**Business Rule**

**Scenario 1 : If the Problem Priority = ‘Critical’ then create “General” and “Root Cause analysis” Problem task.**

(function executeRule(current, previous /\*null when async\*/) {

    // Add your code here

    var type = ['general','rca'];

    for(var i=0;i<2;i++){

        var gr = new GlideRecord('problem\_task');

        gr.initialize();

        gr.short\_description = 'Task type is '+type[i];

        gr.problem\_task\_type = type[i];

        gr.problem = current.sys\_id;

        gr.insert();

    }

})(current, previous);

**Scenario 2 : Write a functionality to transfer the work notes from Incident to associated Problem.**

(function executeRule(current, previous /\*null when async\*/) {

    // Add your code here

    var gr = new GlideRecord('problem');

    gr.addQuery('sys\_id',current.problem\_id);

    gr.query();

    if(gr.next()){

        gr.work\_notes = current.work\_notes.getJournalEntry(1);

        gr.update();

    }

})(current, previous);

**Scenario 3 : Write a functionality to count the Associate Incident.**

**(To do: adjust the count if the Associate Incident gets deleted)**

(function executeRule(current, previous /\*null when async\*/ ) {

    // Add your code here

    var gr = new GlideRecord('problem');

    gr.addQuery('sys\_id', current.problem\_id);

    gr.query();

    if(gr.next()) {

        gr.u\_associate\_integer\_count = gr.u\_associate\_integer\_count + 1;

    }

    gr.update();

})(current, previous);