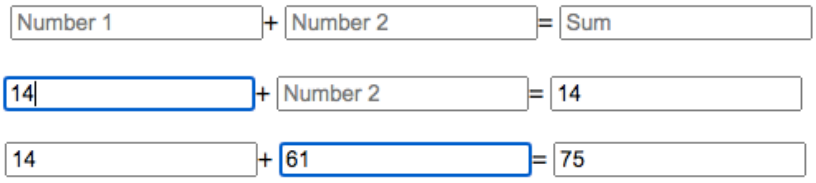
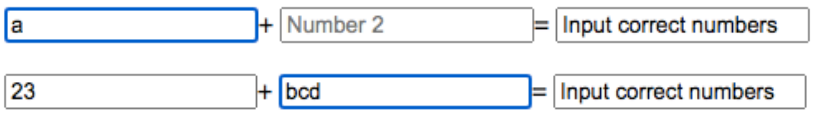
# JS-A3#3

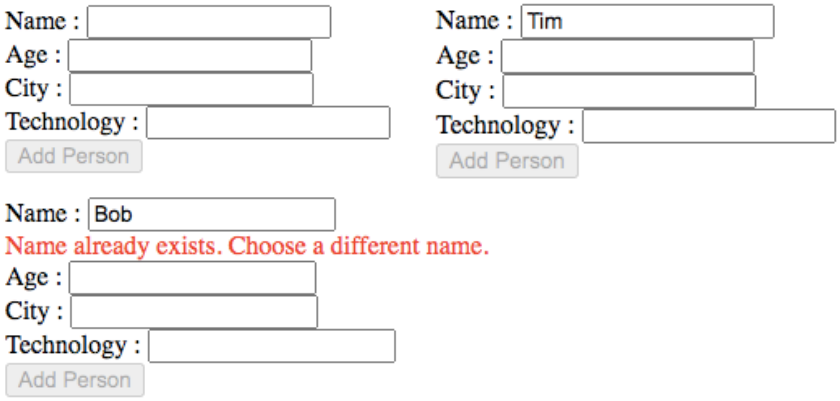
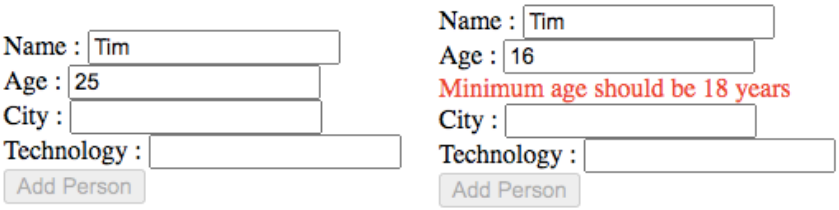
| | |
|---|---|
| Task 1 | Watch the lesson video carefully twice (at least). Then answer the following or write the code as discussed in the video for<br>a. Show the text below as it is being typed. Style it based on the length<br>b. Show a text field to enter the name. When the focus is lost, check whether the name is at least 6 chars. If not, show an error message and disable Submit button.<br>c. List of words starting with the text<br>d. Choosing the payment option and then the provider<br>e. Explain the concept of events<br>f. How can we handle events<br>g. Name the common event types with an explanation of each event |

| | |
|---|---|
| Task 2.1 | Three text fields are shown. The first two take input and in the third, the sum is shown. As the numbers are entered, show the correct sum. If any text field is empty, take its value to be 0.<br>  |
| Task 2.2 | If in any of the text fields, check for non numerical values. Google, learn and use the isNaN() function. If any of the text fields has non numerical input, the sum text field show display a message, Input correct numbers.<br>  |

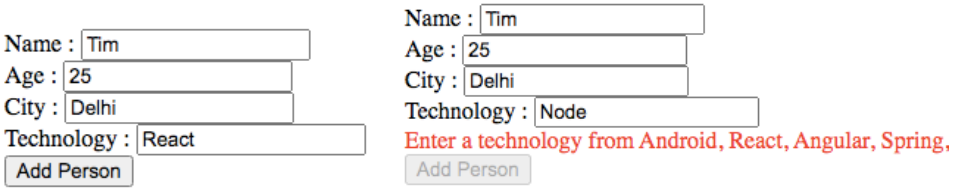
| | |
|---|---|
| Task 3.1 | In a text field, ensure that only 0 or 1 are entered. If any other character is entered, remove it. In the keyup event, find the last character of the value of the text field. If it is not 0 or 1, remove the last character and update the value of the text field.<br>  |
| Task 3.2 | In a text field, ensure that no b or B are entered.<br>  |
| Task 3.3 | In a text field, ensure that no special characters are entered.<br>  |

# JS-A3#3

| | |
|---|---|
| Task 3.4 | In a text field, ensure that only digits and + - * / and . are entered.<br>  |

| | |
|---|---|
| Task 4.1 | Given an array of JSON,<br>[<br>{"name":"Jack","age":28,"city":"London","tech":"Android"},<br>{"name":"Mary","age":32,"city":"Paris","tech":"React"},<br>{"name":"Bob","age":35,"city":"New York","tech":"Angular"},<br>{"name":"Steve","age":26,"city":"Delhi","tech":"Spring"}<br>]<br>If the name already exists, show the error message when the value in the name text field is changed.<br>  |
| Task 4.2 | For age, show an error message is age is less than 18.<br>  |
| Task 4.3 | For city, show an error message if city is not one of the cities in the array. In the error message show the names of cities from which to enter.<br>  |

# JS-A3#3

| | |
|---|---|
| Task 4.4 | For technology, show an error message if city is not one of the tech in the array. In the error message show the names of technologies from which to enter.<br>  |
| Task 4.5 | If all the 4 fields are correctly entered, enable the Add Person button.<br>To do it, keep an array of booleans corresponding to the 4 fields. Initially all the values in the array are false, as none of the fields are correctly entered. Whenever a field is changed, update the boolean according. Also update the status of the button to enabled if all the booleans are true and disabled otherwise. |

| | |
|---|---|
| Task 5 | Given an array of JSON<br>[<br>{"name":"Jack","age":28,"city":"London","tech":"Node JS"},<br>{"name":"Mary","age":32,"city":"Mumbai","tech":"React JS"},<br>{"name":"Bob","age":35,"city":"New York","tech":"Angular JS"},<br>{"name":"Steve","age":26,"city":"New Delhi","tech":"Javascript JS"},<br>{"name":"Edwards","age":27,"city":"Los Angeles","tech":"Node JS"},<br>{"name":"Catherine","age":31,"city":"Montreal","tech":"React JS"},<br>{"name":"Williams","age":34,"city":"New York","tech":"Android"},<br>{"name":"John","age":25,"city":"Moscow","tech":"Java"},<br>{"name":"Anna","age":33,"city":"Moscow","tech":"Node JS"},<br>{"name":"Julia","age":29,"city":"New Jersey","tech":"Android"}<br>]<br><br>Show them in a table, with a dropdown and a text field. The dropdown has the options Name starts with, City starts with and Tech starts with.<br><br>Select the Appropriate Filter  Enter the filter string<br><br>**Name** \| **Age** \| **City** \| **Technology**<br>Jack \| 28 \| London \| Node JS<br>Mary \| 32 \| Mumbai \| React JS<br>Bob \| 35 \| New York \| Angular JS<br>Steve \| 26 \| New Delhi \| Javascript JS<br>Edwards \| 27 \| Los Angeles \| Node JS<br>Catherine \| 31 \| Montreal \| React JS<br>Williams \| 34 \| New York \| Android<br>John \| 25 \| Moscow \| Java<br>Anna \| 33 \| Moscow \| Node JS<br>Julia \| 29 \| New Jersey \| Android |

# JS-A3#3

**Task 5.1**

Depending on the option chosen in the dropdown, the table is filtered as text is typed in the text field. Also, if the option chosen in the dropdown is changed, the table is filtered appropriately. Note that if no option is chosen in the dropdown, the table is not filtered.

Name starts with J

| Name | Age | City | Technology |
|---|---|---|---|
| Jack | 28 | London | Node JS |
| John | 25 | Moscow | Java |
| Julia | 29 | New Jersey | Android |

City starts with M

| Name | Age | City | Technology |
|---|---|---|---|
| Mary | 32 | Mumbai | React JS |
| Catherine | 31 | Montreal | React JS |
| John | 25 | Moscow | Java |
| Anna | 33 | Moscow | Node JS |

City starts with Mo

| Name | Age | City | Technology |
|---|---|---|---|
| Catherine | 31 | Montreal | React JS |
| John | 25 | Moscow | Java |
| Anna | 33 | Moscow | Node JS |

Tech starts with And

| Name | Age | City | Technology |
|---|---|---|---|
| Williams | 34 | New York | Android |
| Julia | 29 | New Jersey | Android |

**Task 5.2**

To the dropdown add the option Name, City or Tech starts with.

Name, City or Tech starts with N

| Name | Age | City | Technology |
|---|---|---|---|
| Jack | 28 | London | Node JS |
| Bob | 35 | New York | Angular JS |
| Steve | 26 | New Delhi | Javascript JS |
| Edwards | 27 | Los Angeles | Node JS |
| Williams | 34 | New York | Android |
| Anna | 33 | Moscow | Node JS |
| Julia | 29 | New Jersey | Android |

Name, City or Tech starts with Ja

| Name | Age | City | Technology |
|---|---|---|---|
| Jack | 28 | London | Node JS |
| Steve | 26 | New Delhi | Javascript JS |
| John | 25 | Moscow | Java |

**Task 5.3**

In the dropdown, add the option, Present in Name, City or Tech i.e. the text can be present anywhere in the name, city or tech.

Present in Name, City or Tech JS

| Name | Age | City | Technology |
|---|---|---|---|
| Jack | 28 | London | Node JS |
| Mary | 32 | Mumbai | React JS |
| Bob | 35 | New York | Angular JS |
| Steve | 26 | New Delhi | Javascript JS |
| Edwards | 27 | Los Angeles | Node JS |
| Catherine | 31 | Montreal | React JS |
| Anna | 33 | Moscow | Node JS |

Present in Name, City or Tech ar

| Name | Age | City | Technology |
|---|---|---|---|
| Mary | 32 | Mumbai | React JS |
| Bob | 35 | New York | Angular JS |
| Edwards | 27 | Los Angeles | Node JS |

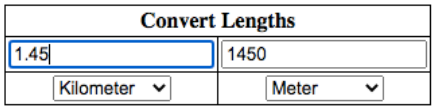
**Task 6**

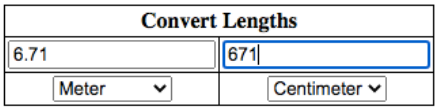
A length converter is shown below. It has 2 text fields and 2 dropdowns. The options in the dropdown are Kilometer, Meter, Centimeter and Millimeter. Initially, it looks like this.

| Convert Lengths | |
|---|---|
| 1 | 1 |
| Kilometer | Kilometer |

As the value in one textfield is changed, the value in the other textfield is updated suitably to reflect the correct conversion.

# JS-A3#3




If the option in the left dropdown or the right dropdown is changed, then the value in the right textfield is changed to reflect the correct conversion.