

To do List
Looks Like a piece
of paper

To Do #	Completed #	All #
1	~~~~~	○
2	~~~~~	✗✓
3	~~~~~	○✓

Input Task	set task
	Clear All
	Complete All

PAGE STATE

"active" "completed" "All"

tooblist []

list object { Text: ''
active: Bool
id/key: Num

3

What do I need?

- USE Local Storage
- Display All to-do list items
- Have 3 Tabs that displays To-do, completed to-do, and All
- If no input is there or no tasks have a displayed message to urge user in the direction to go.
- When task is completed check or cross out task
- See a count of remaining items
- Ability to remove one item at a time
 - Possible stretch choose more than one to delete.
- When completed, check or strike out task
- Remove All tasks
- Uncomplete All Tasks

All Page: {

 NavBar

 All Task list

}

NavBar {

To Do

Completed

All

}

To Do: sets state to
all items that are active(True)

Completed: sets state to
all items with active(False)

All: sets state to display
all tasks

All Task List {

ordered list of strings

pulled from TextField input

Completed check display

Delete this task button

(or select this task for deletion)

Input Task Field

Get Task Button

Clear Completed status Button

Complete all button

}

Page Load

- User is prompted to input a task and hit submit button
- When button is clicked
 - Text in input field is added to the first object that has a unique id, an active state, and the entered text.
 - That object is then added to the ListItems array
 - This array is stored in Local Storage.
 - Input field is then clear and ready for the next entry
 - That object is then added to the ToDoList array.
 - The TODO Tab and completed tabs update depending on the contents of the ToDoList array
 - if active = true display todo
 - if active = false display is completed.

List+Item Component

Text - if active = True \rightarrow regular text
if active = False \rightarrow strikeout

Button For changing state of item (active = True or False)

Button for deleting specific item from TODOList array.

}

use one local storage key

and put many items in it

Key: [0; 2]

Count into local storage

use count from local storage to set count in APP

~~~~~ 

<div> ~~~~~ </div>  
<icon> activeIconUrl </I>

<Icon> Delete this task </B>

Three icons

✓ - Complete

✗ - delete this task

○ - Active

Function deleteTask () {

localStorage.removeItem ('props.key');

Unique  
key given  
when task is made.

}

<Icon onClick={deleteTask} > </I>

let activeIconUrl =

if (Props.active === true) {  
 ○ icon  
}

if (Props.active === false) {  
 ✗ icon  
}

Function changeActive () {

Set task to the Id of

Add to local storage ( )  
E

When adding to local storage

I want to update an array of objects in local storage with a key of tasks.

Also need to increment the count key by one every time we add a new task.

The new tasks unique id should be the current count number in local storage.

What happens on page load.

- Sets taskNumber to
  - JSON Parsed
    - Ternary statement that says if localStorage has an item with the key of count use that value.  
else set to 0
- Set value of inputValue as an empty string with useState.
- Set count as the value of taskNumber - Should be 0 -
- User inputs task in box.
  - After user clicks away from box it will update the inputValue with the entered text.
- User clicks set task
  - App runs addTask
    - Sets a boilerplate for one task object called Temp
      - It has Text, id, & active
      - Text = inputValue, {id = count}, active = true  
{  
should be 0}
- App runs getTask array which adds Temp to the taskArray



- App then runs add to local storage with the parameters of count and props.Task array.

This function sets item in local storage with the key of 'task' and value of the stringify version of task array

setItem of key: 'count' and value of count

- Then calls setCount(taskNumber + 1)

I need the task array to be updated before the update state is ran for local storage.

On first render I want temp to equal the boilerplate for each task.

I can generate an array of tasks, each with a unique id, in local storage.

User can input any text and add it as a task.

- I need to be able to delete specific items from the array.
- I need to be able to activate/deactivate specific items - and change the icon based on this.
- Change the active key to either True or False based on complete all or activate all buttons.
- Change active "page" based on if the item key active is true or false.
- Remove all completed items on button click.

Function deleteOneTask(task)

localStorage.removeItem(task.id)

How do I point to the  
exact id in local storage?

# JSON API RETURN

Task {

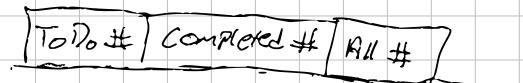
  Text: "What-Task?"

  id: Number

  active: Boolean

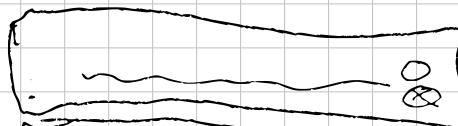
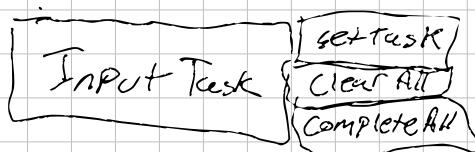
  3

  1  
  2  
  3

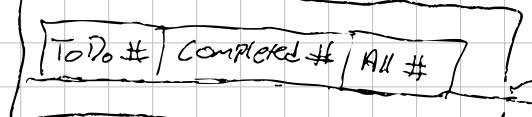


- 1 ~~~~~ Ø
- 2 ~~~~~ Ø
- 3 ~~~~~ Ø

- Task I



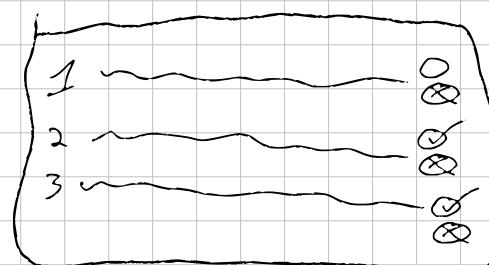
- Task Item



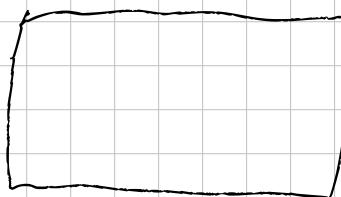
- Nav Bar



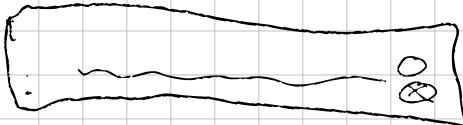
- Input Area



- Task Field



- App Body  
Contains All



— Task Item

~~~~~ — user inputted text

○ — Icon to display if task is active or not. Changes to ○ when not active.

✖ — Delete this task Button



— Nav Bar

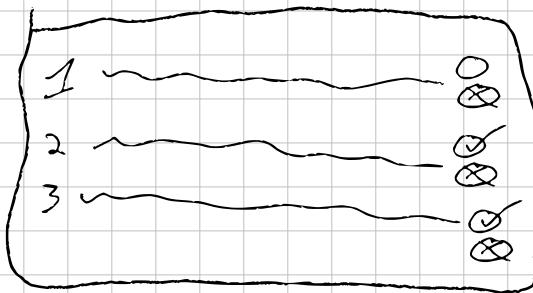
[ToDo #] — Tag that changes state to a list of tasks that have active: true

— This displays number of tasks in this category.

[Completed #] — Changes state to active: false
— Same as last

[All #] — Changes state to show both active: true & false.

— Same as last



- Task Field

An ordered list of TaskItem



- Input Area

[Input Task] ← Input Field for users Task

[setTask] - Button that sets the task into an array.

[clearAll] - This Button clears both the array and the local storage.

[completeAll] - Changes active: to true on all Items in the array.

- App Body /
- NAV Bar /
- Task Field
- Task Item
- Input Field
 - Input Text
 - Set task
 - Clear all
- Complete All