MAJESCO

The Future of Insurance Starts Here

Ticket & Screen Plugin

AGENDA

- Dev-Studio Overview
- Dev-Studio Homepage
- Dev Studio Plugins Overview
- Ticket Plugin
- Screen Plugin & properties
- Properties Editor
- Composite Objects
- Best Practices

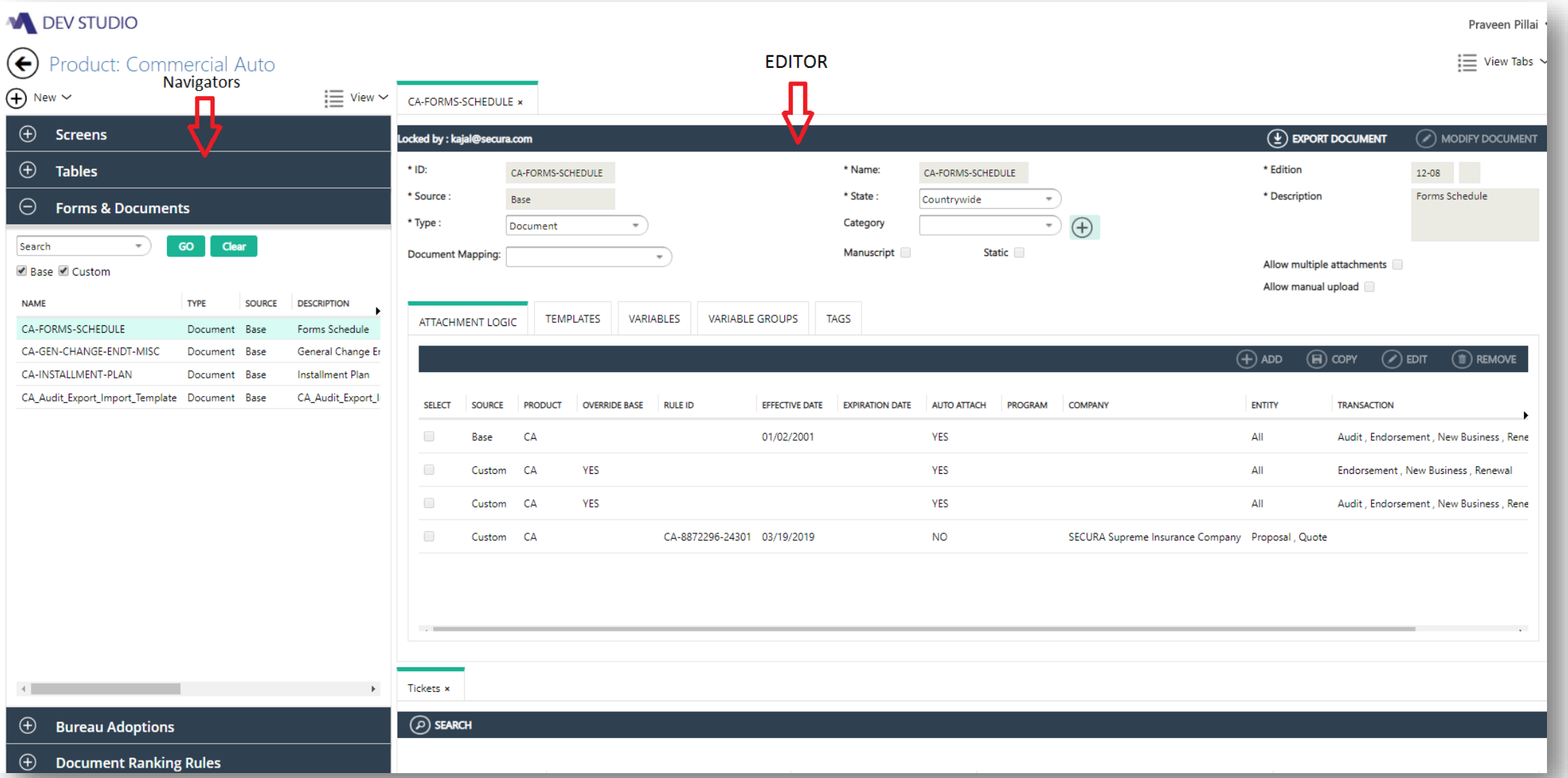
Dev Studio Overview

- Dev Studio is Product Development and Configuration tool.
- Currently it is used for developing and configuring Insurance product for multiple Line of Business like Commercial Auto, General Liability, etc.
- It is also used as a deployment tool.
- **DevStudio** is a configuration engine that allows you to configure your product depending upon your requirements.
- **Policy** reads the configuration defined in Dev Studio and displays the product.
- Majesco platform has 3 components:
- **DevStudio** - Configure changes like creating Product, LOB followed by Build
- **Platform** - Includes core services like logging, security, versioning etc.
- **Policy** - Read the configuration from build installed here.(Target environment)

Dev Studio Homepage

MAJESCO

Navigator & Editor



MAJESCO

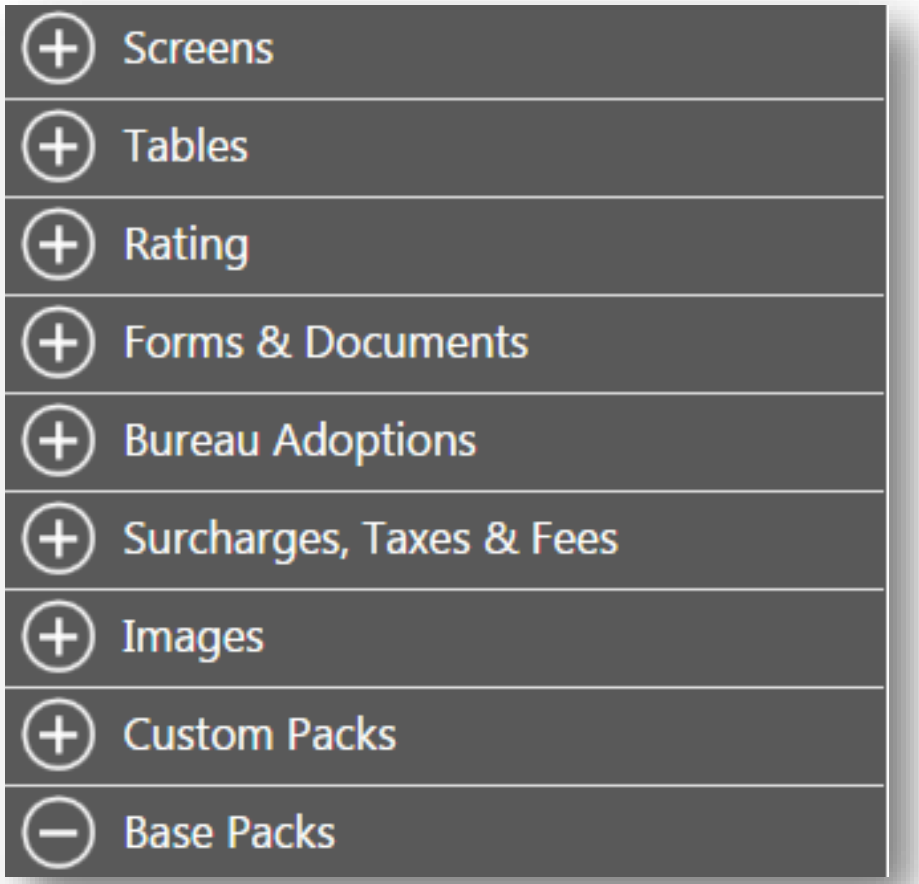
Dev Studio Homepage Overview

Product : This section contains tiles which used to take build & install on specific products. It contents tiles for Bureau(ISO) & Non Bureau(custom) products.

LOB (Line of Business): This section contains tiles which are used to add or modify LOB specific screens , forms, rating algorithm ,rates etc. It contents tiles for Bureau(ISO) & Non Bureau(custom) LOB's.

General : This section contents tiles which are used to add company , deployment environment, new Product ,new LOB, underwriter etc.

Dev Studio Plugins Overview



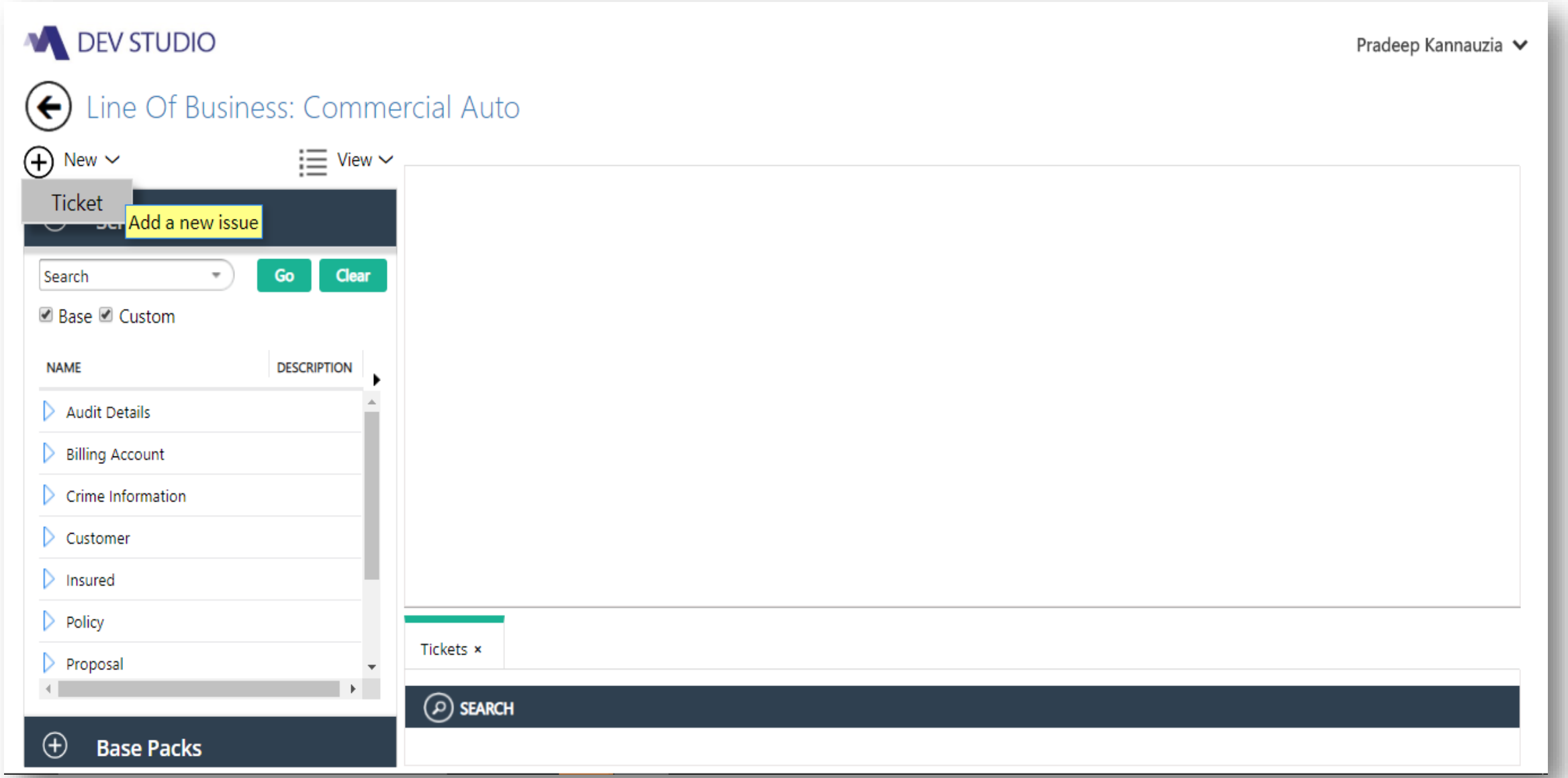
- Used for configuration of Screens
- Used for creating table and Loading Data – Custom/ ISO/ NAIC/
- Used to configure rating methods and steps
- Used for configuring forms and documents
- Used for adopting rates specific to company/ states
- Used for configuring State Surcharges
- Used for configuring images for printing
- Used for creating Custom Packs
- Used for creating Base Packs

MAJESCO

Ticket Plugin

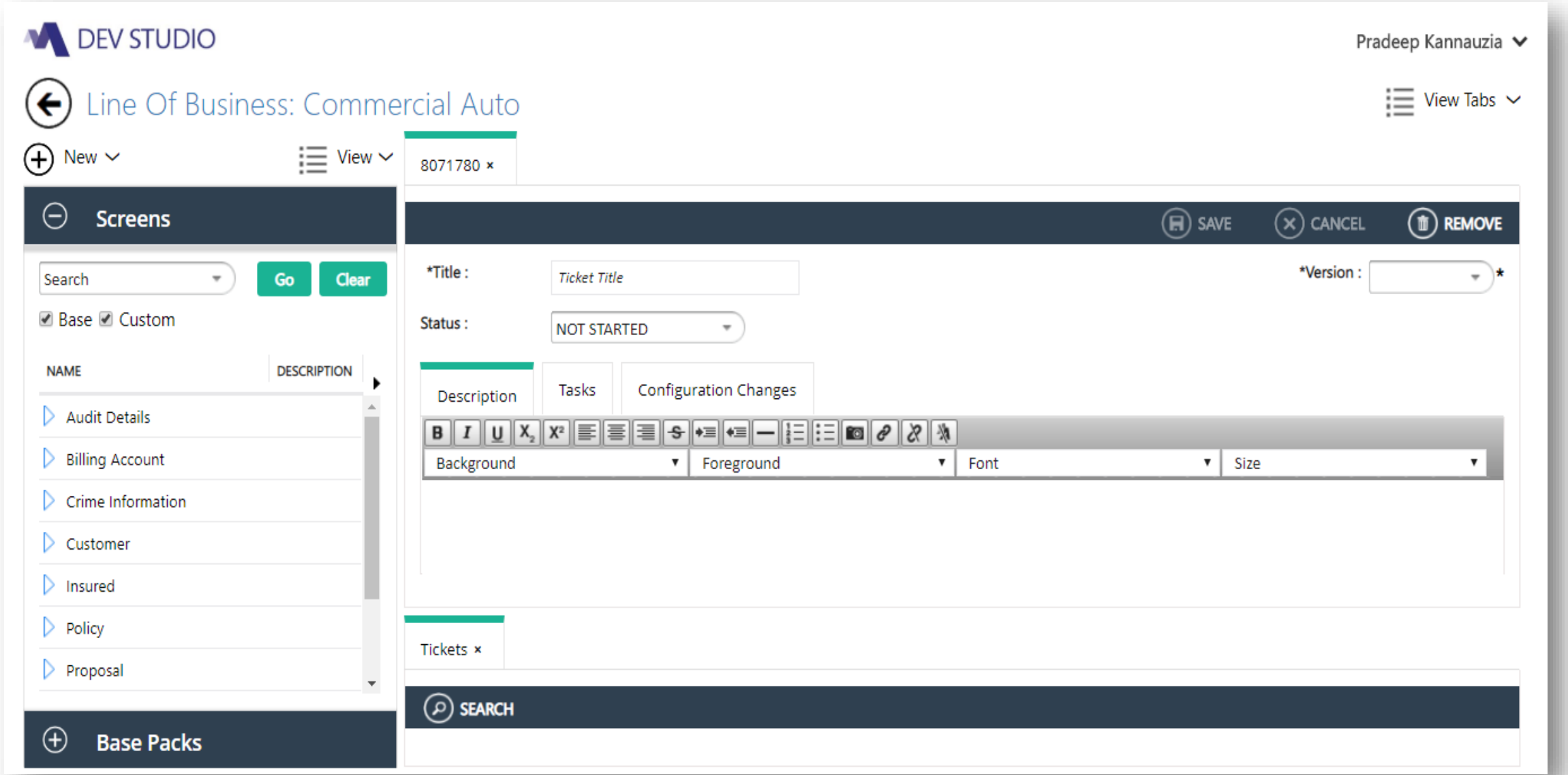
- Ticket plugin is used to create new tickets.
- To perform any add/modify/delete operation through Dev-studio we need a ticket.
- Each tickets are mapped to release version number. So whatever changes we perform using tickets those changes are mapped to specified release version number.
- Steps to create new Ticket:
  - ✓ Enter ticket Title.
  - ✓ Select Release Version
  - ✓ Select status for the issue.
  - ✓ Click Save.

Ticket Plugin



MAJESCO

Ticket Plugin



MAJESCO

Screen Plugin Overview

- Screen Plugin of Dev Studio provides the functionality required to manage and configure various Screen (Object) and its fields, which are required throughout the lifecycle of an insurance policy.
- A screen plugin includes entities, screens, objects and fields that can be configured by using tools and properties available in the screen editors.
- In Dev Studio every individual screen is called as an object.
- All objects(Screens) will be visible on expanding Screens plugin.
- Every object(Screen) has some properties which is available on properties tab.
- Every object can have multiple fields on it. And every field has its own properties which is available on properties tab.
- Toolbox tab is used to add new tools(fields/controls/objects) on object.

Screen Plugin and properties

- **Keywords used throughout Screen Plugins**
- **Entities:** Entities are the Screen (Object) that does not have any parent associated with it. These screens are root screen, only child screen can be associated with them. E.g. Policy, Insured, Proposal etc.
- **Screen:** A screen is an object that is created as a child of an entity (building block for screen). Screen can be an object or combination of more than one object. Screen can be a child or parent of another screen. For Example: "General Liability", "Liquor Liability“ etc.

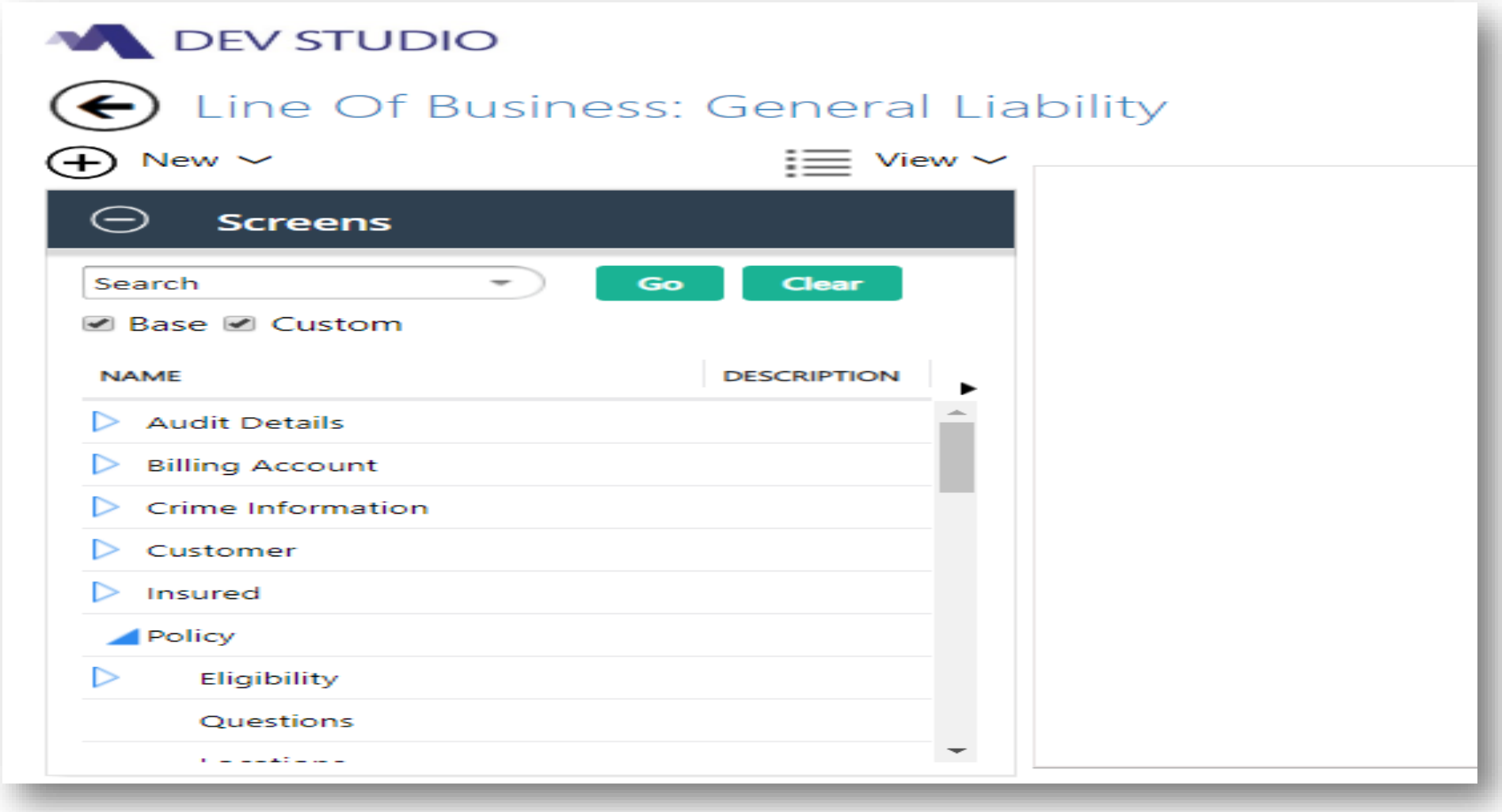
  "Liquor Liability" object is child of "General Liability" object and the parent of "Classifications" object.
- **Object:** Object is made up of a collection of different kinds of fields. For Example: Location, Address, Name, etc.
- **Field:** Fields are the lowest attribute that has been defined on user interface (Object) to capture the policy data that may or may not be visible on screen/objects. For Example: "Insured Name", "Policy Number", "Composite Rating" etc.

Screen Plugin and properties

- **Common Objects :** Common Objects are reusable objects that are created to be used across different Products/Line of Business. Fields that are created under these common objects are also common across the Products/Line of Business. Any object that is created above the Line of business is treated as a common object. For Example: On Policy Entity, Locations object is a common object, where Location address is child of the Locations object. Thus all the objects under the Policy entity are common objects. Which can be used in any product or line of business without recreating them
- **Line of Business/Product Objects :** Line of Business/Product is an object created for a specific Product/Line of Business and cannot be used in other Line of Business.
- For Example: Under General Liability, Premium & Production are general liability specific objects.

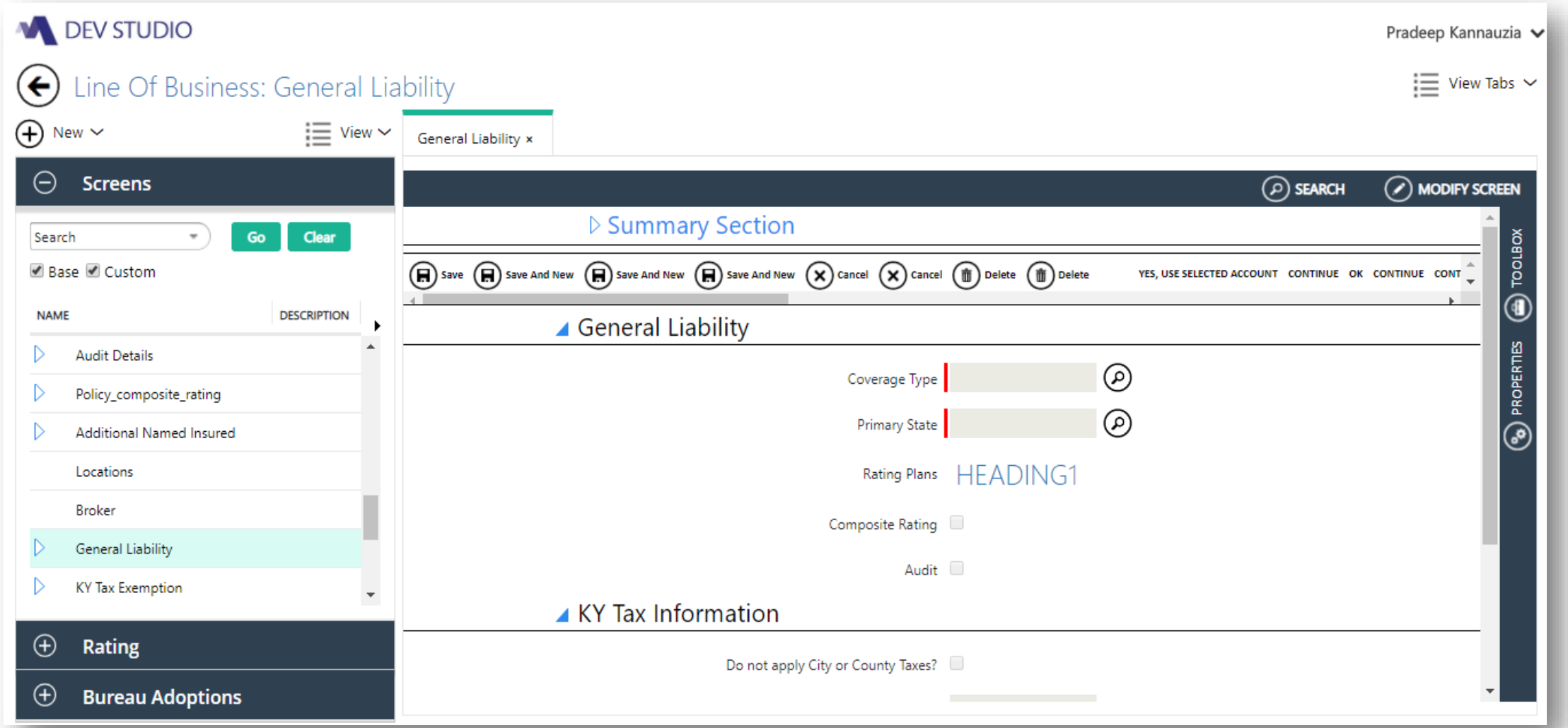
Screen Plugin and properties

- Object Panel



Screen Plugin and properties

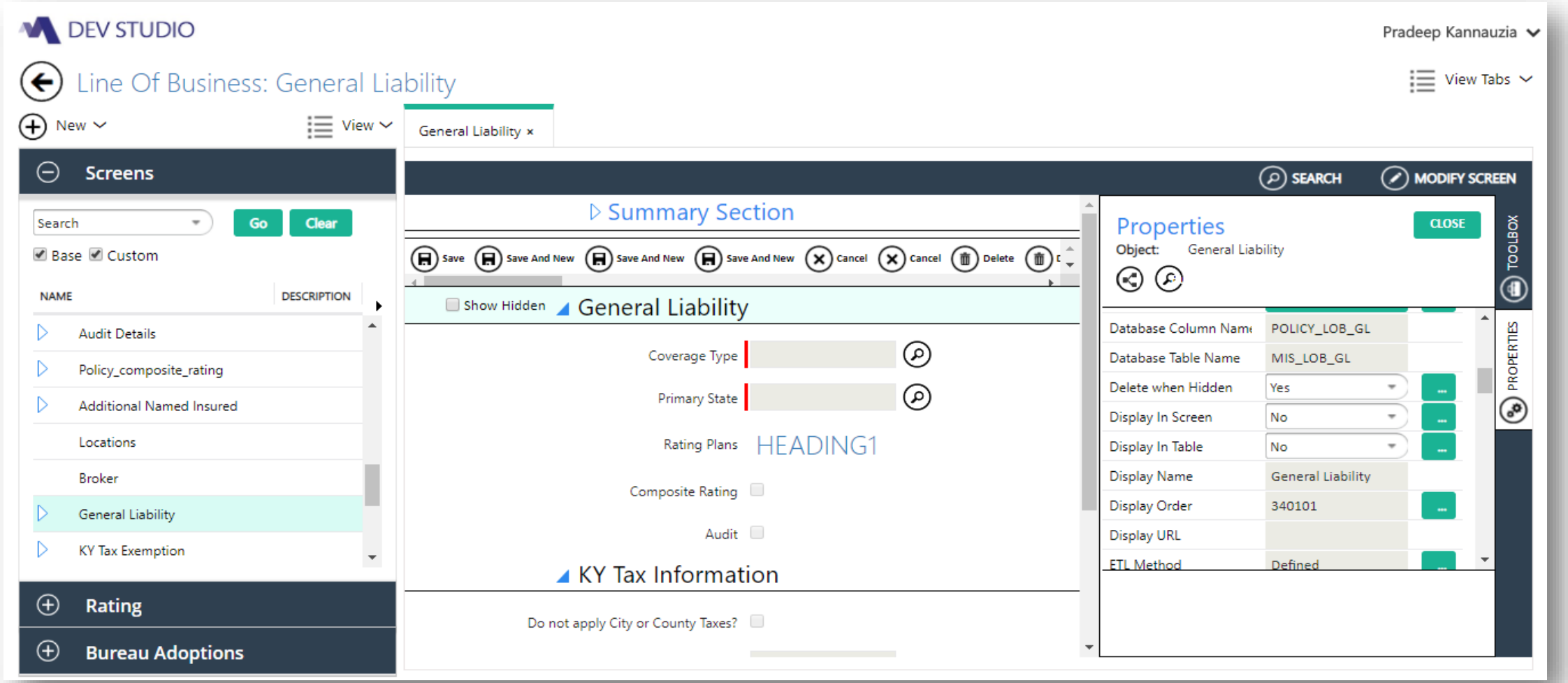
- Object Editor



MAJESCO

Screen Plugin and properties

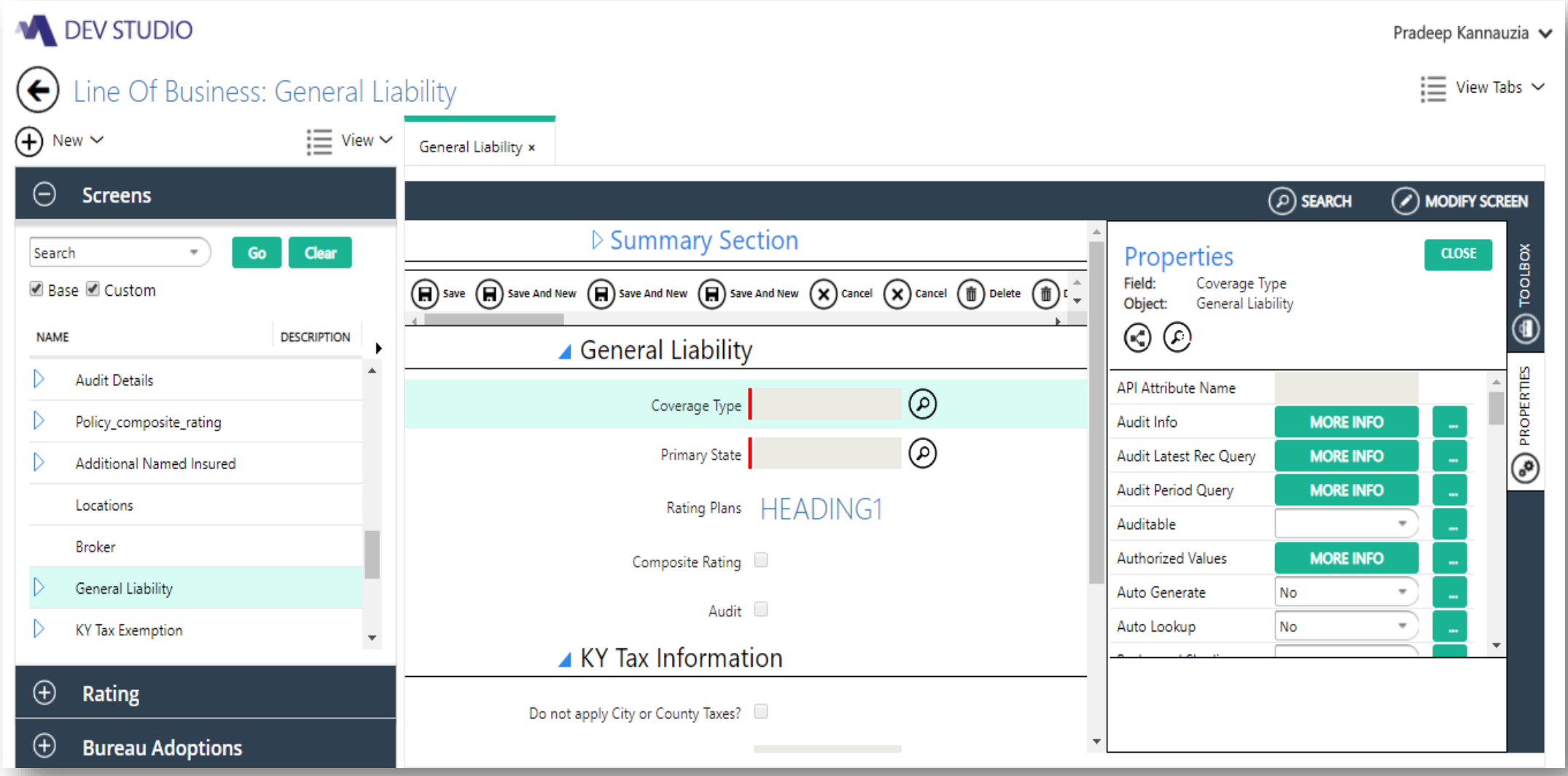
- Object Properties Panel



MAJESCO

Screen Plugin and properties

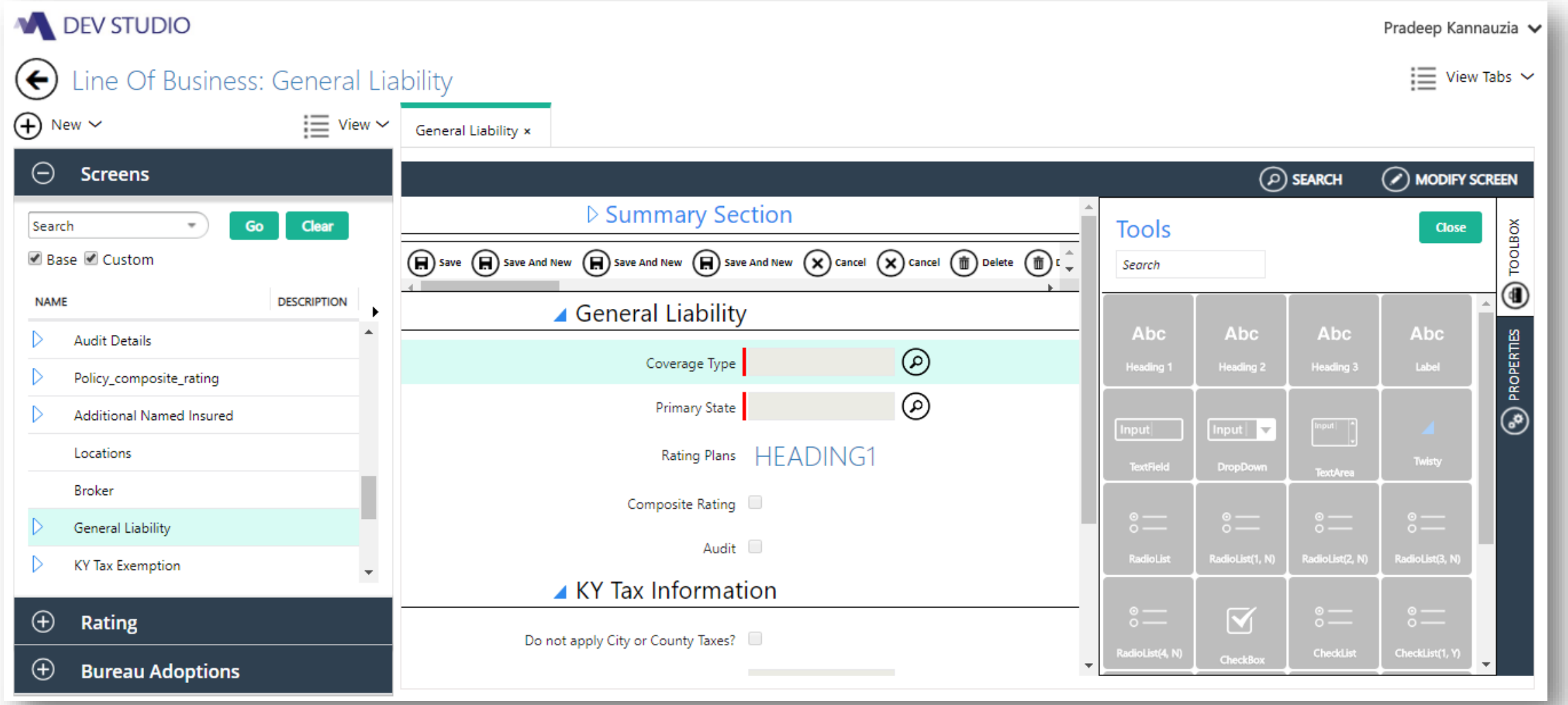
- Object Field Properties Panel



MAJESCO

Screen Plugin and properties

- Toolbox Panel



MAJESCO

Create a new Object

- An Object is used to create a screen in Dev Studio. Each object, once created, generates a table in the database by the system automatically.
- A new window with the Object Section is opened to enter this new object and to define the name of the table, along with the other attributes described in next slides.
- Once the object is created, every field created in the object corresponds to a column in the object table.
- These tables and columns for an object are used to store the data for the object and fields during runtime.
- Every object created is linked with the Parent Object through Foreign Key (which gets created when you drag the Object toolbar). The Foreign key shows the Object on the UI.

Create a new Object

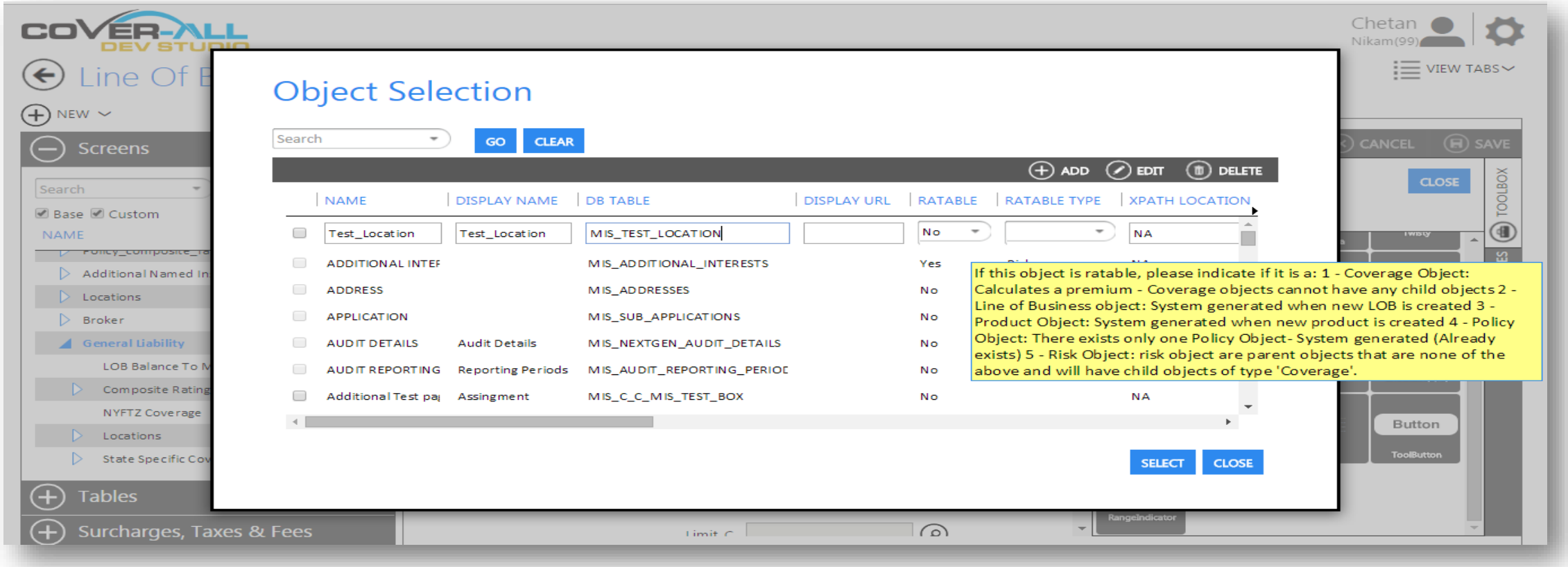
- When the Object is reused, only the foreign key changes while the entire screen or objects remains unchanged.
- Object can be one of two types:

  1. **Non Ratable Object:** It’s a normal object and ratable property set to “No” and ratable type properties of these object is null. These objects are not use for the rating or premium calculation.

  2. **Ratable Object:** It’s an object for which ratable properties set to "YES" and Ratable type properties is defined to "COVERAGE". These objects are participate in rating and used to calculate the coverage or premium of policy. Ratable object's parent must be a ratable object.

Screen Plugin and properties

- **Creating a new Object :**
  - Object tool is used to create a screen in Dev studio. This object represents table in database. We can use the predefine object or create the new object.
  - TOOLBOX-Object selection



Screen Plugin and properties

- Toolbox Editor : Toolbox Editor is used to create the object or field using drag and drop function.

- Toolbox Editor provides below Tool
  - OBJECT
  - HEADING1, HEADING 2, HEADING 3, LABEL
  - TEXTFIELD, TEXTAREA
  - DROPDOWN, CHECKBOX, TOOLBUTTON, TWISTY
  - RADIOLIST, CHECKLIST, RANGEINDICATOR

Toolbox Editor

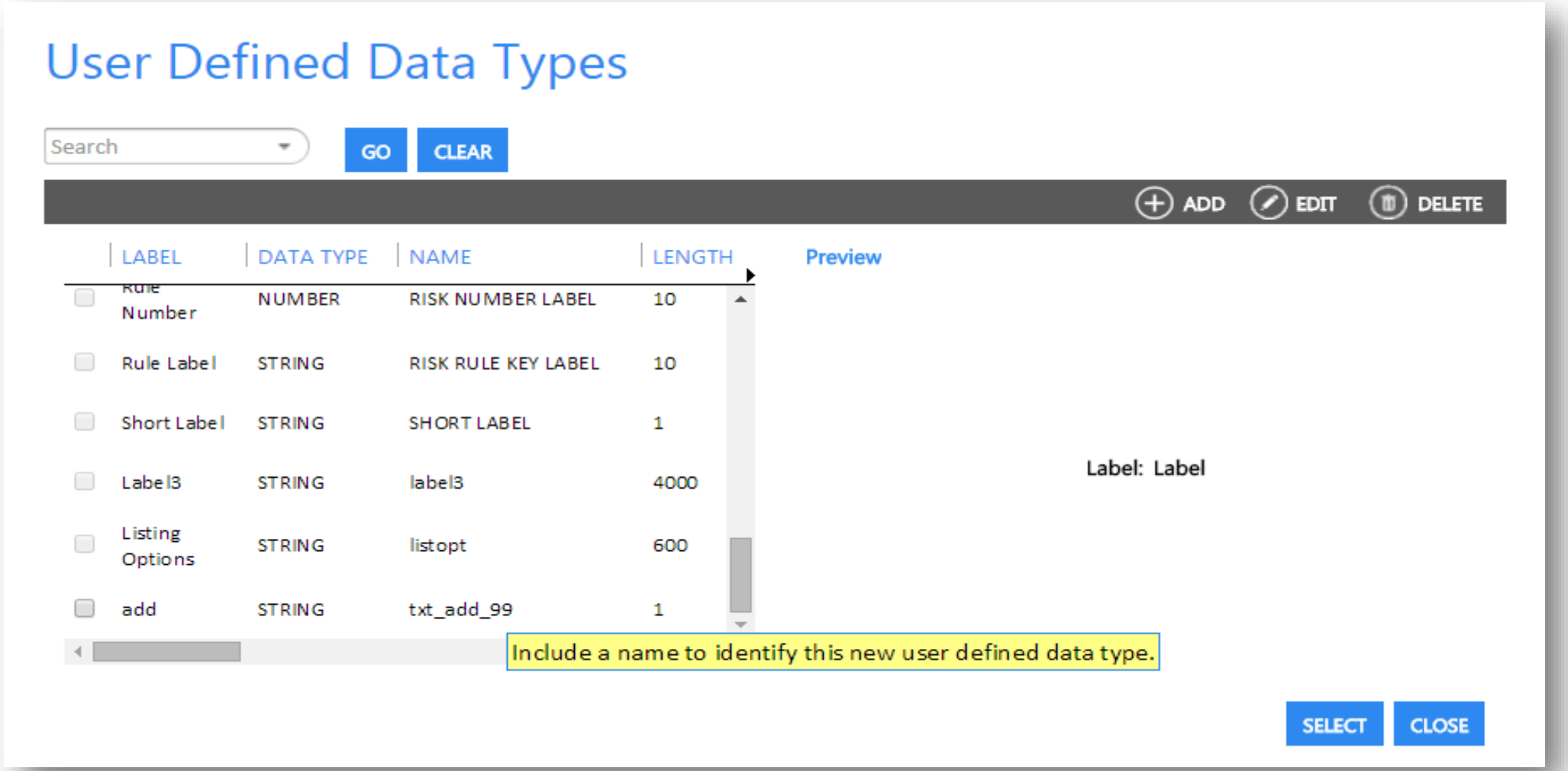
- Confluence Link:
- https://confluence.majesco.com/display/MPCL/Screens+Plugin#ScreensPlugin-ToolboxEditor

Screen Plugin and properties

- **Data Types For Tools(Primitives) :**
  - Any tool that is used from the Toolbox has to be associated with a data type. Data Type is nothing but the definition of type of data that is going to be used to store in a field. For Example: “Location Number” field(Text Field) would be associated with a data type of “Number”
  - Similarly, every tool in the toolbox editor is associated with some data types. Some are defined by the system and some are user defined.
  - While creating a new Object/field user can use existing data type or can create new data type. Preferred is to use existing one.

Datatypes

Screen Plugin and properties



MAJESCO

Properties Editor

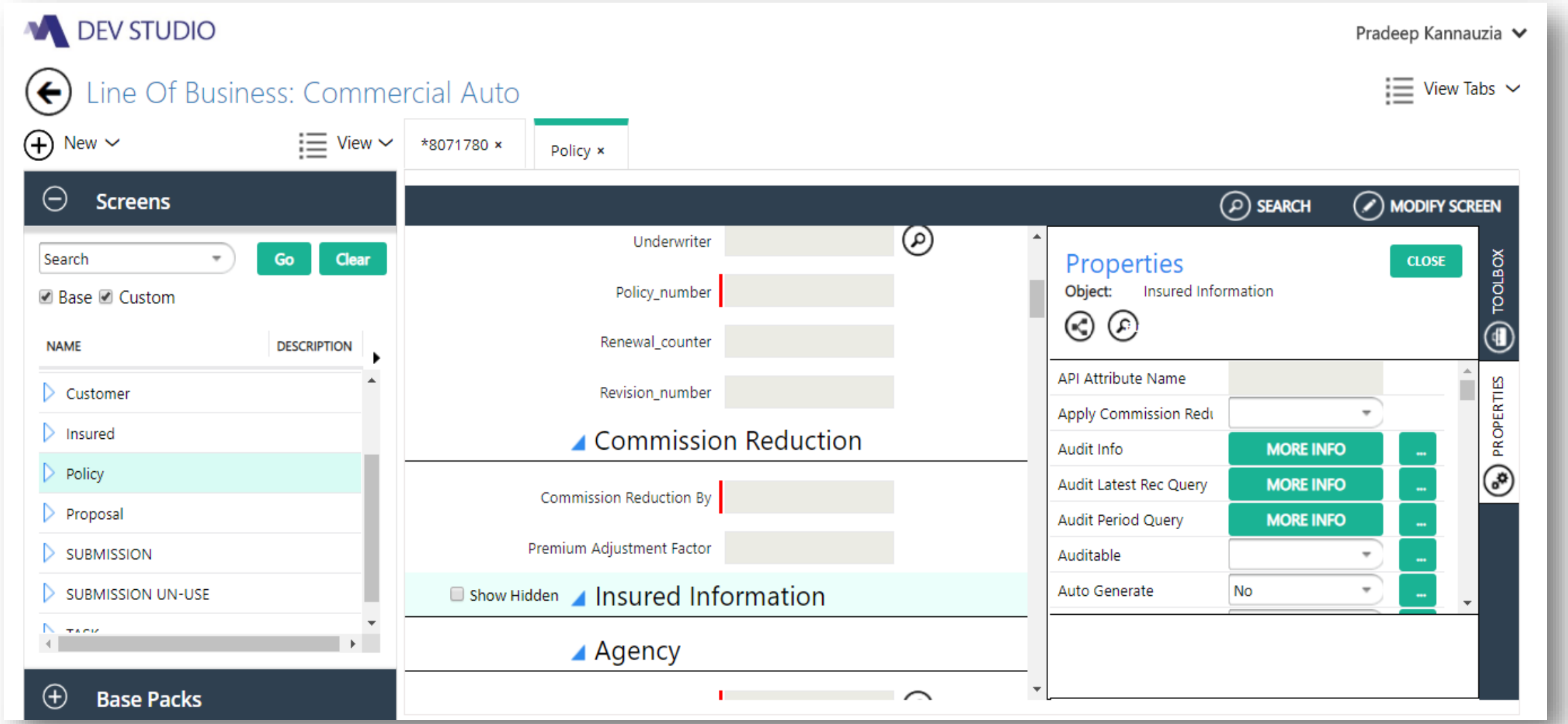
- The Properties Editor is used to define the properties of any object or field that is created. This editor can be accessed by navigating to the Properties icon on the Right Panel. Dev Studio groups the properties based on their use and functionality.
- To add/edit the properties of object/field, the user has to select a field and then click on the properties tab on the Editor panel on the right. Here the user can view and set different kind of properties and their value for selected object/field.On the properties panel the field selected and object the field belongs to is shown.
- Some properties are only available either for field or object, while other properties are common for both. Some properties are used as Boolean value of Yes or No, while some require adding of Rules to set the property for the Object/Field.

Properties Editor

Composite Objects

- An object can compose another object within it.
- This can be configured to reuse existing objects.
- When the Object is reused, only the foreign key changes while the entire screen or objects remains unchanged.
- For e.g. An object by name Insured can have an object Address composed within it. And also an object Producer can have Address object composed within it. This is because both Insured and Producer screens need to have address associated with it. Hence there can be an independent Address object which can be used on multiple objects.
- This is referred as Foreign Key relation.

Composite Objects



MAJESCO

Best Practices

- While adding a field in existing object that has child object, collapse the child object to prevent the new field from being added to child object accidently.
- If creating a new object, ensure ***Fk_Column_Name/DB_column_name*** length does not exceed 26 characters and it should be always in uppercase without spaces.
- Do not use oracle keyword for Column and table names.
- Save changes/Screen regularly after adding 1 or 2 customization's, avoid saving changes after adding 8-10 customization at a time, at times you may lose the customization you have added.
- You can always delete fields/rules added in the same release without any issue. After install we cannot delete anything so mark those as hidden yes.
- If creating a new coverage code, ensure its length does not exceed 15 characters.
- Need to be careful while adding first field for a non-ratable object to verify its added as a part of the respective object only.
- Column names or table names should follow naming convention.

  Coverage object: MIS_COV_<LOBCODE>_<TABLENAME>

  Non coverage object: MIS_<LOB_CODE>_<TABLENAME>

MAJESCO

THANK YOU!