▶ **QUESTION**: Which of these classes are part of the ServiceNow client-side API? More than one response may be correct.

- *GlideSystem (gs)*
- *GlideUser (g_user)*
- *GlideDateTime*
- *GlideDate*
- *GlideForm (g_form)*

▶ **QUESTION**: When do onSubmit Client Scripts execute their script logic? More than one response may be correct.

1. When a user clicks the **Submit** button.
2. When a user clicks the **Delete** button.
3. When a user clicks the **Update** button.
4. When a user clicks the **Save** menu item in the *Additional Actions* menu.
5. When a user clicks the **Lookup** button on a reference field.

▼ **QUESTION**: True or False? A single Client Script can execute its script logic when a user loads a record into a form AND when a user saves/submits/updates a form.

ANSWER: **False**. A single Client Script can be either onLoad OR onSubmit but cannot be both. The *Type* field determines whether a Client Script is an onLoad or an onSubmit Client Script.

▼ **QUESTION**: True or False? UI Policies require scripting to make form fields *Mandatory*, *Visible*, or *Read only*.

ANSWER: **False**. Use UI Policy Actions to set field attributes without scripting.

▼ **QUESTION**: Which of the following is a strategy for debugging Client Scripts or UI Policies? More than one response may be correct.

1. Browser's Developer Console
2. Debug UI Policies module
3. JavaScript try/catch
4. JavaScript Log and *jslog()*
5. Field Watcher

ANSWER: **All** of the responses are debugging strategies for client-side script logic.

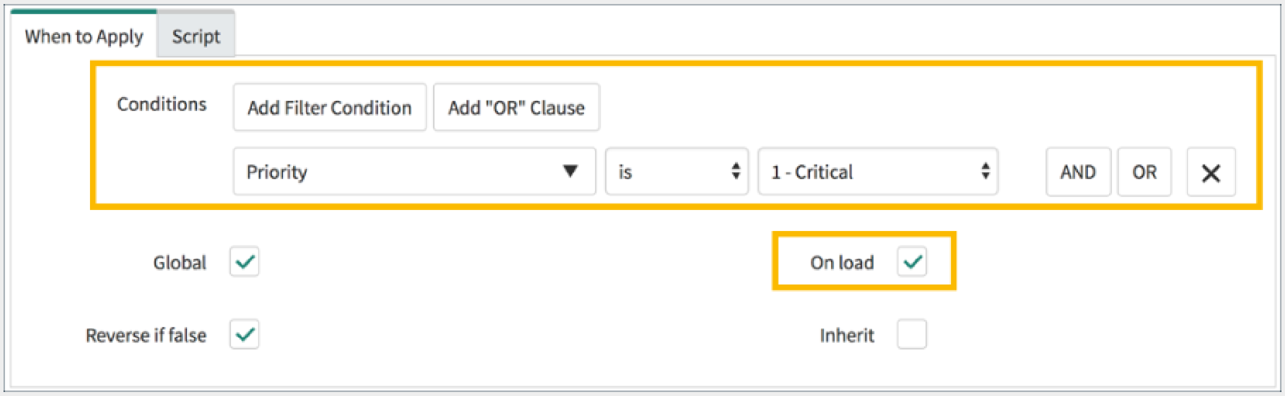
▼ **QUESTION**: Do UI Policies execute UI Policy scripts only when the *Condition* field evaluates to *true*? Explain your reasoning.

ANSWER: If the *Reverse if false* option is selected, UI Policies execute the *Execute if false* script when the *Condition* field evaluates to false.

▼ **QUESTION**: When can UI Policies execute their logic? More than one response may be correct.

1. When a record is loaded into a form.
2. When field values change on a form.
3. When a form is saved, submitted, or updated.

ANSWER: **1** and **2**. The *On load* option determines whether a UI Policy executes its logic when a record is loaded into a form. UI Policy logic also executes whenever the *Conditions* field evaluates to true.



Conditions: Priority ▼ is 1 - Critical

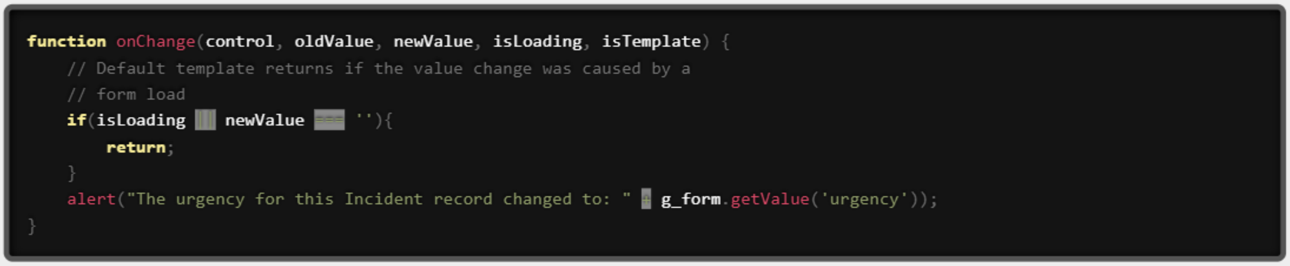
Global ☑

On load ☑

Reverse if false ☑

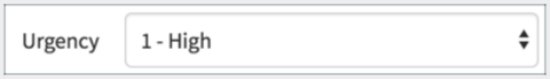
Inherit □

▼ **QUESTION**: Examine the onChange Client Script script for the *Urgency* field:



```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {
    // Default template returns if the value change was caused by a
    // form load
    if(isLoading || newValue == ''){
        return;
    }
    alert("The urgency for this Incident record changed to: " + g_form.getValue('urgency'));
}
```

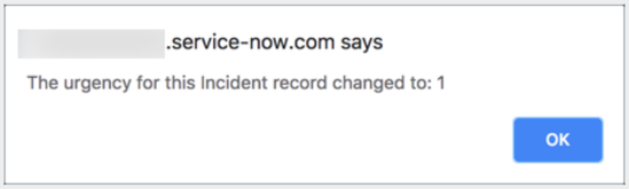
If a user changes the value in the *Urgency* field to **1 - High**, what text will appear in the alert?



Urgency: 1 - High

1. The urgency for this Incident record changed to: 1 - High
2. The urgency for this Incident record changed to: High
3. The urgency for this Incident record changed to: 1

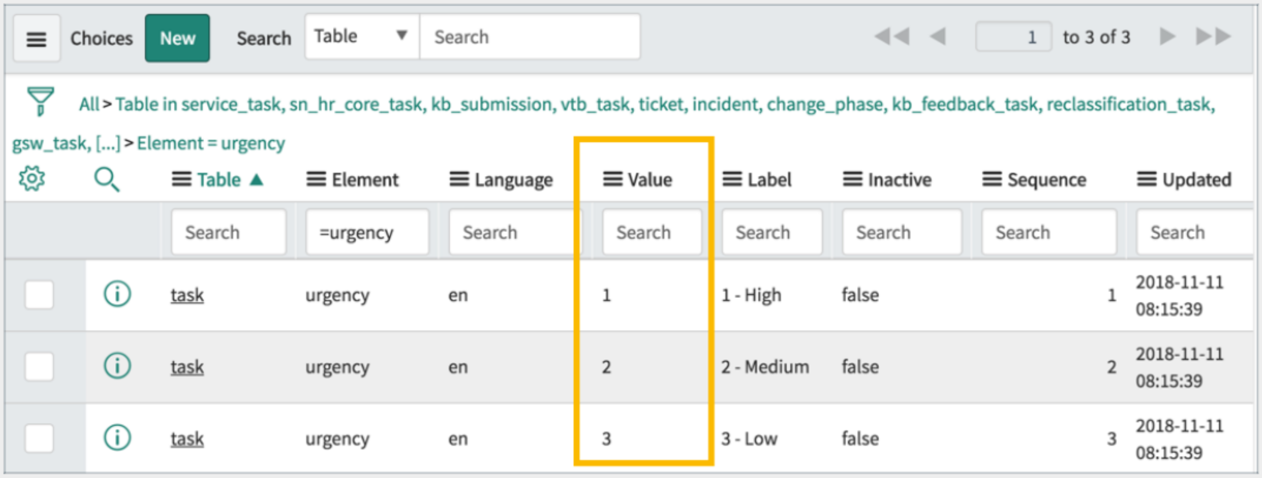
ANSWER: The correct response is **3**.



[redacted].service-now.com says

The urgency for this Incident record changed to: 1

Scripts use choice field *Values* and not field *Labels*. Form fields use *Labels* for choice fields.

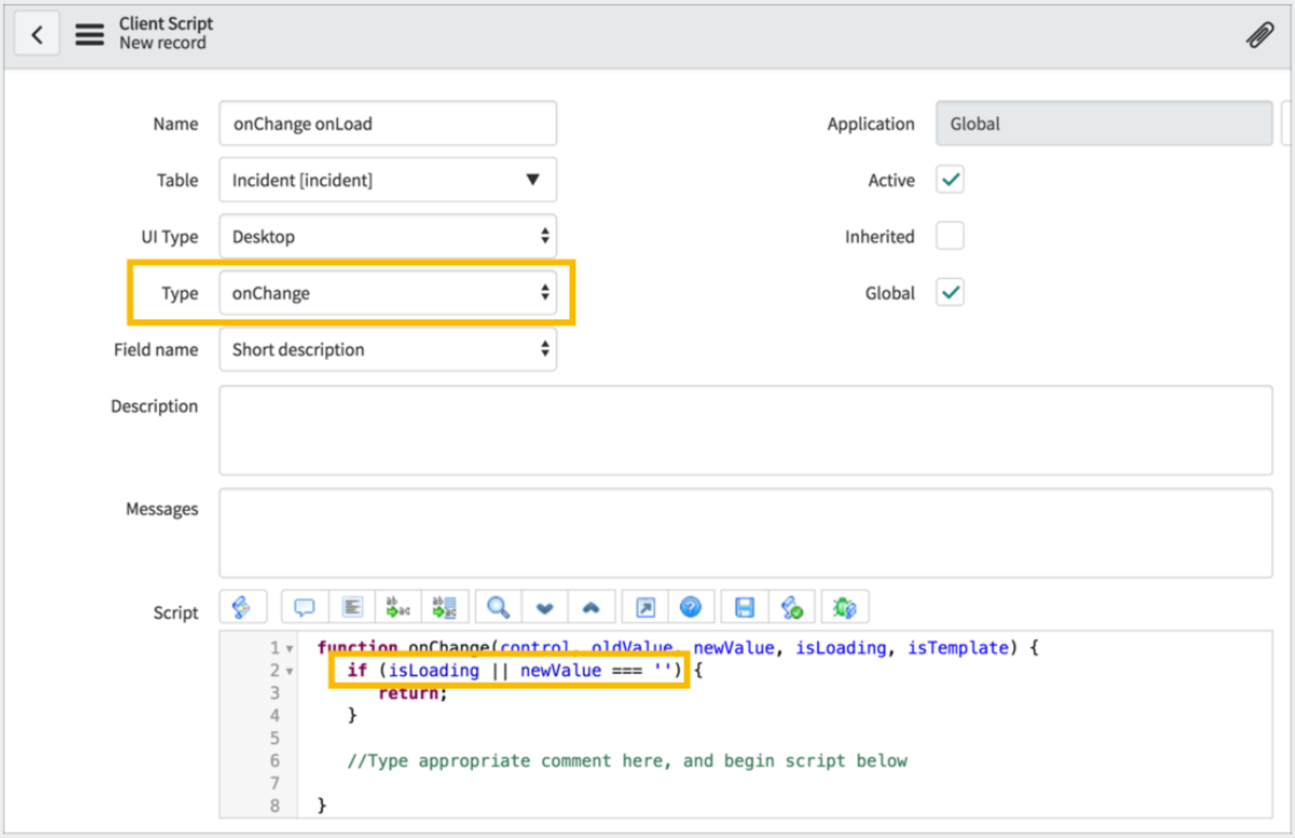


All > Table in service_task, sn_hr_core_task, kb_submission, vtb_task, ticket, incident, change_phase, kb_feedback_task, reclassification_task, gsw_task, [...] > Element = urgency

		Table ▲	Element	Language	Value	Label	Inactive	Sequence	Updated
		Search	=urgency	Search	Search	Search	Search	Search	Search
□	ⓘ	task	urgency	en	1	1 - High	false	1	2018-11-11 08:15:39
□	ⓘ	task	urgency	en	2	2 - Medium	false	2	2018-11-11 08:15:39
□	ⓘ	task	urgency	en	3	3 - Low	false	3	2018-11-11 08:15:39

▼ **QUESTION**: True or False? A single Client Script can execute its script logic when a user loads a record into a form AND when a user changes a value in a field.

ANSWER: **True**. This question is a little tricky. You must examine the Client Script trigger configuration as well as the onChange Client Script script template. When the Client Script *Type* field value is **onChange**, the script template includes logic to check whether the field value changed due to a form load. When a form loads a record from the database, all field values on the form change from no value to the values in the record being loaded from the database. The default client script logic returns (does not execute the remaining script logic) if the *isLoading* value is **true**. The *isLoading* value is **true** if the field value changed due to a form load.



Name	onChange onLoad	Application	Global
Table	Incident [incident]	Active	☑
UI Type	Desktop	Inherited	□
Type	onChange	Global	☑
Field name	Short description		
Description			
Messages			

Script

```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {
   if (isLoading || newValue === '') {
      return;
   }

   //Type appropriate comment here, and begin script below

}
```

To execute an onChange Client Script's logic when a form loads, remove the ***isLoading*** check from the onChange script template.

```
// Default template returns if the value change was caused by a
    // form load
    if(isLoading || newValue === ''){
        return;
    }

    // Remove the isLoading test and the onChange script executes
    // on form load
    if(newValue === ''){
        return;
    }
```