

Completare l'applicazione

Per completare l'applicazione mancano due passaggi: implementare la funzione `toggle` che permette di impostare un elemento come fatto o da fare; una funzione `delete` per eliminare tutti i `todo` completati.

Toggle

Per impostare un elemento a fatto o non fatto bisogna cambiare il valore del campo `done`, ma per farlo bisogna prima individuare l'elemento `todo` su cui vogliamo agire.

Gli elementi `todo` sono identificati tramite il campo `id`, quindi il problema si riduce a comunicare al server il valore del giusto `id`. Nella pagina HTML (template EJS per la precisione), troviamo il ciclo di stampa dei vari `todo`. In quella fase possiamo creare un form personalizzato per ognuno di essi:

```
<ul>
  <% todos.forEach( function (todo) { %>
    <li>
      <%= todo.text %> - <%= todo.done %>

      <!-- Creiamo qui un form personalizzato per ogni todo -->
      <form action="/todos/toggle" method="POST">
        <!-- non deve essere l'utente a selezionare l'id,
              quindi utilizziamo un campo nascosto e lo
              precompiliamo con todo.id -->
        <input type="hidden" name="id" value="<%= todo.id %>">

        <input type="submit" value="Toggle">
      </form>
    </li>
  <% }); %>
</ul>
```

Il server deve gestire questa richiesta POST. Il fatto che stiamo memorizzando la nostra lista di `todo` in memoria ci mette di fronte ad un piccolo problema: bisogna cercare l'elemento con l'`id` che arriva dal client e modificarlo:

```
app.post('/todos/toggle', function (req, res) {
  // il client dice che bisogna modificare l'elemento con id:
  const id = parseInt(req.body.id);

  // cerca il todo con questo id
  const todo = todos.find( function (todo) {
    return todo.id === id;
  });

  // se lo ha trovato lo modifica
  if (todo) {
```

```
        todo.done = !todo.done;
    }

    // infine mostra la home page
    res.redirect('/');
});
```

Delete

La funzione `delete` deve eliminare tutti i `todo` che hanno il campo `done` pari a `true`.

Nel client basterà creare un solo form e non sarà necessario passare nessuna informazione al server:

```
<!-- Elimina tutti i todo che hanno done=true -->
<form action="/todos/delete" method="POST">
    <input type="submit" value="Delete">
</form>
```

Il server gestisce la richiesta andando a selezionare (filtrare) gli elementi che hanno la proprietà `done` pari a `false` e costruirà una nuova lista di `todos` contenente solo questi elementi (di fatto eliminando quelli che hanno `done` pari a `true`):

```
app.post('/todos/delete', function (req, res) {
    // seleziona solo i todo che hanno done=false
    //
    // Nota: todos viene riscritta, quindi deve essere
    // dichiarata con let e non con const!
    todos = todos.filter( function (todo) {
        return todo.done === false;
    });

    // infine mostra la home page
    res.redirect('/');
});
```