Certainly! Here are three functions that utilize the properties `isClicked` and `task` to perform read, write, and update operations on local storage:

Function 1: `writeToLocalStorage`

```javascript
function writeToLocalStorage(isClicked, task) {
  const data = {
    isClicked: isClicked,
    task: task
  };

  localStorage.setItem('myData', JSON.stringify(data));
}
```

This function takes the `isClicked` boolean and `task` string as parameters, creates an object `data` with those properties, and then stores it in the local storage with the key `'myData'`.

Function 2: `readFromLocalStorage`

```javascript
function readFromLocalStorage() {
  const data = localStorage.getItem('myData');

  if (data) {
    return JSON.parse(data);
  } else {
    return null;
  }
}
```

This function retrieves the data stored in the local storage with the key `'myData'`. If the data exists, it is parsed from JSON format and returned as an object. If the data does not exist, it returns `null`.

Function 3: `updateLocalStorage`

```javascript
function updateLocalStorage(isClicked, task) {
  const data = readFromLocalStorage();

  if (data) {
    data.isClicked = isClicked;
    data.task = task;
    writeToLocalStorage(data.isClicked, data.task);
  }
}
```

This function first reads the existing data from the local storage using the `readFromLocalStorage` function. If the data exists, it updates the `isClicked` and `task` properties with the new values and then calls the `writeToLocalStorage` function to store the updated data.

These three functions can be used together to write, read, and update local storage based on the values of `isClicked` and `task`.