

# Two Simple Client Scripts

## Lab 2.1

15-20 minutes

### Lab Objectives

You will perform the following in this lab:

- Write and test two simple Client Scripts. Consider saving the Client Scripts module as a Favorite before beginning this lab.
- Create an *onLoad()* Client Script that generates an alert.
- Create an *onCellEdit()* Client Script to stop Incident records from being set to Resolved or Closed from a list view.

### A. Create an *onLoad()* Client Script

1. Create a new Client Script.

Name: **Lab 2.1 onLoad Alert**

Table: **Incident [incident]**

UI Type: **Desktop**


Type: **onLoad**

Active: **Selected (checked)**

Inherited: **Not selected (not checked)**

Global: **Selected (checked)**

Description: **Lab 2.1 onLoad Client Script**

Name	<input type="text" value="Lab 2.1 onLoad Alert"/>	Application	<input type="text" value="Global"/>	
Table	<input type="text" value="Incident [incident]"/>	Active	<input checked="" type="checkbox"/>	
UI Type	<input type="text" value="Desktop"/>	Inherited	<input type="checkbox"/>	
Type	<input type="text" value="onLoad"/>	Global	<input checked="" type="checkbox"/>	
Description	<input type="text" value="Lab 2.1 onLoad Client Script"/>			

2. Note the `onLoad()` function template populates the script field.

Replace the existing comment and add the following alert to the function.

```
1 function onLoad() {  
2     //Type appropriate comment here, and begin script below  
3     alert("The form has finished loading and is ready for user input");  
4 }
```

3. Select **Format Code** to re-align the script. This is a good habit to get into.
4. Select **Submit**.



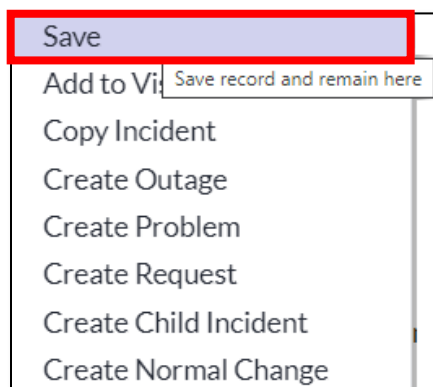
TIP FROM  
THE FIELD:

The keyboard shortcut to **Format Code** starts with **Ctrl+A** for Windows and **Cmd+A** for Mac to select all code. Then **Shift+Tab** to format the code.

## B. Test Your Work

1. Open any Incident record. What happens when the form loads?  

---
2. Is this the behavior you expected? If not, debug and re-test.
3. Select **OK**.
4. Make a change to any field value.
5. Select **Save** on the form's Context menu to save the record and remain on the form.



6. Does the **Lab 2.1 onLoad Alert** Client Script execute again? Why or why not? Explain your reasoning:  

---
7. Create a new Incident. What happens when the form loads for the new incident? Is this the behavior you expected?  

---
8. Make the Client Script **inactive**.
  - a) Open the **Lab 2.1 onLoad Alert** Client Script.
  - b) Uncheck the **Active** field.
  - c) Select **Update**.

## C. Create an onCellEdit() Client Script

1. Create a new Client Script.

Name: **Lab 2.1 onCellEdit Alert**  
Table: **Incident [incident]**  
UI Type: **Desktop**  
Type: **onCellEdit**  
Field Name: **State**  
Active: **Selected (checked)**  
Inherited: **Not selected (not checked)**  
Global: **Selected (checked)**  
Description: **Lab 2.1 onCellEdit Alert**
2. Examine the pseudo-code for the script you will write:
  - Create the saveAndClose variable with the value of true.
  - If the newValue of State is Resolved, alert the user that they cannot change State to Resolved from a list and set the value of the saveAndClose variable to false.
  - If the newValue of State is Closed, alert the user that they cannot change State to Closed from a list and set the value of the saveAndClose variable to false.
  - Execute the callback function returning the value of saveAndClose.
    - If true is returned, the new value of the State field is committed.
    - If false is returned, the new value of the State field is not committed.

3. Write the following script:

```
1  function onCellEdit(sysIDs, table, oldValues, newValue, callback) {  
2      var saveAndClose = true;  
3  
4      if (newValue == 6) { //Resolved  
5          alert("you cannot change the state to 'Resolved' from a list");  
6          saveAndClose = false;  
7      }  
8  
9      if (newValue == 7) { //Closed  
10         alert("you cannot change the state to 'Closed' from a list");  
11         saveAndClose = false;  
12     }  
13  
14     callback(saveAndClose);  
15 }
```

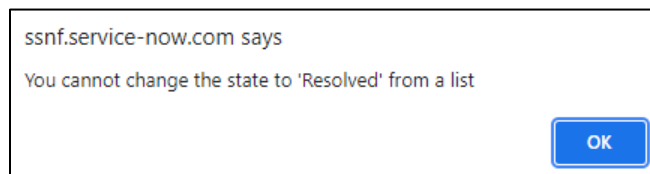
4. Select **Submit**.

## D. Test Your Work

1. Navigate to **Incident > Open**.
2. Without opening a record, pick any Incident in the list. Record the current value of its **State** field:

---

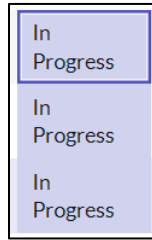
3. Update the value of *State* to **Resolved** from the list view.
  - a) Double-click the **State** field.
  - b) Select **Resolved** from the drop-down list.
  - c) Select the **checkmark** to save the change.
4. Did an alert advising you cannot change the State to 'Resolved' from a list appear? If not, debug and re-test your script.



5. Select **OK**.
6. Did the **Resolved** value remain in the *State* field? Why or why not?

---

7. Hold the **Shift** key down on your keyboard and select **multiple State** fields. The background of the selected items changes to purple.



8. Double-click any one of the selected **State** fields, change its value to **Closed** and select **Save**.
9. Did an alert advising you cannot change the State to 'Closed' from a list appear? If not, debug and re-test your script.
10. Select **OK**.
11. Did the **Closed** value remain in any of the State fields? Why or why not?

- 
12. Select **OK**.

13. Make the *Lab 2.1 onCellEdit Alert* Client Script **inactive**.

## Lab Completion

Good job! You have successfully created `onLoad()` and `onCellEdit()` Client Scripts. Your testing efforts helped you to see how each method works.