Cognizant Week 4:

ACL scripting:

In ServiceNow, **Access Control Lists (ACLs)** manage data access by defining who can view, create, update, or delete records. **ACL scripting** customizes these rules beyond basic configurations.

1. **Purpose:** ACLs ensure data security by controlling user access at various levels.
2. **Components:**
    - **Table-level ACLs:** Apply to entire tables.
    - **Field-level ACLs:** Apply to specific fields within a table.
    - **Record-level ACLs:** Apply to individual records.
3. **Types of ACL Rules:**
    - **Read:** Access to view records.
    - **Write:** Access to modify records.
    - **Create:** Access to add new records.
    - **Delete:** Access to remove records.
4. **Condition Scripts:** Evaluate access permissions based on custom conditions.

Client-side Scripting:

Client-side scripting in ServiceNow is used to enhance the interactivity and responsiveness of forms and user interfaces directly within the user's browser. It involves scripting that runs on the client side (in the user's web browser) rather than on the server side.

Types of Client-Side Scripts

1. **Client Scripts:**
    - **Purpose:** Automate and manage form behavior based on user actions. For example, you can control how fields are displayed or validated as users interact with the form.
    - **Events:** Scripts can be triggered by events such as when a form loads, when a field value changes, when a form is submitted, or when a cell in a list is edited.
2. **UI Policies:**

- **Purpose:** Control the appearance and behavior of form fields based on certain conditions. This includes making fields visible or hidden, read-only, or mandatory depending on user input or other criteria.

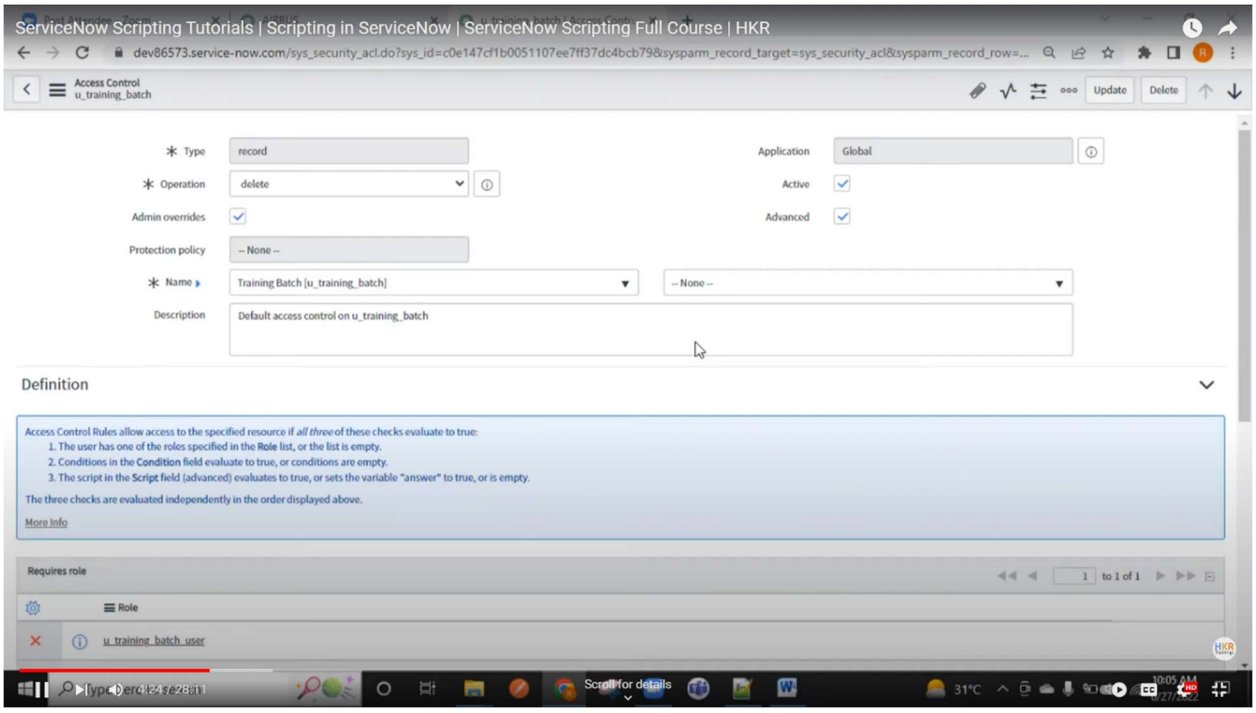
3. **UI Actions:**
    - **Purpose:** Add interactive elements like buttons, links, or context menu options to forms or lists. These actions can perform tasks or trigger other scripts when users interact with them.
4. **Catalog Client Scripts:**
    - **Purpose:** Customize the behavior of Service Catalog items. They can be used to dynamically change the form or field properties based on user selections in a catalog item.

Key Concepts

- **g_form Object:** Provides methods to interact with form fields, such as setting values, making fields read-only, or controlling their visibility.
- **Client-Side Execution:** These scripts run in the user's browser, so they can affect the user experience directly but should be optimized to avoid performance issues.
- **User Experience:** Improves user interaction by providing real-time feedback and adapting the form behavior to user actions.

Best Practices

- **Performance:** Ensure scripts are efficient to prevent slow performance or lag in the user interface.
- **Security:** Avoid including sensitive logic or data in client-side scripts, as these scripts can be accessed and manipulated by end users.
- **Testing:** Test scripts thoroughly across different browsers and devices to ensure consistent behavior and performance.



Customize a Widget:

Customizing a widget in ServiceNow involves modifying its behavior and appearance to meet specific needs. Widgets are components used in the Service Portal to display information or functionality. Here's how you can customize a widget:

1. Accessing the Widget

1. **Navigate to Widget List:**
    - Go to the Service Portal application in ServiceNow.
    - Select **Service Portal > Widgets** to see the list of available widgets.
2. **Find the Widget:**
    - Search for the widget you want to customize. Widgets are listed by name or you can use the filter to locate them.
3. **Open the Widget:**
    - Click on the widget name to open its record for editing.

2. Customizing Widget Behavior

1. **Edit Widget Code:**

- **HTML Template:** Modify the HTML to change the structure and layout of the widget.
- **Client Script:** Adjust the JavaScript code to alter how the widget behaves on the client side. This script handles interactions and dynamic updates.
- **Server Script:** Change the server-side JavaScript to manage data retrieval and processing before sending it to the client.

2. **Update Widget Options:**
    - In the widget record, you can adjust properties like **Title**, **Description**, and **Widget ID**.

3. Customizing Widget Appearance

1. **CSS Customization:**
    - Add or modify CSS styles to change the appearance of the widget. This can be done in the widget's HTML template or by referencing a custom stylesheet.
2. **Service Portal Themes:**
    - If you want to apply global styling, you can adjust the Service Portal theme settings to impact the overall look and feel of the widget across the portal.

4. Testing and Previewing

1. **Preview Widget:**
    - Use the **Preview** feature to see how your changes look and behave in the Service Portal before publishing them.
2. **Test Functionality:**
    - Ensure that all interactions and data manipulations work as expected in different scenarios and across various devices.

5. Publishing Changes

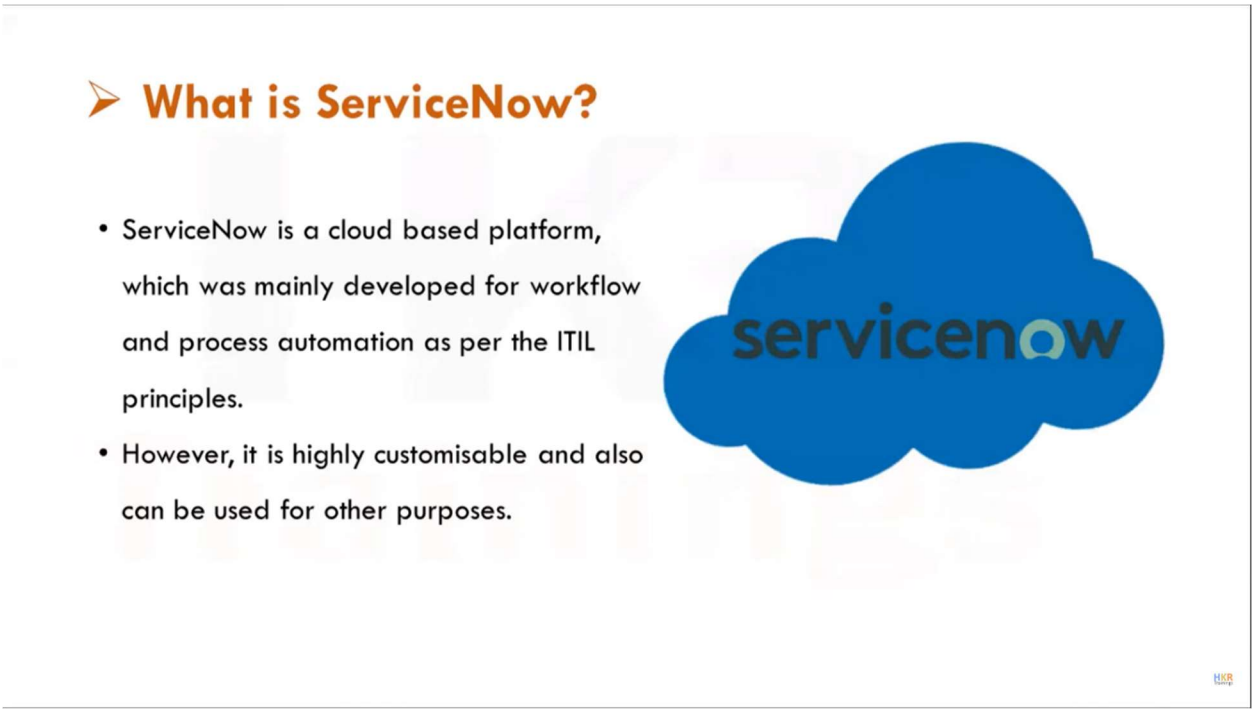
1. **Save and Publish:**
    - After making your changes, save them and publish the widget if required. This ensures that your customizations are reflected in the live portal.
2. **Check for Conflicts:**
    - Verify that your customizations do not conflict with other widgets or portal elements.

6. Documentation and Maintenance

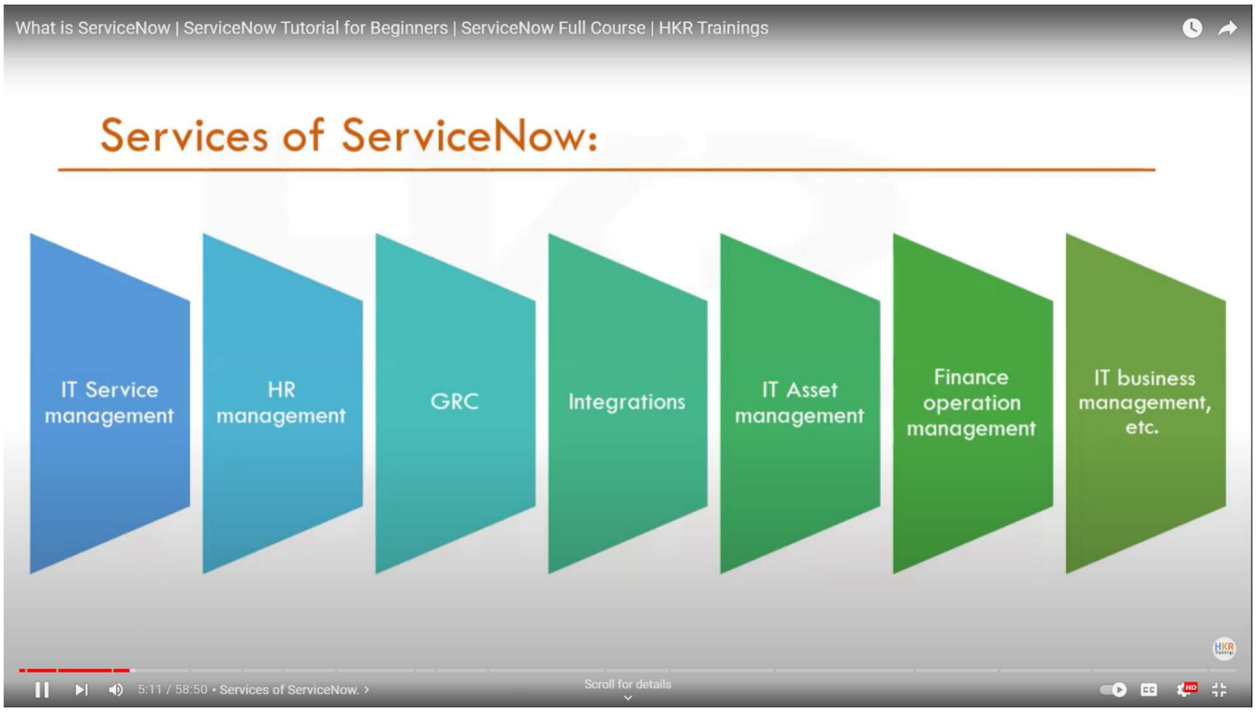
1. **Document Changes:**
    - Keep track of any modifications you make to the widget for future reference and maintenance.
2. **Regular Updates:**
    - Review and update the widget as needed to ensure compatibility with ServiceNow updates and changes in business requirements.

Video 2:

What is ServiceNow:



Services of ServiceNow:



How to get free SNOW instances?:

To get a free instance of ServiceNow (often referred to as SNOW), you can explore several options:

1. ServiceNow Developer Program

- **Developer Instance:** ServiceNow offers a free personal developer instance through its Developer Program. This instance is intended for learning, development, and testing purposes.
- **Registration:** Sign up for a free account on the Developer Portal. Once registered, you can request a personal instance, which is usually available within minutes.
- **Features:** The developer instance includes most of the features and capabilities of a full ServiceNow instance, allowing you to explore and experiment with various functionalities.

2. ServiceNow Community

- **Community Access:** Joining the ServiceNow Community provides access to discussions, knowledge articles, and sometimes, special offers or access to resources. While this doesn't directly provide a free instance, it's a valuable resource for learning and networking.

3. ServiceNow Training and Certification

- **Training Courses:** Some ServiceNow training and certification courses include access to a temporary instance for hands-on practice. Check with ServiceNow or accredited training providers for specific course details.

4. Educational Institutions

- **Academic Programs:** If you are a student or associated with an educational institution, ServiceNow may offer special programs or discounts through academic partnerships. Contact ServiceNow or your institution's IT department for more information.

5. ServiceNow Trial

- **Trial Requests:** Occasionally, ServiceNow offers trial instances for potential customers. These trials are typically limited in duration and designed for evaluating the platform. Contact ServiceNow sales or visit their website to inquire about any current trial offers.

Instance Activity:

Instance activity in ServiceNow refers to the various operations and actions that occur within a ServiceNow instance. This includes user interactions, system processes, and data manipulations. Understanding instance activity is crucial for managing, monitoring, and optimizing your ServiceNow environment. Here's an overview of key aspects related to instance activity:

1. User Activity

- **Login and Usage:** Tracks when users log in and their activities, such as creating, updating, or deleting records.
- **Audit Logs:** Records user actions and changes made to records, providing an audit trail for security and compliance purposes.
- **Performance Monitoring:** Observes how user interactions impact system performance, helping to identify bottlenecks or inefficiencies.

2. System Processes

- **Scheduled Jobs:** Refers to background processes and scheduled tasks that run at specified intervals, such as data imports or automated notifications.
- **Business Rules:** Executes server-side scripts based on specific triggers or conditions, such as record creation or updates.
- **Workflows:** Manages complex processes and tasks through automated workflows, including approvals and task assignments.

3. Data Activity

- **Record Changes:** Monitors changes to records, including creation, modification, and deletion, often tracked through audit logs.

- **Data Import:** Tracks the activity related to importing data into ServiceNow from external sources or systems.
- **Query Performance:** Assesses the performance of queries and data retrieval operations to optimize database performance.

4. System Logs

- **System Log:** Provides a comprehensive log of system events, including errors, warnings, and informational messages.
- **Application Log:** Logs events related to specific applications and modules, useful for debugging and troubleshooting.

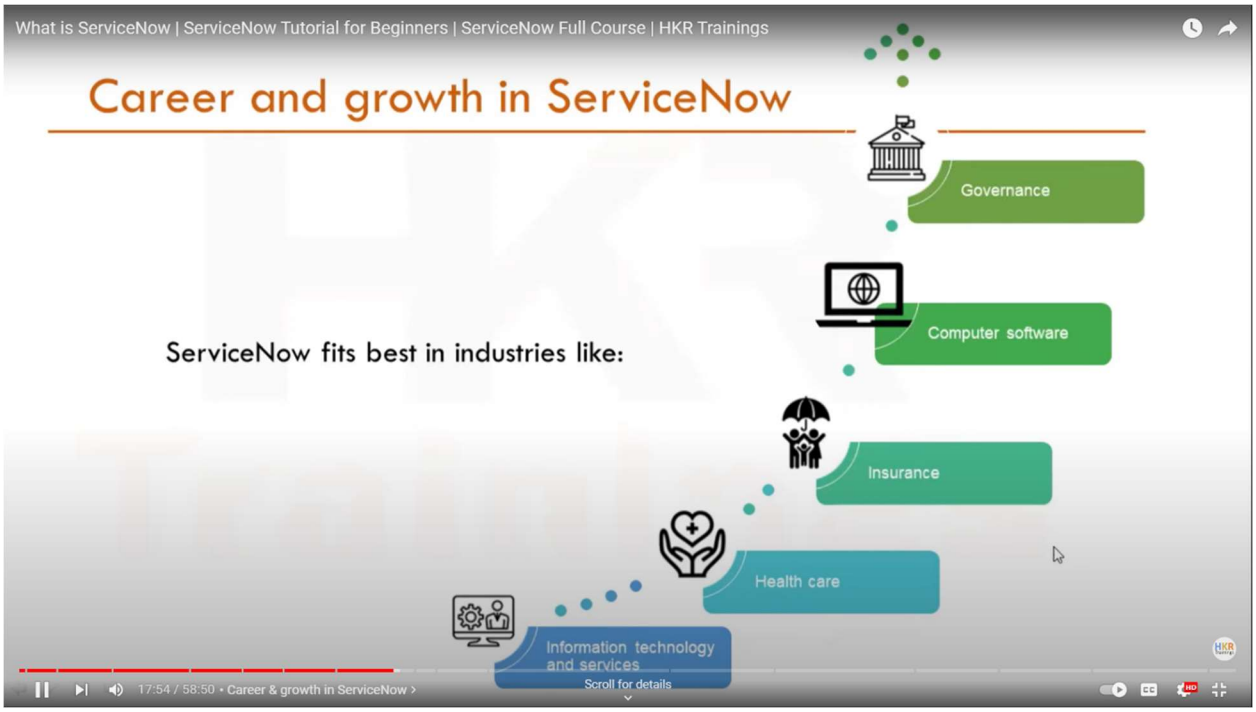
5. Monitoring Tools

- **Performance Analytics:** Offers dashboards and reports to visualize and analyze instance activity, including key performance indicators and trends.
- **Instance Scan:** Performs health checks and scans to identify issues or potential improvements in the instance configuration.

6. Security and Compliance

- **Access Control Logs:** Monitors access to data and system resources, ensuring compliance with security policies.
- **Change Management:** Tracks changes to the system and applications to ensure they are documented and approved.

Career and growth in ServiceNow:



List:

In ServiceNow, **lists** are used to display and manage collections of records from a table. Lists provide a structured way to view and interact with multiple records at once. Here's a summary of key aspects related to lists in ServiceNow:

1. List Views

- **Default View:** The standard view of records for a table, displaying columns with data.
- **List Layouts:** Customizable layouts to show specific fields and columns.

2. List Features

- **Sorting:** Organize records by ascending or descending order based on column values.
- **Filtering:** Apply filters to show records that meet specific criteria.
- **Grouping:** Group records by one or more columns for better organization and analysis.

3. List Actions

- **Inline Editing:** Edit fields directly within the list view without opening individual records.

- **Context Menus:** Access additional actions like edit, delete, or view details through context menus (right-click or options menu).
- **Mass Update:** Apply changes to multiple records simultaneously.

4. List Controls

- **Pagination:** Navigate through large sets of records using pagination controls.
- **Search:** Use the search box to find records based on keywords or criteria.

5. List Personalization

- **Personal Lists:** Users can create personal lists to customize their view of data according to their needs.
- **Saved Filters:** Save frequently used filters for easy access in the future.

6. List Columns

- **Column Configuration:** Customize which columns are displayed and their order.
- **Column Preferences:** Adjust settings like width and visibility of individual columns.

7. Related Lists

- **Sub-lists:** Display related records from other tables within the main record view. For example, displaying related incidents within an account record.

8. List Reporting

- **Exporting Data:** Export list data to formats like CSV, Excel, or PDF.
- **Performance Analytics Integration:** Use data from lists for performance analytics and reporting.

9. List Widgets (Service Portal)

- **Service Portal Lists:** Display lists of records in Service Portal widgets for user interaction and management.

10. List Control Scripts

- **Client Scripts:** Customize list behavior with client-side scripts for enhanced interaction.

Lists are a fundamental component of ServiceNow, providing a powerful way to manage and interact with data efficiently.

Forms:

In ServiceNow, **forms** are used to display and manage individual records from a table. They are crucial for data entry, editing, and viewing within the ServiceNow platform. Here's a concise overview of forms in ServiceNow:

1. Form Structure

- **Header:** Contains information such as record title, form actions (e.g., save, delete), and related links.
- **Tabs:** Organize fields into sections, making the form easier to navigate.
- **Fields:** Display individual pieces of data, such as text fields, drop-down lists, or checkboxes.
- **Related Lists:** Show related records from other tables within the form, such as tasks associated with an incident.

2. Form Types

- **Standard Forms:** Basic forms used for displaying and editing records.
- **Catalog Item Forms:** Custom forms used in Service Catalog for service requests and orders.
- **Record Producers:** Forms that allow users to create new records in a specific table via the Service Catalog.

3. Form Features

- **Field Types:** Various field types include text, number, date, choice, reference, and more.
- **Form Layouts:** Customize the arrangement of fields and sections to fit user needs.
- **Field Attributes:** Configure field properties such as mandatory, read-only, or hidden.

4. Form Actions

- **Submit:** Save the record and trigger any associated workflows or notifications.
- **Cancel:** Discard changes and return to the previous view.
- **Update:** Save changes to an existing record.

5. Form Personalization

- **Form Designer:** A visual tool for customizing form layouts, including adding or rearranging fields and sections.

- **UI Policies:** Control field behavior based on conditions, such as making fields mandatory or visible.
- **Client Scripts:** Add custom scripts to enhance form functionality and interactivity.

6. Form Validation

- **Business Rules:** Enforce data validation and business logic on the server side.
- **Client Scripts:** Validate data and control form behavior on the client side before submission.

7. Form Views

- **Default View:** The standard form layout used for most users.
- **Personalized Views:** Users can customize their own form views to show relevant fields or sections.

8. Form Controls

- **Form Actions Menu:** Provides options for performing actions on the record, such as adding comments or attaching files.
- **Contextual Menus:** Access additional actions or related records via menus or buttons.

9. Form Security

- **Access Controls:** Define who can view, create, update, or delete records based on roles and permissions.
- **Field-Level Security:** Control access to specific fields within a form.

10. Form Performance

- **Load Optimization:** Ensure forms are optimized for performance by minimizing complex scripts and queries.
- **Testing:** Regularly test forms to ensure they function correctly across different scenarios and user roles.