Week 4 learnings

Scripting
Client side: Make requests , whatever u do on the browser side, Frontend, related to forms, tables, modify incident form

Server side: Return Responses , Backend process , to get data, ACL

Global Scope
Baseline applications
Custom applications built on ServiceNow versions prior to Scoping

Privately Scoped
Custom applications

Scoped apps
- Separate namespace
- Delegated Development
- Source control integration
- Publish to app repository
- Easy file management

Global apps
- Everything in global
- Source Control integration
- Publish to app repository
- Easy file management

Scope and delegated development
- Admins can grant non-admin users the ability to develop, deploy, and delete scoped applications
- Access can be granted to only scoped applications
- Admins use Studio to manage Delegated Developers and the content they can access

What is an update set?
- A container for capturing customizations

What is the ServiceNow application repository?

The ServiceNow application repository:
- Enables companies to store published applications for installation on any instance belonging to that company.
  glide.appcreator.company.code value must be the same on all instances (maint only property).
- Standardizes applications and versions installed on instances.
- Makes it easy to install/uninstall/update apps.
  Admin role required.

Business logic
- What needs to be done
- Process
- Stakeholders
- Input/Output

Database logic
- Database tables and columns
- Relationships

User Interface
- User profiles

- Desktop/tablet/mobile

Scoped Architecture and Design

- Client -side
    - Forms
    - Client-side Scripts
    - UI Actions
- Server-side
    - Script Includes
    - GlideAJAX

User experience(UX) options:
- Mobile
- Classic

ServiceNow APIs
There are six API categories:
- Client: Client-side API for desktop apps
- Client Mobile: ServiceNow Classic mobile application API. Not for ServiceNow Agent, Now Mobile, or ServiceNow Onboarding
- Now Experience UI Framework: Agent Workspace component API
- Server Scoped: Scoped application API for server-side
- Server Global: Global application API for server-side
- REST: Restful APIs for interacting with a ServiceNow instance

ServiceNow Script Editor
All scripts, regardless of the script type, have two parts:
- Configuration: specifies when to execute the script logic
- Script: contains the script logic specifying what to do when the configuration criteria are met

Although the configuration is different for every script type, the Script Editor is the same for all scripts. The Script Editor features include:
- Real-time syntax checking
- Toolbar
- Syntax highlighting
- Scripting assistance
- Links to API documentation

Client Script
Client Scripts manage the behaviour of forms, fields and lists in a real time
- Make fields mandatory
- Set one field in response to another
- Modify choice list options
- Hide/Show form sections
- Hide fields
- Display an alert
- Manage/prohibit list editing

Client Scripts execute client-side (web browser)
    Browsers may present items in different ways
Triggers specifies when to execute
Script specifies what to execute
Client scripts execute when their trigger condition is met
Additional info if the client script is triggered by a field value change

Client script Trigger-Order
- Use the Order field when multiple Client Scripts for the same table have conflicting logic
- Executes in order from lowest to highest

• Does not appear on the Client Script form baseline
    Configure the form to add the field

1)Client Script Type - onLoad()
    • Script runs when a form loads and before control is given to the user
    • Typically used to manipulate a form's appearance or content on the screen

2)Client Script Type - onSubmit()
    • Script runs when a form is saved, updated or submitted
    • Typically used for field validation

3)Client Script Type - onChange()
    • Script runs when a particular field value changes
    • Typically used to
        Respond to field values of interest
        Modify one field value in response to another
4)Client Script type-onCellEdit()
    Script runs when a particular field on a list changes value

Client-side APIs: What Data Can You See in a Client Script?

• Use ServiceNow's predefined client-side classes and methods to
    Control how the platform looks and behaves in a web browser
    Enhance the end user experience
• This class reviews the most popular Client-side APIs
    - g_form (GlideForm)
    - g_user (GlideUser)
    - g_scratchpad (in conjunction with a Display Business Rule)

g_form Object
The Glide Form API provides useful methods to
    - Customize forms
    - Manage form fields and their data

g_form Methods
• Access GlideForm methods using the g_form global object
    g_form.<method_name>(parameter information);
• Examples
    g_form.setValue('impact', 1);
    g_form.showFieldMsg('state', 'Change is waiting approval','info');
• Commonly used g_form method examples
    Draw attention to an area on the form: flash(), showFieldMsg()
    Get field information: getValue()
    Change a field value: setValue(), clearValue()
    Change a choice list: addOption(), clearOptions()
    Get form information: getSections(), isNewRecord()
    Add/clear form messages: addFormMessage(), clear AllFormMessages()

Referencing Field Names
• g_form object methods refer to fields by their field name, not their label
• Easy way to locate a field name
    - Right-click a field's label
    - The field name appears on the Context menu

Client-side APIs: g_user Object

• The GlideUser API provides useful methods to access information about the currently logged in user
• Access GlideUser methods using the g_user global object

g_user object Properties and Methods

g_user Properties:

- firstName
- lastName
- userID
- userName

Syntax: g_user.<property>
Example:

```
alert("Logged in user: " + g_user.userName);
```

g_user Methods:

- getClientData()
- getFullName()
- hasRole()
- hasRole Exactly()
- has Role FromList()
- hasRoles()

Syntax: g_user.<method_name>(parameter information);
Example:

```
if (g_user.hasRole('itil'){
alert("Logged in user is a Fulfiller");
}
```

Client-side Debugging: Debugging Client Scripts - Desktop

- Many times, your scripts do not work as expected
- You can debug them using the following strategies:
    - ServiceNow built-in client-side debugging tools
    - Script debug messages
- Response Time Indicator
    - JavaScript debugging tools
        - alert()
        - try/catch
    - Browser tools (for example, JavaScript console, Web Console)
    - Third-party tools

What is a UI Policy?

- Defines the behavior and visibility of fields on a form
    - Mandatory
    - Visible
    - Read-only
- Condition must be true for it to execute
- Executes after Client Scripts

What is UI Policy Scripting?

- Ability to script complex conditions and execute advanced behavior
    - Show/Hide sections (tabs)
    - Remove/add/change/validate data in fields
- Uses the full power of JavaScript

UI Policy Execution

- Trigger specifies when to execute
- UI Policy Actions and/or Scripts specify what to execute
- Every field in the record can be evaluated even if it is not visible on the form
- UI Policies do not have a Name field

Use the Short description field as a pseudo name

- Not all UI Policies require scripting

UI Policy Trigger - When to Apply?

UI Policies execute based on evaluation of the Conditions

Build conditions with the Condition Builder rather than scripting for better performance

If left blank, the UI Policy logic always executes

What Data Can You See in a UI Policy?

- Local variables declared in a script
- Predefined client-side Global variables, such as:

- g_form (GlideForm)
- g_user (GlideUser)
- g_scratchpad (in conjunction with a Display Business Rule)

What is a Catalog Client Script?

Manages the behavior of Catalog Items when presented to users

What is a Catalog UI Policy

- Manages the behavior of Catalog Items when presented to your users
- Offers the use of the Condition builder to easily configure a condition vs. scripting it
- Condition must be true for the policy to execute

Variable Sets

- Catalog Client Scripts and Catalog UI Policies can also be applied to Variable Sets
- Executes every time the Variable Set is used

What Data Can You See in Catalog Client Scripts and Catalog UI Policies?

- Local variables declared in a script
- Predefined client-side Global variables, such as

g_form (GlideForm)

g_user (GlideUser)

What is a Business Rule?

- JavaScript that runs when

A table is queried

A record is displayed, inserted, updated or deleted

- Can be used to change values in fields when certain conditions are met
- Execute server-side

Faster

Does not monitor forms or lists

When Business Rules Run:

- Watches the database for record access (Problem) and responds to select access types (Update)
- Trigger specifies when to run (execute)
- Not all Business Rules require scripting
- Set the criteria for execution of the business Rule logic

User or System Query>Before Query Rules>Database Query>Display Rules>Form Submit>Before>Rules>Database Update>After Rules>Async Rules

Before Query

Execute before a query is sent to the database

Display

- Execute after the data is read from the database and before the form is presented to the user
- Primary purpose is to populate the g_scratchpad global object
- Example: Provide Client Scripts with access to data from other records

Before
Execute after form submission and before record is updated in the database
Example: Populate 'closed_by' with the name of the currently logged in user

After

- Execute after form submission and record update in the database
- Example: Cascade changes made to the approval field of a Service Catalog request to the requested items attached to that request

Async

- Execute after records are inserted/modified/queried
- Run asynchronously as Scheduled Jobs

Example: Notify subscribers when CIs are affected by an Incident
Example: SLA calculations

Server-side Global Variables: What Data Can You See?

- Local variables declared in a script
- Predefined server-side Global variables
  current
  previous
  g_scratchpad
- Display Business Rules only
- Works in conjunction with a client-side script

The GlideSystem API

- Collection of methods
- Execute server-side
- Access system-level information
  Logged in user
  System
  Date and time

User Methods

- Return information about the currently logged in user

Examples
- getUser(), getUserName(), getUserID(), getUserDisplayName()
- hasRole(), has RoleinGroup()

System Methods

- Primarily work with form objects, tables, fields, and logging
- Examples
  getProperty(), getPreference()
  getDisplayColumn(), tableExists()
  nil(), eventQueue()
  print(), log(), logError(), getMessage()

Date and Time Methods

- Work with Date and Time data

- Examples
  beginningOfLastWeek(), endOfLastWeek()
  beginningOfNextMonth(), endOfNextMonth()
  dateDiff()
  now(), nowDateTime()
- minutesAgo(), quartersAgo(), monthsAgo(), yearsAgo()

GlideDate and GlideDateTime Method

- The Glide Date and Glide DateTime APIs are used to manipulate date and time values
- You can create a Glide DateTime object from a GlideDate object by passing in the GlideDate object as a parameter to the GlideDateTime constructor

What is a GlideRecord?

- Used for database operations instead of writing SQL queries
- An object containing zero or more records from the same table (ordered list)
- Contains records (rows) and fields (columns)
- Executes server-side

Create New
Step 1: Create a GlideRecord Object for the Table of Interest
addQuery
Step 2: Build the Query Condition(s)
  addQuery() builds a SQL select statement (not seen by the user)
  Each addQuery() call adds a new "where" clause to the select statement
Query
Step 3: Execute the Query
Process Returned Record
Step 4: Process Returned Records with Script Logic

get(): Use to query for a single Record

GlideQuery:
- Server-side API for querying data
- Written in 100% JavaScript (global Script Include)
- Uses GlideRecord behind the scenes

What does a Script Include?

- Stores reusable JavaScript for execution on the server
- Must be called to run
- Can be client-callable

Three types of Script includes
1. Store One Classless Function
   - Stores one re-usable function
   - Used server-side only
   - Can also be referred to as On Demand
2. Define a New Class
   - Collection of functions
   - Can be called by client-side scripts
   - Standard to include "Utils" in the name (for example, Travel RequestUtils)
3. Extend an Existing Class
   - Adds functionality to an existing Class
   - Can be called from client-side scripts

Flow Designer

Use Flow Designer to develop new flows or when there is a need for a major redesign to an existing Workflow developed in the Workflow Editor.

What is Flow Designer?

Design environment that creates Flows
Flows consist of a trigger and at least one action
Triggers start a flow
Actions automate processes or record operations
Subflows are a sequence of reusable actions

Flow Designer Editor Page
The flow editor uses properties, a trigger, a sequence of actions, and the data collected or created to create a flow.

Flow Designer Trigger
The trigger specifies the conditions that start the flow. When the trigger condition is true, the system starts the flow.

Actions

- A group of reusable operations that automate the Now Platform features without having to write code.
- Process Analysts can use Actions, Flow Logic or Subflows.

Inline Scripts

Enables users with coding experience to write inline scripts that set and modify input values during the configuration of an action or flow.

Complex Data

Provides an easy to understand, configurable and graphical organization of complex structured data
Allows structured data to be encoded and stored in a machine-readable format, such as JSON or XML

What is ServiceNow?

It is a cloud based platform, which was mainly developed for workflow and process automation as per the ITIL principles

Services of ServiceNow:
- IT Service management
- HR management
- GRC
- Integrations
- IT Asset management
- Finance operation management
- IT business management

How to get Free SNOW Instances
Step 1: SignUp from https://developer.servicenow.com/app.do#!/home
Step 2: Fill the Registration form
Step 3: Verify your account
Step 4: Now Login to your ServiceNow Developer Platform.
Step 5: Request/create an instance.
Step 6: Choose the ServiceNow Developer Instance Version
Step 7: Instance Credentials Info
Step 8: Login into your Service Now Developer instance
Instance Activity
- If the instance is inactive for 10 days, then the instance is released

• If your instance is inactive for more than 24 hours, then your instance may go into hibernation state.

Career and Growth in ServiceNow

➤ServiceNow is expected to continue to grow even in future.

➤Currently, Cloud Platform is the Very popular.

➤In the cloud platform, ServiceNow is the best tool to use because of its simplicity and ease of use.

ServiceNow fits best in industries like:

- Governance
- Computer Software
- Insurance
- Health care
- Information technology

Voucher Code for SNOW CSA Exam:

Once u complete the css course you can write the exam and apply the voucher in nowlearning.servicenow.com>Certification>Get certified> System Administrator

You may purchase a voucher when you complete your course

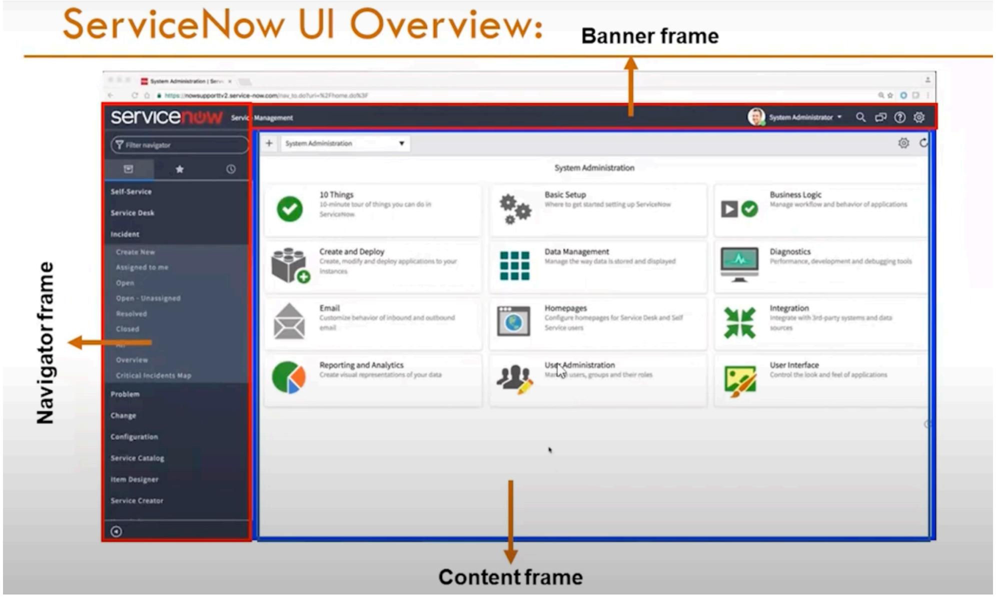
Follow a paid virtual instructor led course, on completion of your course you will receive a voucher

Service Now UI Overview:

• We interact with the application and modules of the ServiceNow platform through the user interface using a web browser.

• The version of the user interface that accompanies the Istanbul version of ServiceNow is called U116.

ServiceNow UI Overview:


Banner frame
Navigator frame
Content frame
servicenow
Service Management
System Administrator
Filter navigator
Self-Service
Service Desk
Incident
Create New
Assigned to me
Open
Open - Unassigned
Resolved
Closed
Overview
Critical Incidents Map
Problem
Change
Configuration
Service Catalog
Item Designer
Service Creator
System Administration
10 Things
10-minute tour of things you can do in ServiceNow
Basic Setup
Where to get started setting up ServiceNow
Business Logic
Manage workflow and behavior of applications
Create and Deploy
Create, modify and deploy applications to your instances
Data Management
Manage the way data is stored and displayed
Diagnostics
Performance, development and debugging tools
Email
Customize behavior of inbound and outbound email
Homepages
Configure homepages for Service Desk and Self Service users
Integration
Integrate with 3rd-party systems and data sources
Reporting and Analytics
Create visual representations of your data
User Administration
Manage users, groups and their roles
User Interface
Control the look and feel of applications

ServiceNow Components:

• The Components are the basic elements of your page.

• Components range from the basic elements like labels, and buttons to more complex experience components like lists and forms.

• These components can be added to your page to create or personalize your workspace D or portal.

Modules

Modules are the elements that make up the ServiceNow application navigator

Some of the modules in ServiceNow are:• ITSM:

- Incident Management
- Problem Management
- Change and Release Management
- Request Management
- Asset and Cost Management
- Walk-Up Experience
- Agent Workspace
- Now Mobile, etc.

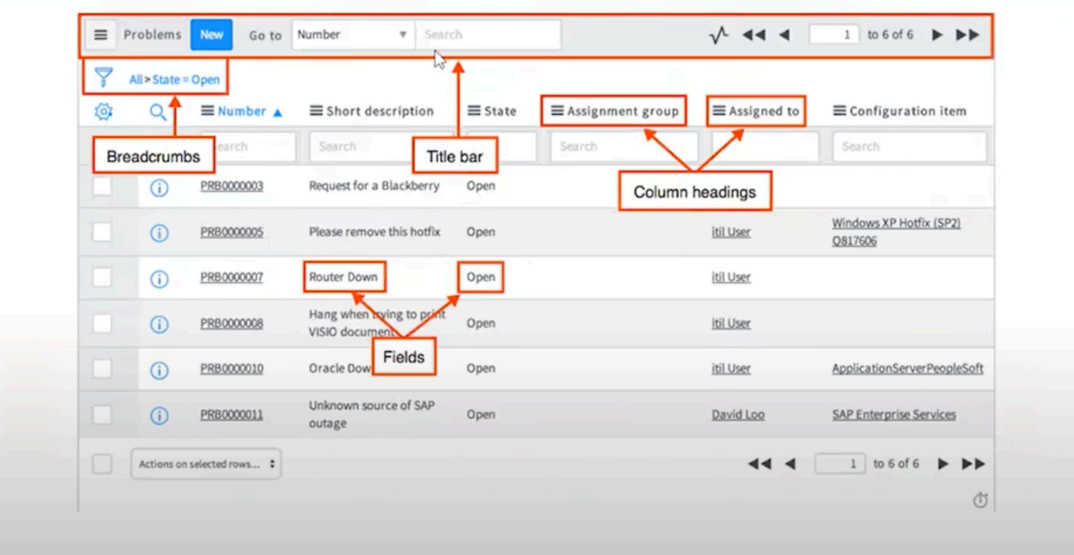
Incident Module:
Interruption to the service
- An incident is a situation where normal service operations are interrupted, disrupted or degraded.
- In ServiceNow, an open incident indicates that the customer is strongly affected or it represents a business risk.
- The process of managing the incident lifecycle is called as an Incident management.

Problem Module
- New
- Assess
- Root Cause Analysis
- Fix in Progress
- Resolved
- Closed

A problem is a cause of one or more incidents.
The process of managing the lifecycle of all the problems that arises or could arise in an IT service is called as Problem management

Change Module
- A change request contains detailed information regarding the change, like the reason for the change, the risk, the priority, the change type, and the change category.
- A systematic approach for controlling the life cycle of all changes, making it easier to make beneficial changes with less disruption to the IT services is called ServiceNow Change Management.

List
- A list displays a set of records from a table.
- Users have the ability to search, sort, filter and edit data in lists.
- Users can search, sort, filter, and edit data in lists. Lists can be integrated into forms and can have sublists.
- The list interface includes a title bar, breadcrumbs and filters, columns of data, and a footer. Every column in a list represents a field in the table.



Number ▲	Short description	State	Assignment group	Assigned to	Configuration item
PRB0000003	Request for a Blackberry	Open			
PRB0000005	Please remove this hotfix	Open		itil User	Windows XP Hotfix (SP2) Q817606
PRB0000007	Router Down	Open		itil User	
PRB0000008	Hang when trying to print VISIO document	Open		itil User	
PRB0000010	Oracle Dow	Open		itil User	ApplicationServerPeopleSoft
PRB0000011	Unknown source of SAP outage	Open		David Loo	SAP Enterprise Services

Forms

- A form is a content page that displays the fields and values of a single record in a database table.
- Forms have a 1-column layout, a 2-column layout, or a blend of both.

Forms are opened from:

modules in the Application navigator
or
by clicking a record number in a list