SERVICENOW

Week : 1

Module 1 - ServiceNow Platform and Development Fundamentals

1.What is ServiceNow ?

- ServiceNow targets mid to large-enterprise company
- Bill McDermott currently serving as the CEO of ServiceNow
- Fred Luddy ServiceNow founder and current board chairman Born in New Castle Former CTO of Peregrine Systems
- 2003:Fred Luddy founds the company as GlideSoft
- 2006: Company name changed from GlideSoft to ServiceNow.
- 2012:NOW becomes a publicly traded company
- 2018: ServiceNow #1 on Forbes most innovative companies
- 2019:Bill McDermott named CEO of ServiceNow
- Information Technology (IT): usually does not produce revenue is an expense (oftentimes the largest) ;exists to enable or enhance the ability of the revenue-producing business people; is a necessary evil.

- **ServiceNow is a software company based in Santa Clara, California, founded by Fred Luddy in 2003, to solve problems large enterprises face with traditional IT delivery by providing a robust, simple to use, cloud-based environment in which businesspeople can solve the business problems themselves.**

2. ServiceNow Platform Overview

The Now Platform

- The Now Platform is an Application Platform as a Service (APaaS).
- ServiceNow is a cloud-based.
- ServiceNow provides and supports the infrastructure computer resources.
- ServiceNow provides a platform upon which you can develop your own custom solutions.
- ServiceNow provides a robust set of applications and workflows to support most common business processes.
- All applications (OOB and custom) for the entire enterprise are supported by a single, common, data-model and database.

Applications and Workflows

ServiceNow comes with a robust suite of applications which are categorized (functionally) into 4 primary workflows:

- IT Workflows: 79 applications that support internal IT functions
- Employee Workflows: 43 applications targeted at the needs of employees
- Customer Workflows: 93 applications that support functions related to customers
- Creator Workflows: 23 applications designed to enable ServiceNow platform development and operations support

Now Platform Architecture (2)

When you purchase an instance, it is ServiceNow's responsibility to support the IT infrastructure and compute resources needed to enable and secure that instance.

- Backups & Security

  ServiceNow provides 4 weekly full data backups and 6 days of daily differential backups. The entire platform is secured using multiple technologies which have been certified by third-party security organizations.

- Domain Separation (multi-tenancy)
  - The ServiceNow platform provides the ability to separate data, processes, and administrative tasks on an instance into logical groupings called domains.
  - All users can potentialy see records from the 'global domain', but only users who belong to a domain can see domain-specific records.

Role-based Access

★ Not every member of an organization needs access to all information all the time. ServiceNow uses role-based user can get the information they need, and no more. The primary components include:

★ A User is an individual that has been given access to an instance. Users are usually assigned to 1 or more groups and can be granted muitipie roles. A user with no roles assigned is called a self-service user. They can login and access actions like viewing the homepage, Service Catalog, articles, and surveys.

★ A Group is a set of users who share a common purpose and need access to similar data. Multiple roles can be assigned to a single group.

★ A Role is a collection of permissions. A role can be assigned to an individual user, a group of users, or another role. Multiple roles can be assigned to a single role. It's best to assign roles to a group rather than an individual user.

User Authentication

❑ When a user attempts to login to an instance, ServiceNow validates their identity and enables access to functions and data based upon their related groups and roles. The platform can support several methods of user authentication including:

• Local database authentication
• OAuth 2.0 |
• External Single Sign-on (SSO)
• Digest Token
• LDAP
• Multi-factor Authentication

3.ServiceNow User Interface Overview

ServiceNow User Interface Overview

The ServiceNow User Interface (UI) is designed to be user-friendly, allowing users to efficiently navigate and interact with the platform. Here's an overview of its key elements:

Key Elements of the Interface:

1. **Application Navigator:**
   - A menu on the left side of the screen that provides access to all available applications and modules. Users can search and filter for specific applications or modules, making navigation faster.

2. **Global Search:**
   - A search bar at the top of the interface that allows users to search across the entire platform, including records, knowledge articles, and other data. It provides quick access to information without navigating through menus.

3. **Favorites:**
   - Users can bookmark frequently used applications, modules, or records to quickly access them from the "Favorites" section in the Application Navigator.

4. **History:**
   - Tracks recently accessed records and modules, allowing users to quickly revisit them. The "History" section is located below the "Favorites" in the Application Navigator.

5. **Connect Chat:**
   - An integrated messaging feature that enables real-time communication between users within the platform. It allows for collaboration on tasks, incidents, and other records.

6. **Contextual Help:**
   - A feature that provides on-demand help and guidance related to the current screen or task. It offers articles, documentation, and tips to assist users in real-time.

Security and Scripting:

1. **Access Control Lists (ACLs):**
   - ACLs define the permissions for users to access and interact with different parts of the platform, such as tables, records, and fields. They are essential for ensuring data security and user access management.

2. **UI Policies:**
   - UI Policies dynamically change the behavior and appearance of forms in real-time, such as making fields mandatory, read-only, or hidden based on specific conditions.

3. **Business Rules:**
   - Server-side scripts that execute whenever records are inserted, updated, deleted, or queried. Business Rules automate processes and enforce business logic across the platform.

4. **Client Scripting:**
   - Scripts that run on the client-side (i.e., in the user's browser) to manage and manipulate the user interface. Client scripts include onLoad, onChange, onSubmit, and onCellEdit, enabling real-time interaction and validation.

These components and features make the ServiceNow UI intuitive and powerful, enabling users to perform their tasks efficiently while maintaining security and consistency across the platform.

4.ServiceNow Branding Overview

ServiceNow allows organizations to customize and brand their instances to align with their corporate identity. Branding the ServiceNow platform enhances the user experience and ensures consistency with the company’s visual standards.

1. Branding Introduction:

Branding in ServiceNow involves customizing the look and feel of the platform’s user interface (UI). This can include modifying colors, logos, fonts, and layouts to match the organization’s brand guidelines. The branding process affects various parts of the platform, including the ServiceNow Portal,

dashboards, and forms.

2. Company Guided Setup:

The **Company Guided Setup** is a step-by-step wizard within ServiceNow that helps administrators and developers configure and brand their instance. It simplifies the branding process by guiding users through tasks such as:

- Setting up company logos and themes.
- Configuring basic platform settings like language, date format, and time zone.
- Customizing the look and feel of the ServiceNow instance to reflect the organization's branding.

3. ServiceNow Portal:

The **ServiceNow Service Portal** is a customizable, self-service interface that users interact with to access various services, submit requests, and find information. Branding the Service Portal involves:

- **Themes and Colors:** Adjusting the color scheme to match the company’s branding.
- **Logos and Images:** Adding company logos, banners, and other imagery.
- **Widgets:** Customizing or creating new widgets to provide functionality in line with the company’s needs.
- **Layouts:** Modifying the layout of the portal pages to improve navigation and user experience.

4. UI Builder:

UI Builder is a tool within ServiceNow that allows developers and designers to create and customize workspaces and portal pages using a drag-and-drop interface. It enables detailed branding and customization by:

- **Designing Custom Interfaces:** Creating unique, branded experiences without needing extensive coding skills.
- **Reusable Components:** Utilizing pre-built components like headers, footers, and content blocks, which can be customized to fit the brand.
- **Responsive Design:** Ensuring that the branded interfaces are optimized for different devices and screen sizes.

- **Real-Time Preview:** Viewing changes in real-time to see how branding adjustments impact the user experience.

UI Builder offers a flexible and powerful way to implement and manage branding across different areas of the ServiceNow platform.

Benefits of Branding ServiceNow:

- **Consistency:** Ensures the platform aligns with the organization's visual identity, enhancing user familiarity and trust.
- **Improved User Experience:** Tailoring the interface to meet user needs and expectations, making the platform more intuitive and user-friendly.
- **Professionalism:** Reflects the company's commitment to quality and attention to detail.

By leveraging these tools and features, organizations can effectively brand their ServiceNow instance to create a cohesive and engaging user experience.

5.ServiceNow Lists and Filters Overview

In ServiceNow, lists and filters are essential tools for managing and viewing records within the platform. They allow users to efficiently navigate and interact with large sets of data, such as incidents, changes, or service requests.

1. ServiceNow List View Interface:

The **List View** interface displays records from a table in a grid or spreadsheet-like format. Each row in the list represents a single record, while each column represents a field within the record.

- **Columns:** Represent the fields in the records, such as "Number," "Priority," "State," and "Assigned To."
- **Rows:** Each row corresponds to an individual record, such as a specific incident or change request.
- **Title Bar:** Displays the name of the table, the number of records, and controls for configuring the list view.
- **List Toolbar:** Contains options for filtering, searching, grouping, and exporting list data.

2. Standard Paradigm:

The standard paradigm for working with lists in ServiceNow includes:

- **Viewing Records:** Users can view records in a table format, making it easy to scan and locate specific information.
- **Editing Inline:** Users can edit certain fields directly within the list without opening the full record.
- **Sorting and Grouping:** Lists can be sorted by clicking on column headers and grouped by specific fields to organize data more effectively.
- **Pagination:** If there are many records, the list will be paginated, allowing users to navigate through multiple pages of records.

3. List Control:

List Control refers to the set of tools and options available for managing and interacting with lists:

- **Personalize List Columns:** Users can add, remove, or rearrange columns in the list to display the most relevant information.
- **List Layout Options:** Users can switch between different layouts (e.g., compact, detailed) to better suit their viewing preferences.
- **Export:** Allows users to export the list data in various formats such as Excel, CSV, or PDF.
- **Update All:** Enables users to make bulk updates to selected records directly from the list view.

4. Filter Conditions:

Filter Conditions allow users to narrow down the records displayed in a list based on specific criteria. Filters are essential for focusing on relevant data without having to sift through unrelated records.

- **Creating Filters:** Users can create filters by specifying conditions that records must meet to be displayed. Conditions can be based on any field in the table (e.g., "State is Open," "Priority is High").
- **Multiple Conditions:** Users can combine multiple conditions using AND/OR logic to create complex filters.
- **Saving Filters:** Frequently used filters can be saved for quick access,

allowing users to apply them easily in the future.
- **Quick Filters:** Users can apply quick filters by right-clicking on a field value in the list and choosing to filter by that value.

5. Refresh List:

Refresh List is a feature that allows users to reload the data in a list to ensure it reflects the most current information.

- **Manual Refresh:** Users can manually refresh the list by clicking the "Refresh" button on the List Toolbar.
- **Auto-Refresh:** In some cases, lists can be set to automatically refresh at regular intervals, ensuring that users always see the latest data.

Summary:

ServiceNow's lists and filters provide powerful tools for managing and interacting with data. The list view interface, combined with controls, filter conditions, and refresh options, allows users to efficiently navigate, view, and update records, ensuring that they can quickly access the information they need.

6.Forms in ServiceNow

Forms in ServiceNow are used to create, view, and edit records in the platform. They provide a structured interface for users to interact with data, making it easy to input and manage information within various tables.

1. The Standard Layout:

- **Header:** The top section of the form, which usually contains the record's title, the Save/Submit buttons, and key information like the record number and state.
- **Fields:** These are the individual data entry points on the form, each corresponding to a column in the database table.
- **Sections:** Forms are often divided into sections to group related fields together, making the form easier to navigate and understand.

2. Form Field Types:

Forms can contain various types of fields, including:

- **Text Fields:** Simple input fields for entering text.
- **Choice Lists (Dropdowns):** Allows users to select a value from a predefined list.
- **Date/Time Pickers:** Fields for selecting dates and times.
- **Reference Fields:** Links to records in another table, allowing users to reference related data.
- **Checkboxes:** For binary options (e.g., Yes/No, True/False).
- **HTML Fields:** Allows for rich text formatting, used for longer, formatted content.

3. Saving Changes:

- **Save/Update:** Commits any changes made to the form and closes it, returning the user to the previous screen.
- **Insert:** Saves the form data as a new record but does not close the form. Instead, it redirects to a blank form, ready for another entry.
- **Insert & Stay:** Saves the form data as a new record but keeps the user on the current form, allowing for further edits.

4. Form Sections:

- **Form Sections:** These are logical groupings of related fields within a form, often separated by headings or collapsible panels. They help users focus on specific parts of the form without being overwhelmed by all fields at once.

5. Related Lists & Formatters:

- **Related Lists:** These are lists of records from other tables that are linked to the current record. For example, a list of all incidents related to a specific problem record.
- **Formatters:** Special UI elements that enhance the display of a form. Examples include activity streams (showing recent changes or comments) and journal fields (for tracking notes).

6. Form Views:

- **Form Views:** Different layouts of the same form, showing different sets of fields depending on the user's role or the task at hand. For example, a "Manager View" might show more detailed financial information, while a "Technician View" focuses on operational data.
- **Creating & Editing Views:** Administrators can create and customize views to suit different user needs, tailoring the form to display only relevant information.

7. Form Personalization:

- **Form Personalization:** Allows individual users to modify how forms appear to them. Users can add or remove fields, rearrange sections, and set preferences to streamline their workflow.

8. Adding Attachments:

- **Adding Attachments:** Users can upload files to a record via the form interface. This is useful for attaching documents, images, or other files related to the record.

9. Form Templates:

- **Form Templates:** Pre-defined forms with fields already filled in with default values. Templates are used to streamline the data entry process, particularly for records that require similar information.

Summary:

Forms in ServiceNow are a central tool for data entry and record management. They are highly customizable, with various field types, sections, views, and personalization options to cater to different user needs and roles. Understanding how to work with forms efficiently is key to using ServiceNow effectively.

7.A Hands-on ServiceNow Tool Demo Overview

This guide covers a hands-on introduction to using the ServiceNow platform, exploring its core features, navigation, and essential tools for users.

1. Logging In:

- **Accessing ServiceNow:** Users log in to the ServiceNow instance using their credentials through a web browser. The URL for the instance is typically provided by the organization.
- **User Interface Landing Page:** Upon logging in, users are directed to the home page, which provides a dashboard view of their tasks, announcements, and key metrics.

2. ServiceNow Next Experience UI:

- **Next Experience UI:** This is the modern user interface of ServiceNow, offering a more intuitive and responsive design. It includes enhancements like customizable workspaces and an updated layout to improve user experience.

3. The Navigation Bar:

- **Navigation Bar:** Located at the top of the ServiceNow interface, it provides quick access to key functions like the Global Search, user profile settings, and application navigator.

4. ServiceNow Applications Overview:

- **ServiceNow Applications:** ServiceNow is built on a modular architecture, offering various applications for different business functions, such as IT Service Management (ITSM), HR Service Delivery (HRSD), Customer Service Management (CSM), and more. Each application contains specific modules related to its function.

5. The Application Navigator:

- **Application Navigator:** This is the menu on the left side of the screen, where users can browse through the available applications and modules. Users can search for specific applications or modules using the search bar at the top of the navigator.

6. The ServiceNow Store:

- **ServiceNow Store:** An online marketplace where users can browse and

install third-party applications and integrations that extend the functionality of their ServiceNow instance. The Store offers a wide range of apps for different use cases, developed by both ServiceNow and its partners.

7. ServiceNow Application Training and Certifications:

- **Training and Certifications:** ServiceNow offers extensive training resources and certifications for users to deepen their understanding of the platform. Certifications are available for different roles, such as system administrators, developers, and implementation specialists, helping professionals validate their skills.

8. Working with Lists and Forms Overview:

- **Lists and Forms:** Lists display records in a tabular format, while forms provide a structured view for entering and editing individual records. Both are fundamental components of the ServiceNow platform for data management.

9. List Views:

- **List Views:** Users interact with data in a grid-like structure, where they can sort, filter, group, and manage records. List views can be customized to display specific columns, and users can save their preferred configurations for future use.

10. Form Views:

- **Form Views:** Forms are used to create, view, and edit individual records. Users can switch between different form views depending on their role or the specific task at hand. Views can be customized to display only the most relevant fields.

11. Knowledge Management in ServiceNow:

- **Knowledge Management:** ServiceNow's Knowledge Management application allows organizations to create, manage, and share knowledge articles. These articles help users find solutions to common problems, reducing the number of support requests.

12. The ServiceNow Database:

- **ServiceNow Database (CMDB):** The Configuration Management Database (CMDB) is the central repository for all configuration items (CIs) and their relationships. It forms the backbone of the ServiceNow platform, providing a single source of truth for IT assets, services, and dependencies.

Summary:

This hands-on demo introduces the fundamental elements of the ServiceNow platform, including how to navigate the user interface, work with lists and forms, and access training resources. Understanding these core features is essential for effectively using ServiceNow in day-to-day operations.

8.Introduction to Importing Data in ServiceNow

Importing data into ServiceNow is a fundamental task that allows organizations to bring external data into the platform. This process is crucial for initializing the ServiceNow instance with existing records, integrating with other systems, and keeping data up-to-date.

1. Importing Data into ServiceNow:

Data can be imported into ServiceNow using various methods, each suitable for different types of data and use cases:

- **Data Sources:** ServiceNow supports various data sources such as CSV files, Excel spreadsheets, XML files, and JDBC connections to databases. These data sources provide the raw data that needs to be imported.

- **Import Sets:** Import sets are temporary tables used to store data being imported before it is mapped to the target tables. This allows for data transformation and validation before final importation.

- **Transform Maps:** A key feature that maps the data fields from the import set to the target tables in ServiceNow. Transform maps allow for data transformation, such as converting data formats or applying logic during the import process.

- **Coalesce Fields:** Fields that are used to determine whether a record already exists in the target table. If a match is found, the existing record is updated; otherwise, a new record is created. This prevents duplicate entries.

- **Scheduled Data Imports:** ServiceNow allows for scheduled data imports, enabling automated and recurring data import processes from external sources.

2. Importing Data via Integrations:

Integrating with external systems is another powerful way to bring data into ServiceNow. Integration methods include:

- **Web Services (REST/SOAP APIs):** ServiceNow provides robust support for REST and SOAP web services, allowing data to be imported via API calls. This is commonly used for real-time integrations where data needs to be synced across systems continuously.

- **IntegrationHub:** A no-code/low-code platform in ServiceNow that facilitates integration with external applications and services. IntegrationHub provides pre-built connectors and flows that simplify the integration process.

- **Inbound Email Actions:** ServiceNow can process incoming emails to create or update records. This method is useful for importing data that is received via email, such as incident reports or service requests.

- **MID Server:** The ServiceNow MID (Management, Instrumentation, and Discovery) Server facilitates secure data imports from on-premises systems or databases into the ServiceNow cloud. It is essential for organizations with strict security requirements or legacy systems.

Key Steps in the Data Import Process:

1. **Prepare the Data:** Ensure that the data is clean, formatted correctly, and ready for import. This includes removing duplicates, correcting data types, and ensuring consistency.

2. **Create a Data Source:** Define the source of the data within ServiceNow,

specifying the format and location of the data file or external source.

3. **Load Data into Import Set:** Import the data into an import set table where it can be reviewed and transformed as needed.

4. **Define Transform Map:** Create and configure a transform map to map fields from the import set to the target table, applying any necessary data transformations.

5. **Run the Transform:** Execute the transform map to move data from the import set to the target table, updating existing records or creating new ones as needed.

6. **Validate Data:** After the import, validate the data in the target table to ensure accuracy and completeness.

Benefits of Importing Data into ServiceNow:

- **Data Centralization:** Allows for a unified view of data within ServiceNow, improving decision-making and service delivery.
- **Automation:** Reduces manual data entry, saving time and reducing the risk of errors.
- **Integration:** Seamlessly connects ServiceNow with other business systems, ensuring data consistency across the organization.

Summary:

Importing data into ServiceNow, whether through direct uploads or integrations, is a critical task that enhances the platform's functionality and ensures that it remains a central hub for organizational data. Properly setting up data imports and integrations ensures that data is accurate, consistent, and accessible within ServiceNow.

9.Creating a Data Source in ServiceNow

Creating a Data Source in ServiceNow is the first step in integrating external data into the platform. Data Sources define where the data is coming from, how it is formatted, and how it should be processed once it is loaded into ServiceNow.

1. What is a Data Source?

A **Data Source** in ServiceNow is a record that defines the external data source from which data will be imported into the platform. It specifies the format, location, and method of data retrieval, allowing ServiceNow to interact with various external systems, files, or databases.

2. Steps to Create a Data Source:

Here's how to create a Data Source in ServiceNow:

Step 1: Navigate to Data Sources
- Go to the **Application Navigator** in ServiceNow and type "Data Sources" in the search bar.
- Select **System Import Sets > Administration > Data Sources** from the menu.

Step 2: Create a New Data Source Record
- Click on the **New** button to create a new Data Source.
- Fill in the necessary fields to define your Data Source:

  - **Name:** Enter a descriptive name for the Data Source (e.g., "Employee Data from HR System").
  - **Import Set Table:** This is automatically created and named after the Data Source. It temporarily holds the imported data before transformation.
  - **Type:** Choose the type of data source, such as File, JDBC, LDAP, Web Service, etc.
  - **File Retrieval Method (for File Data Sources):** If the source is a file, specify how ServiceNow should access the file (e.g., attach manually, FTP, or HTTP).
  - **Format:** Specify the data format (e.g., CSV, Excel, XML).

Step 3: Configure Additional Settings
- Depending on the type of Data Source selected, additional configuration settings may be required:
  - **JDBC Data Sources:** Define the connection parameters like database type, connection URL, username, and password.
  - **Web Service Data Sources:** Specify the endpoint URL, HTTP method

(GET, POST), and any required headers or authentication credentials.
  - **File Data Sources:** Upload the file or specify the path for retrieval if using FTP or HTTP.

Step 4: Test the Data Source
- After configuring the Data Source, it's important to test the connection or data retrieval to ensure that ServiceNow can successfully access the external data.

Step 5: Save the Data Source Record
- Click **Save** or **Submit** to save the Data Source record in ServiceNow.

3. Loading Data from the Data Source:

Once the Data Source is created, data can be imported into ServiceNow:

- **Load Data:** ServiceNow loads the data from the external source into an **Import Set Table** based on the defined Data Source.
- **Transform Map:** A **Transform Map** is then used to map the data from the Import Set Table to the appropriate target table in ServiceNow.
- **Transform Data:** Execute the transformation to move the data from the Import Set Table to the target table, ensuring it is correctly formatted and integrated into the ServiceNow database.

4. Summary:

Creating a Data Source in ServiceNow is a crucial step in the data integration process. It defines how external data will be brought into the platform, whether from files, databases, web services, or other systems. Properly setting up and testing a Data Source ensures that data is accurately and reliably imported into ServiceNow, where it can be used for reporting, automation, and service delivery.

10.Understanding Import Sets in ServiceNow

Import Sets in ServiceNow are a powerful tool used to bring external data into the platform. They act as an intermediary, temporarily storing incoming data before it is transformed and mapped to the target tables within ServiceNow.

1. What are Import Sets?

An **Import Set** is a table in ServiceNow that temporarily holds data imported from an external source. The data in an Import Set can come from various data sources, such as files, databases, or web services. Once the data is loaded into the Import Set, it can be processed, transformed, and mapped to the appropriate target tables within the platform.

2. How Import Sets are Created:

Import Sets are created automatically when a Data Source is defined, but they can also be created manually if needed.

Step 1: Defining a Data Source:
- When a Data Source is created in ServiceNow, an Import Set table is automatically generated to store the incoming data.
- The Import Set table is named after the Data Source and can be customized if required.

Step 2: Loading Data into the Import Set:
- The data from the external source is loaded into the Import Set table. This can be done manually by uploading a file or automatically through a scheduled data import.
- The Import Set table now holds all the raw data from the source, ready for transformation.

3. Transforming Data in Import Sets:

Once the data is loaded into an Import Set, it needs to be transformed to fit the structure and format of the target table in ServiceNow. This is done using **Transform Maps**.

Transform Maps:
- **Definition:** A Transform Map is a set of field mappings between the fields in the Import Set table and the fields in the target table.
- **Field Mapping:** Each field in the Import Set is mapped to a corresponding field in the target table. This mapping ensures that the data is correctly aligned when it is moved to the target table.
- **Data Transformation:** Transform Maps can include scripts and logic to

manipulate data during the transformation process. For example, you can format dates, convert data types, or apply calculations before the data is transferred.

Step-by-Step Transformation:
1. **Create a Transform Map:** Define a new Transform Map by specifying the source Import Set table and the target table in ServiceNow.
2. **Map Fields:** Map each field from the Import Set to a field in the target table. Use scripting if necessary to transform the data during the mapping process.
3. **Set Coalesce Fields:** Identify one or more fields in the Import Set that will be used to check for existing records in the target table. If a match is found, the existing record is updated; otherwise, a new record is created.
4. **Run the Transform:** Execute the Transform Map to transfer the data from the Import Set to the target table. The data is now in its final form within the ServiceNow database.

4. Advantages of Using Import Sets:

- **Flexibility:** Import Sets allow for complex data transformations, making it possible to import data from various sources and formats into a standardized structure within ServiceNow.
- **Data Validation:** Data can be validated and cleansed during the transformation process, ensuring accuracy and consistency before it is committed to the target table.
- **Reusability:** Once a Transform Map is created, it can be reused for similar data imports, saving time and effort in future integrations.

5. Summary:

Import Sets in ServiceNow serve as a crucial component in the data import process, acting as a staging area for incoming data. By utilizing Transform Maps, users can effectively transform and map data fields to the appropriate target tables, ensuring that data is accurately integrated into the platform. Import Sets provide flexibility, data validation, and reusability, making them an essential tool for managing data imports in ServiceNow.

11.ServiceNow Transform Maps & Field Maps

Transform Maps and **Field Maps** in ServiceNow are key components in the process of importing, transforming, and mapping data from external sources into the platform. They ensure that data is accurately and efficiently transferred to the appropriate tables within ServiceNow.

1. What is a Transform Map?

A **Transform Map** in ServiceNow is a set of instructions that define how data from an Import Set table is mapped to fields in a target table. The Transform Map controls how incoming data is processed, transformed, and inserted or updated in the target table.

Key Features of Transform Maps:
- **Field Mapping:** Specifies which fields in the Import Set correspond to which fields in the target table.
- **Data Transformation:** Allows the application of scripts and logic to modify or format data during the import process.
- **Coalescing:** Identifies records that already exist in the target table to avoid duplicates.

2. What is a Field Map?

A **Field Map** is a specific mapping between a single field in the Import Set table and a field in the target table. Field Maps are created as part of a Transform Map and determine how each piece of data is transferred and transformed.

Key Features of Field Maps:
- **Source Field:** The field in the Import Set table from which data is being drawn.
- **Target Field:** The field in the target table where the data will be placed.
- **Transformation Logic:** Allows for data transformation, such as converting data types, trimming text, or calculating values.

3. Process of Importing, Transforming, and Mapping Data:

The process of importing and mapping data in ServiceNow using Transform Maps and Field Maps typically involves several steps:

Step 1: Load Data into an Import Set
- **Data Source Creation:** A Data Source is created to define where the external data comes from (e.g., a CSV file, database, or API).
- **Import Set Creation:** The data from the external source is loaded into an Import Set table, which temporarily stores the data before transformation.

Step 2: Create a Transform Map
- **Define the Transform Map:** In ServiceNow, create a new Transform Map by specifying the Import Set table as the source and the target table where the data will be inserted.
- **Map Fields:** Use Field Maps to specify how each field in the Import Set corresponds to a field in the target table.

Step 3: Apply Transformation Logic
- **Data Transformation:** If needed, apply transformation logic within the Field Maps. For example, you might convert a date format, merge two fields into one, or calculate a value based on multiple inputs.
- **Scripting:** ServiceNow allows the use of scripting (e.g., JavaScript) to perform complex transformations during the mapping process.

Step 4: Coalesce to Prevent Duplicates
- **Coalesce Fields:** In the Transform Map, specify one or more fields as "coalesce fields." These fields are used to identify whether a record already exists in the target table. If a match is found, the existing record is updated; otherwise, a new record is created.

Step 5: Execute the Transform Map
- **Run the Transformation:** Execute the Transform Map to move the data from the Import Set table to the target table. The data is now fully integrated into ServiceNow, with transformations applied as needed.

4. Example Use Case:

Suppose you have an external system that tracks employee information and you need to import this data into the ServiceNow HR table. Here's how you would use Transform Maps and Field Maps:

- **Import Set:** The raw employee data (e.g., from a CSV file) is loaded into an Import Set table.

- **Transform Map:** Create a Transform Map that links the Import Set to the ServiceNow HR table.
- **Field Mapping:** Map fields like "Employee ID," "First Name," "Last Name," "Department," and "Job Title" from the Import Set to the corresponding fields in the HR table.
- **Transformation Logic:** Apply any necessary transformations, such as formatting names, or concatenating the first and last names into a full name field.
- **Coalesce Fields:** Use "Employee ID" as the coalesce field to ensure that if an employee record already exists, it is updated instead of creating a duplicate entry.

5. Summary:

Transform Maps and Field Maps in ServiceNow are essential tools for importing and mapping data from external sources. By defining how data is mapped and transformed between Import Set tables and target tables, these tools ensure that data is accurately integrated into the platform. With the ability to apply complex transformation logic and prevent duplicates through coalescing, Transform Maps and Field Maps are vital for maintaining data integrity and consistency in ServiceNow.

12.ServiceNow Incident Management Tutorial and Task Administration

ServiceNow Incident Management is a core component of IT Service Management (ITSM) that helps organizations efficiently handle and resolve incidents. This tutorial covers the key capabilities of ServiceNow's ticket and task management, including Incident, Problem, and Change management, as well as task creation, assignment rules, collaboration, and visual task boards.

1. Incident Management

Incident Management focuses on restoring normal service operation as quickly as possible and minimizing the impact on business operations. It involves tracking and managing incidents reported by users.

Key Capabilities:
- **Incident Creation:** Incidents can be created manually by users through

the ServiceNow interface or automatically through integrations and email.
- **Incident Tracking:** Each incident is assigned a unique number and tracked throughout its lifecycle.
- **Incident Prioritization:** Incidents are categorized and prioritized based on impact and urgency.
- **Resolution and Closure:** Incidents are resolved and closed once the issue is addressed, with documentation of the resolution and any relevant information.

Task Creation and Assignment:
- **Creating Tasks:** Tasks are created within an incident to handle specific actions required to resolve the issue. For example, a task might be created for a technician to perform a system check.
- **Assignment Rules:** Automatic assignment rules can be set up to assign incidents and tasks to appropriate groups or individuals based on predefined criteria (e.g., category, priority, location).
- **Task Assignment:** Users can manually assign tasks to themselves or others, ensuring that responsibilities are clearly defined.

2. Problem Management

Problem Management aims to identify the root cause of incidents and prevent recurrence by addressing underlying issues.

Key Capabilities:
- **Problem Creation:** Problems are created based on recurring incidents or when an incident reveals a potential underlying issue.
- **Root Cause Analysis:** Investigation is conducted to determine the root cause of the problem.
- **Solution Implementation:** Solutions or workarounds are developed and implemented to address the problem.
- **Problem Closure:** Problems are closed once a permanent solution is found and applied.

3. Change Management

Change Management is focused on managing changes to the IT environment in a controlled and systematic manner.

Key Capabilities:
- **Change Requests:** Changes are requested and documented through Change Requests, which outline the details of the proposed change.
- **Change Evaluation:** Changes are evaluated for potential risks and impacts.
- **Approval Process:** Changes must go through an approval process involving stakeholders and change advisory boards.
- **Change Implementation:** Approved changes are implemented and monitored.
- **Change Closure:** Changes are reviewed and closed after implementation to ensure successful deployment.

4. Task Collaboration

Task Collaboration enables team members to work together effectively on incidents and tasks:

- **Comments and Notes:** Users can add comments and notes to incidents and tasks to provide updates, ask questions, and collaborate on resolution.
- **Attachments:** Files and documents can be attached to incidents and tasks to provide additional context or information.
- **Notifications:** Users receive notifications about updates, assignments, and changes to tasks and incidents, ensuring they stay informed.

5. Visual Task Boards

Visual Task Boards provide a visual representation of tasks and their status, helping teams manage and prioritize work more effectively.

Key Features:
- **Board Creation:** Create boards to visualize and manage tasks related to specific incidents, problems, or changes.
- **Card View:** Tasks are represented as cards on the board, which can be moved between columns to reflect their current status (e.g., To Do, In Progress, Done).
- **Customizable Columns:** Customize columns to match the workflow and stages of task completion.
- **Drag-and-Drop Functionality:** Easily update task status and progress by dragging and dropping cards between columns.

6. Summary:

ServiceNow provides comprehensive capabilities for managing incidents, problems, and changes, as well as for creating and handling tasks. The system supports task creation, automatic assignment, collaboration, and visualization through visual task boards. By leveraging these features, organizations can improve efficiency, enhance communication, and ensure effective management of IT services and operations.

13.ServiceNow Reporting Tutorial

ServiceNow offers powerful reporting capabilities that help organizations visualize and analyze data, track performance, and make informed decisions. This tutorial covers the different types of reports, how to create and manage them, and how to share reports with users, groups, or through dashboards.

1. ServiceNow Reporting Capabilities

ServiceNow's reporting capabilities allow users to create custom reports, visualize data, and generate insights from various tables and records within the platform. Reports can be used for operational monitoring, performance analysis, and strategic planning.

2. Types of Reports in ServiceNow

ServiceNow supports several types of reports to cater to different needs:

- **List Reports:** Display data from a list view in a tabular format. Users can filter, sort, and group data within the list.
- **Chart Reports:** Visualize data using charts such as bar charts, pie charts, line charts, and more. Useful for showing trends and comparisons.
- **Pivot Table Reports:** Summarize and analyze data in a pivot table format, allowing users to dynamically group and aggregate data.
- **Performance Analytics Widgets:** Display key performance indicators (KPIs) and metrics on dashboards using widgets that update in real-time.
- **Gauge Reports:** Show performance against targets using gauge charts, which are useful for tracking progress toward goals.

3. Creating and Managing Reports

Creating a Report:

1. **Navigate to Reports:**
   - Go to **Reports** in the Application Navigator and select **View / Run** to access the reporting interface.

2. **Create a New Report:**
   - Click **New** to start creating a report. Choose the type of report you want to create (e.g., List, Chart, Pivot Table).

3. **Select Data Source:**
   - Choose the table or data source from which you want to pull data for the report.

4. **Define Report Criteria:**
   - Set up the criteria for your report, such as filters, conditions, and grouping. This determines which records will be included in the report.

5. **Configure Report Layout:**
   - For List Reports, configure the columns and sorting options.
   - For Chart Reports, set up the chart type, data series, and labels.
   - For Pivot Tables, define rows, columns, and aggregation functions.

6. **Preview and Save:**
   - Preview the report to ensure it meets your requirements. Make adjustments as needed.
   - Save the report by giving it a descriptive name and assigning it to a folder for organization.

Managing Reports:

- **Edit Reports:** Access and modify existing reports by navigating to the report's record and making changes to its criteria, layout, or data source.
- **Run Reports:** Execute reports to view real-time data and insights. Reports can be run on-demand or scheduled to run at specified intervals.
- **Delete Reports:** Remove reports that are no longer needed by navigating to the report record and selecting **Delete**.

4. Sharing Reports

Sharing with Users or Groups:

1. **Report Access Permissions:**
   - Set access permissions for reports by defining who can view, edit, or manage the report. Permissions can be set for individual users or groups.
   - Use the **Sharing** tab in the report settings to manage these permissions.

2. **Send Reports via Email:**
   - Share reports with specific users or groups via email. Configure the report settings to include email recipients and customize the email message.

Sharing via Dashboards:

1. **Create a Dashboard:**
   - Go to **Self-Service > Dashboards** and click **Create New** to build a new dashboard.

2. **Add Reports to Dashboard:**
   - Use the **Add Widget** option to include reports as widgets on the dashboard. Choose the report type and configure how it will be displayed.

3. **Customize Dashboard Layout:**
   - Arrange widgets on the dashboard to create a layout that provides a comprehensive view of your data. Customize the size and position of each widget.

4. **Share Dashboard:**
   - Share the dashboard with users or groups by setting permissions and access levels. Dashboards can also be embedded in ServiceNow portals or other applications.

5. Summary

ServiceNow's reporting capabilities allow users to create and manage a variety of report types to analyze and visualize data. By leveraging list reports, chart reports, pivot tables, and performance analytics, users can gain insights

into their operations and performance. Reports can be shared with specific users, groups, or through dashboards to ensure that relevant information is accessible to stakeholders. Effective reporting in ServiceNow helps organizations make data-driven decisions and monitor key metrics and performance indicators.

14.Low Code/No Code Development

Low Code and **No Code** development platforms are tools that allow users to create software applications with minimal or no traditional coding. These platforms aim to simplify the development process, making it accessible to individuals with varying levels of technical expertise.

1. What is Low Code/No Code Development?

- **Low Code Development:** Involves using a visual interface to design applications with minimal hand-coding. Users can drag and drop components, set up workflows, and configure integrations without writing extensive code.
- **No Code Development:** Allows users to build applications entirely through a visual interface, with no coding required. Users rely on pre-built components and templates to create applications.

2. How Low Code/No Code Software Development Works

- **Visual Design:** Users build applications using a graphical interface that includes drag-and-drop elements, pre-built templates, and customizable components.
- **Workflow Automation:** Users set up workflows and business rules through intuitive interfaces, often using simple logic or condition-based configurations.
- **Integrations:** Platforms provide built-in connectors or APIs to integrate with other systems and data sources.
- **Deployment:** Applications are typically deployed through the platform's hosting infrastructure, with automatic updates and maintenance managed by the provider.

3. Pros and Cons

Pros:

- **Speed of Development:** Rapidly develop and deploy applications due to pre-built components and simplified processes.
- **Accessibility:** Enables non-technical users to create and customize applications, democratizing development.
- **Cost-Effective:** Reduces development costs by minimizing the need for extensive coding and specialist developers.
- **Flexibility:** Allows for quick adjustments and updates to applications without needing to rewrite code.
- **Reduced Complexity:** Simplifies complex processes through visual workflows and automation.

Cons:

- **Limited Customization:** May not support advanced or highly customized features that require traditional coding.
- **Performance Issues:** Applications built on low code/no code platforms may not be as optimized or scalable as those developed with custom code.
- **Vendor Lock-In:** Dependence on the platform's features and limitations can create challenges if you need to switch platforms or integrate with other systems.
- **Security and Compliance:** Potential concerns regarding data security, privacy, and compliance, depending on the platform's capabilities and standards.
- **Learning Curve:** While designed to be user-friendly, there is still a learning curve associated with mastering the platform and its features.

4. Career Opportunities in Low Code/No Code Development

The rise of low code and no code platforms has created various career opportunities, including:

- **Low Code/No Code Developer:** Specializes in using these platforms to build and maintain applications. Involves working with visual development tools and configuring workflows.
- **Business Analyst:** Works with stakeholders to gather requirements and design solutions using low code/no code platforms. Focuses on translating business needs into functional applications.
- **Citizen Developer:** Non-technical users who leverage low code/no code

tools to create applications within their departments or organizations. Often has a background in business or a specific industry.
- **Platform Specialist:** Expert in a particular low code/no code platform. Provides guidance on best practices, platform capabilities, and advanced configurations.
- **Consultant:** Advises organizations on implementing and optimizing low code/no code solutions. Helps with selecting the right platform, designing applications, and integrating with existing systems.
- **Training and Support Specialist:** Provides training, documentation, and support for users of low code/no code platforms. Ensures effective use and adoption of the tools within an organization.

5. Summary

Low code and no code development platforms offer a streamlined approach to building applications with minimal coding. They provide significant benefits in terms of speed, accessibility, and cost, but also come with limitations such as reduced customization and potential performance issues. Career opportunities in this field are diverse, ranging from specialized development roles to business analysis and consulting, reflecting the growing importance of these platforms in modern software development.