

**Classroom Session**

**Title: Create , Update, and Delete - Chaining of Requests**

**Learning Objective**

The objective of this session is to learn how to create, update, and delete a change request in ServiceNow by chaining these requests using separate classes for each operation with Rest Assured in Eclipse. The sys\_id of the change request will be stored in the Base class as a public static variable.

**Expected Completion Time**

* **Best Case:** 30 minutes
* **Average Case:** 40 minutes

**Session Details**

**ServiceNow API Credentials:**

1. Go to [ServiceNow Developer](https://developer.servicenow.com/dev.do#!/home).
2. Login with your email ID and password.
3. Click on your profile icon and click "Manage Instance Password" to obtain your ServiceNow credentials.

**Example Credentials:**

* URL: http://dev12345.service-now.com
* Username: Admin
* Password: yourPassword

**Structure of the Project**

Create 1 package and classes for:

1. Create a Package **– chaining**
2. **Base Class**
3. **CreateIncident**
4. **UpdateIncident**
5. **DeleteIncident**

**Pseudo Code Outline**

**1. Base Class**

**Purpose:** Set up the base URL and common configurations. Store the sys\_id.

**Pseudo Code:**

Class BaseTest {

// Define Sys\_Id as global -

**public** **static** String *sys\_Id* ;

// Define Base URI And Authentication inside a method

Public void setUp(){

RestAssured.*baseURI*=" ***<INSTANCE>/api/now/table/incident***";

RestAssured.*authentication*= RestAssured.*basic*(" USERNAME ", " PASSWORD ");

}

}

**2. CreateIncident Class**

**Purpose:** Handle the creation of an incident and store the sys\_id in the Base class.

// Create Incident

Class CreateIncident extends BaseTest {

// Method to create an incident

public void createIncident() {

// Set up request with necessary headers and body

Send POST request to with body containing incident details

Capture response

Extract sys\_id from response

Store sys\_id in sys\_id ;

}

}

**3. UpdateIncident Class**

**Purpose:** Handle updating of the incident using the sys\_id from the Base class.

**Pseudo Code:**

// Update Incident

Class UpdateIncident extends BaseTest {

// Method to update an incident

public void updateIncident() {

// Set up request with necessary headers and updated body

Send PUT request (sys\_id) with updated body

Capture response

}

}

**4. DeleteIncident Class**

**Purpose:** Handle deletion of the incident using the sys\_id from the Base class.

**Pseudo Code:**

// Delete Incident

Class DeleteIncident extends BaseTest {

// Method to delete an incident

public void deleteIncident() {

// Set up request with necessary headers

Send DELETE request with (sys\_id)

Capture response

}

}

**Implementing Chaining of Requests**

* Hint: Add @BeforeMethod in BaseClass and Depends on method for Update and Delete @Test(dependsOnMethod=”chaining.CreateIncident.create”)
* Run From Delete Instance

**Expected Outcome**

Upon completion, you should be able to:

1. Understand how to use Rest Assured to interact with ServiceNow APIs.
2. Create, retrieve, update, and delete a change request in ServiceNow via API.
3. Chain these requests using separate classes for each operation with the sys\_id stored in the Base class.