
  const index = todos.indexOf(todo);
  const element = todo;
  todos.splice(index, 1);
  todos.splice(index+1, 0, element);
  renderTodos();
}

```

FELADAT 4 (1.5 PONT)

Illesszen be egy képernyőképet a lokális tárolóban (localStorage) található perzisztált to-do elemekről!

Illessze be a releváns kódrészleteket!

The screenshot shows a web application titled "To-Do | Mobil- és webes szoftverek". It has a filter bar with "All 3", "Active 0", "Inactive", and "Done". Below the filter, there is a list of to-do items: "Kutya", "Macska", and "Alma". Each item has a set of controls: up/down arrows, a green checkmark, a minus sign, and a red trash icon. At the bottom, there is a form to "Add new to-do" with a text input and a submit button.

On the right side of the screenshot, the browser's developer tools are open, showing the "Application" tab. Under "Storage", the "Local Storage" section is expanded, showing a single item with the key "todosave" and the value [{"name":"Kutya","state":"active"}, {"name":"Macska","state":"active"}, {"name":"Alma","state":"active"}].

```
function saveToLocalStorage(object){
  const value = JSON.stringify(object);
  localStorage.setItem('todosave' , value);
}

function loadFromLocalStorage(){
  const todosSave = localStorage.getItem('todosave');
  return todos =JSON.parse(todosSave);
}
```

Mentés localStorage-ba:

```
function saveToLocalStorage(object){
  const value = JSON.stringify(object);
  localStorage.setItem('todosave' , value);
}
```

Betöltés localStorage-ból:

```
function loadFromLocalStorage(){
  const todosSave = localStorage.getItem('todosave');
  return todos =JSON.parse(todosSave);
}
```