

Get started

Get permission as a Cauzali model designer.

Ask your Salesforce admin to assign you to the permission set named Cauzali Editor.

Tutorials:

- 1) Create your first model
- 2) Working with fields
- 3) Layout
- 4) Select Fields
- 5) Navigation
- 6) Mapping to standard and custom objects
- 7) Different outcomes
- 8) Images
- 9) Use Cauzali on lightning pages, in communities, in flows or in your lightning components
- 10) Scoring
- 11) Hide and disable fields

Tutorial 1: Create your first model

1. From App Launcher  at the top of the page select **Cauzali**.
2. Click on Questionnaires tab.
3. Click on **New** button.
4. Use “MyFirstModel” as Name and press **Save** button.

Now you are in CauZali Designer. To the left, you can see Designer and to the right, there is Preview version of your model.

5. In Designer choose Flow tab and press **Add** button.

Now you can see a “Node”, which is the main element in our decision tree.

Node



Each node has a gray area indicating which kind of node it is. Use the gray area when you use drag and drop. Each node consists of a Type and a Name field and some action buttons. Some nodes have more fields depending on their type.

If you follow all the tutorials you will get familiar with all different types of Nodes.

The default node type is Question. In tutorial “Node type” you will learn more about node types.

6. Press **Content** in the node.
7. Type “MyPage 1” in the Title field and “This is my first page” in the Content field. Press **OK** and **Save**.

Already now can you test your model.

8. In Preview section press **Restart**.

Now you can see your simple model in the Preview part of the screen.

Let's add more logic.

9. In Question field of your node write "Do you have children?"

10. In the node press **New** twice.

You get two new nodes with default names. When trees grow it's recommended to use more informative names. But leave these names for now.

11. In the node named "node-2" write "Yes" in Answer field.

12. Press **Content**, write "Children" in Title field and press **OK**.

13. In the node named "node-3" write "No" in Answer field.

14. Press **Content**, write "No Children" in Title field and press **OK**.

Test your model again. Remember to save the model and press **Restart**.

We continue with a question about housing, and come back to questions about children in a couple of steps.

15. Press **Add** in Designer and you will see a new node called "node-4". Press **Content**, type "Housing" in Title field and press **OK**.

16. From the "node-3" (customer who doesn't have children) we want to jump to the new node about Housing (node-4). Change the Type of the node-3 to "Next" (which means going to the next node of the first level).

Save your model. Now test what happens when you choose "No" to the question about children.

Suppose we want to know how many children our customer has and get more specific data if they have children under 18 years.

17. In node-2 (customer who has children) write "Any kids under 18?" in Question field.

18. Click **New** twice in the node.
19. Again write “Yes” in Answer field in the first node (node-5) and “No” in Answer field in the second node (node-6).
20. Change Type field in the node-6 to Next.

To find out how many children the customer has we will introduce “Fields” in our model.

21. Press **Content** in node-2, choose the “Field Usage” tab and press **Add**.
22. In the new line choose Edit in Usage field, type “numKids” in Name field, “Number of kids” in Label field and choose “Integer” in Type field.

To the right you can see how the page will look like.

23. Press **OK**.

Save model and test. More about fields in the Fields tutorial.

24. In node-5 write “Does any of the kids have a driver license?” in Question field.
25. Add two Answer nodes to the question (press **New** 2 times in the node).
26. In the first Answer node (7) write “Some of them” in Answer field and change node Type to Next.
27. In the second Answer node (8) write “None of them” in Answer field and change node Type to Next.
28. Press **Copy** in the node-8.
29. Change Answer field in the new node(10) to “All of them”.

The copy function is just nice to have in this particular case, but as it copies everything from the node including subnodes it can be very useful in more complex cases.

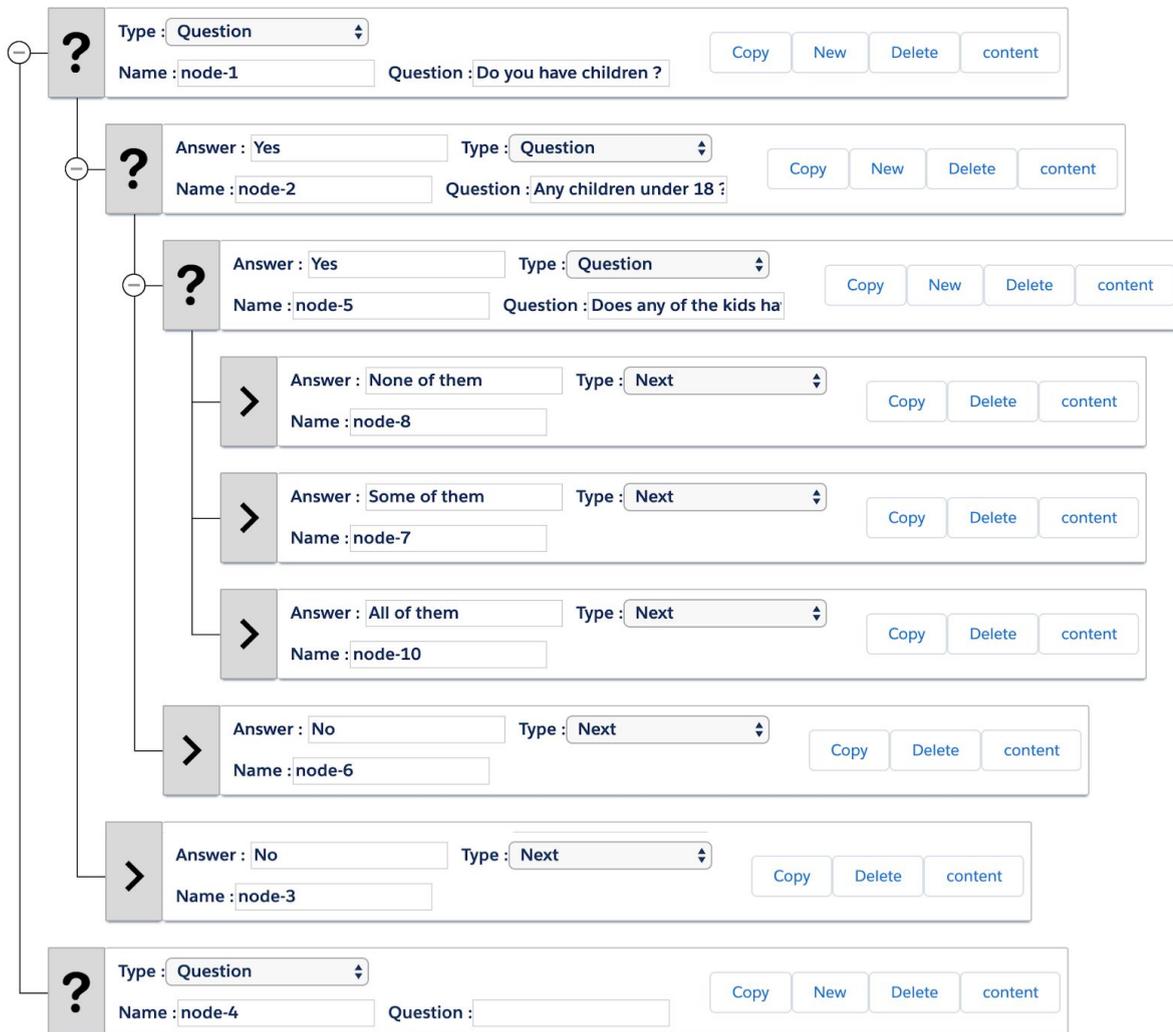
Test your model again (remember to Save before testing).

Hey! Doesn't the order “Some, None, All” sound weird? Won't it be more natural with “None, Some, All”?

30. Place the cursor in the left gray field of the node-7 and drag&drop it under node-8.

Save and Test your model again.

Your model will look like this if you have made the tutorial correct.



This was the basic tutorial. Try next tutorials to learn more about input fields.

Tutorial 2: Working with Fields

Fields are used to get input from the user and save these inputs in Salesforce record. In this tutorial, we are not going to work with Salesforce records, but only Fields in CauZali. In tutorial “Working with Objects” you can learn how to populate Salesforce records with Fields from Cauzali.

Let’s start the tutorial making some Fields.

1. From App Launcher  at the top of the page, select Cauzali.
2. Click on Questionnaires tab.
3. Click on **New** button.
4. Use “MyFieldModel” as Name and press **Save**.
5. In Designer press **Add** and change Type to “Page”.
6. In the node press **Copy**.
7. In the first node change Title to “First page” (press **Content**, fill Title field in and press **OK**).
8. Still in the first node press **New**.
9. In the new node change Type to Next and write “Forward” in Answer field.
10. In the last node change Title to “Last page”.

Save and test your model. See if you can get from the first page to the last page.

Now we are going to create some Fields.

11. In Designer select “Fields & mapping” tab.

Suppose we want the user to fill in name, age and email.

12. Press **Add** button three times and you will see 3 new lines.
13. In the first line write “customerName” in Name field, “Your name” in Label field and in Type field choose “string”.
14. In the second line write “customerAge” in Name field, “Age” in Label field and in Type field choose “integer”.
15. In the third line write “customerEmail” in Name field, “E-mail” in Label field and in Type field choose “email”.

Thus we have defined three fields. Now it’s time to place them on Pages.

16. In Designer select Flow tab.
17. In the first node press **Content**.
18. Choose “Field Usage” tab.

Here you can see three fields you’ve made in step 11-15. You can choose to expose each field on an individual page. You decide if the user can edit or just see the field.

19. Select “Edit” in Usage field in each line. Press **OK**.
20. In the last node of your model repeat step 17-18. Select “Show” in Usage field in each line. Press **OK**.

Save your model and test the model in Preview window to the right (remember to Restart).

Enter your data on the first page and you will see the same data on the last page.

Most Field types allow setting minimum value, maximum value, default value, requirements etc. Let's make some restrictions. For example, age may not be less than 18 or greater than 80. We would also like to stop the user from going forward with the questionnaire without filling in the email.

21. In Designer select “Fields & mapping” tab.

22. In the row with the CustomerAge press Action button to the right just after Delete button. We call it Settings button.
23. Under Validation set 18 in Min field, and set 80 in Max field. Press **OK**.
24. Press Setting button on the email field row.
25. Check the “Required” button under Validation. Press **OK**.

Save and test your model.

When you test your model it's always easy to see the value of all Fields. Just select State tab in Preview window.

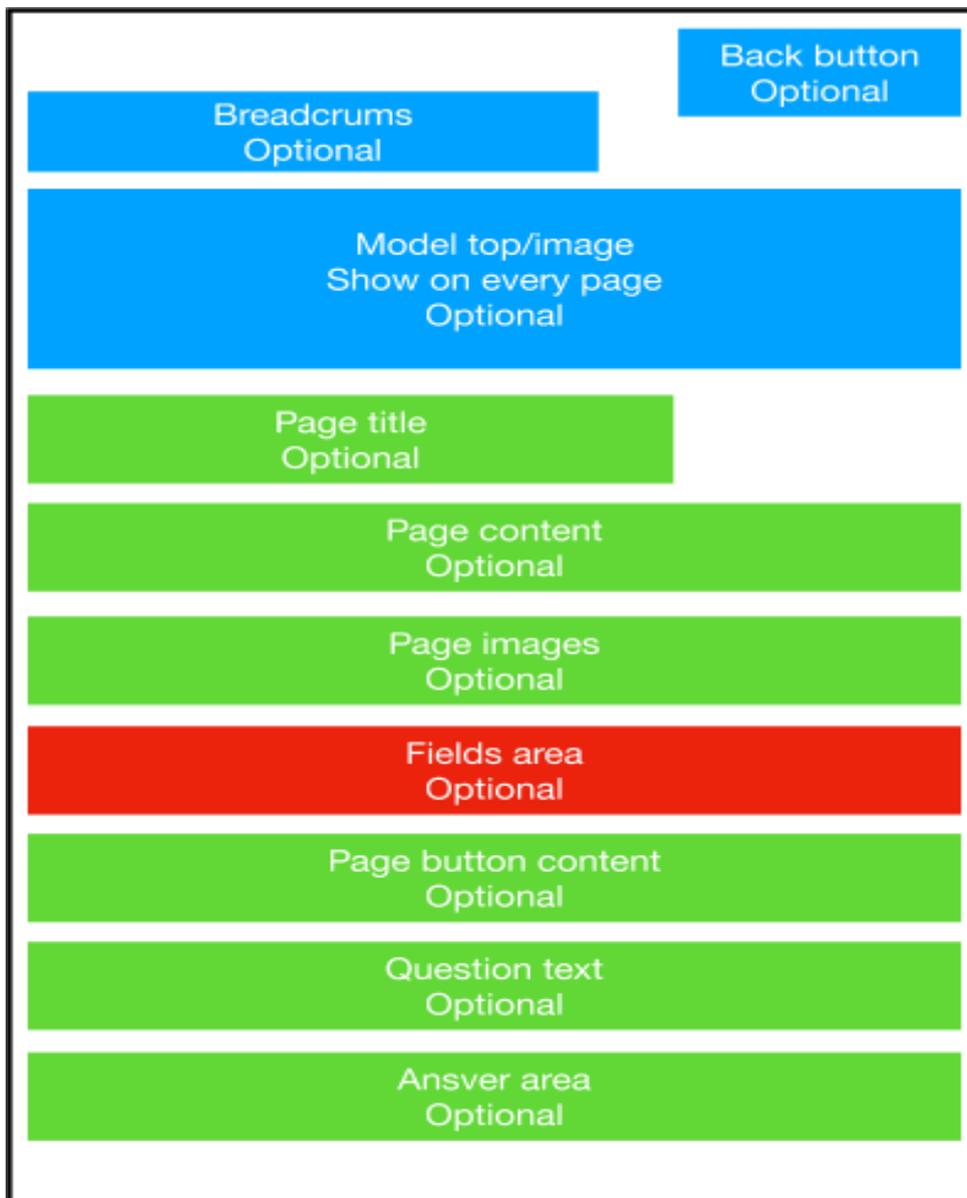
In later tutorials, you will learn about more possibilities with Fields such as Hide and Disable functions, Default values, passwords etc.

Now feel free to play with some other Types of Fields. Though be aware of types: “group”, “description”, “image” and “select”. They are used for more complex models and will be described in detail later.

Tutorial 3: Layout

As mentioned in tutorial 1 we have many different types of nodes. Some of them represent a form with different sorts of content. In these forms, you have a lot of possibilities to make a nice layout.

Basically, there are 3 layers of layout settings. One for the whole model, one for the page and one for the field area on the page. In the following figure, the model layer is blue, the page layer is green and the field layer is red.



As you can see everything is optional and will disappear if it isn't used. Some of the components on the page layer default from settings in the model layer. For example, in the model layer, you can set whether you want answers to be shown as buttons or a list.

CauZali uses Salesforce Lightning Design System to be able to style it just like other lightning components. This is especially beneficial when you place CauZali on a community page for your customer or partner.

Let's see how it works.

1. From App Launcher  at the top of the page, select Cauzali.
2. Click on Questionnaires tab.
3. Click on **New** button.
4. Use "MyLayoutModel" as Name and press **Save**.
5. In Designer press **Add**. By default, it's a Question node.
6. In Question field write "Do you prefer red or yellow ?".
7. In the node press **New** twice.
8. In the first new node write "Red" in Answer field and set Type to "Next".
9. In the second Answer node write "Yellow" in Answer field and set Type to "Next".
10. Just to have a model without dead ends press **Add**, change node type to "page" and give it a title.
11. **Save** and test your model.
12. Now we want to add a back button and breadcrumbs. As you can see in our figure this can be done on the model level.
13. Choose "Setup" tab, and set a checkmark in both "Show Breadcrumbs" and "Show BackButton".

Save and test your model. Choose an answer to see **Back** button and Breadcrumbs. "Preview/Flow/State/Result" tabs are only shown in preview mode.

Now let's work with the page layer.

14. In the first node press **Content**. On the "Text" tab (default) fill in your own text in Title, Content and Content Bottom. To the right, you will see your changes immediately.
15. Choose "Layout" tab and change layout from None to Buttons.

Now let's move to the field level.

16. As learned in tutorial 2 make following 4 fields.

Name	Label	Type
I1	Number_1	Integer
I2	Number_2	Integer
S1	Text_1	String
S2	Text_2	String

17. Press **Content** on the first node and select "Field Usage" tab.
18. Make all 4 fields editable (tutorial 2 - step 19).
19. Press **Add**. In the new field change usage to "Show" and type to "group".
20. Click on Radio button in this field line and press arrow-up until the line is on top.
21. Change Cols to 2 and see what happens in the right preview window.
22. Add some text in the label field and see changes to the right.
23. Make a new group line and place it between the i2 and the s1 field and set cols to "1" (you can just press the + button on the i2 field line).

In the group line, you've just made press the + button and set Usage to "show" and Type to "description".
24. Press Settings button and type some text.

Save and test your model.

If you want to place images in your model both on the model layer, page layers or between fields see Tutorial 8 - Images.

Tutorial 4: “Select” fields

“Select” fields are a special field type in Cauzali that have some similarities with Picklist fields in Salesforce. In Cauzali we call lists of values “Item types”. Each value on the list is called “Item”. Item types are global for all your models.

Let’s make a model where we ask the user which kind of transportation the customer prefers: Car, Bus or Train.

In this Salesforce solution we will use abbreviations for transportation like “ca”, “bu” and “tr”.

Now we will create Item types.

1. In CauZali app, choose “Item Types” and press **New**.
2. In the Name field write “Transportation” and press **Save**.
3. Choose “Related” tab and press **New**.
4. Let's call the Item “T1” in the Name field. This is the only place you will see the name.
5. Write “Car” in the text field (This is what the user can see).
6. Write “ca” in the value field.
7. Leave the rest and press **Save & New**.
8. Make 2 new Items:

Name	text	value
T2	Bus	bu
T3	Train	tr

Sort field is used to change the order of items when the user sees the selection list. And Image field is a really cool feature that you will learn about in tutorial 8.

Now we have defined Item Type as Transportation. Let's use it in "MyFieldModel" from tutorial 2.

9. Open "MyFieldModel".
10. Press **Content** in the first node.
11. Choose "Field Usage" tab and press **Add**.
12. Make the new field editable, set name ="trans", label="Preferred transportation", and type="select".
13. Press Setting button for the field.
14. Go to the bottom of the page and select your "Transportation" item in the itemType field. Don't change the layout field.
15. Press **OK,OK** and **Save**.

Test your model. Use the "State" tab in the Preview window to see the values of your fields.

Be aware that the field is first set when you choose an item. If you want to set a default value to the first item, you have to set default in the field. In this case set default value for the field "trans" to "ca". Another solution is to create Item with no text and no value and put it on the top of the Item list.

Now go back to the setting in the field and change layout type to "Horizontal". Take a look at the Preview. Now change it to "Radios". You can adjust your layout to your preferences.

Wait with images until tutorial 8.

Tutorial 5: Navigation

Until now to navigate from one node to another we have used “Next” as node type. For more specific or complex situations we introduce GoTo and Routing node types. GoTo navigates to a specific node, while Routing navigation depends on simple logic.

Be aware!

Cauzali supports only forward navigation. Designer is not always able to test for backward navigations.

1. From the App Launcher  at the top of the page, select Cauzali.
2. Click Questionnaires tab.
3. Click **New** button.
4. Use “MyNaviModel” as Name and press **Save** button.
5. In Designer, press **Add** five times.
6. In each node change the title to “Page 1”, “Page 2” (remember, you do in **Content**)
7. In the first node write “Where do you want to go ?” in Question field and leave the type as “Question”.
8. Still in the first node press **New** twice.
9. In the first new Answer node write “Page 2” in Answer field and change node type to “Next”.
10. In the second new Answer node write “Page 3” in Answer field and change node type to “Goto” and choose node-3 in Select field.
11. To get a model you can test, under node-2 create a new node with type “Next” and write “Continue” in Answer field.

Save and test this typical use-case where Goto helps you to bypass some parts of your model.

Now we will try the Routing node type.

12. In node-3 change node type to Page. Add a field to the page with type “integer”, name “age”, label “How old is the customer:” (remember to make the field editable).
13. Press **New** in node-3 and write “Next” in Answer field in the new node. Change the node type to “Routing” and press **Content**.
14. Press **Add** to make the first rule. In the row choose Field “age”, Operant “Less than”, Value 21 and GoTo “node-4”.
15. Press **Add** to make the second rule. In the row choose Field “age”, Operant “Greater than”, Value 20 and GoTo “node-5”.

Be aware!

The first true rule will be used and other true rules will be ignored. The model goes into a dead-end if none of the rules is true.

Save and test your model.

Tutorial 6: Mapping to standard and custom objects

The real benefit of Cauzali is the ability to create and update Salesforce records.

In this tutorial we will make the most simple type of mapping. Later on, we will show more mapping possibilities.

Be aware!

Cauzali respects all security settings. It can create and update records only if the end-user has permissions on all levels.

This Tutorial isn't bulletproof if your org is customized with special required fields on leads or special record types.

We start to create a Lead.

1. From App Launcher  at the top of the page, select **Cauzali**
2. Click on Questionnaires tab.
3. Press **New**.
4. Use "MyCreateLeadModel" as Name and press **Save**.
5. Make three fields like this (hint: use the "Fields and mapping" tab):

Name	Label	Type
Custname	Customer name	String
No_emp	Number of employees	Integer
Email	Email	Email

6. Add 2 nodes to your model.
7. Change type in both nodes to Page.
8. Create a new node under node-1. Change it to Next node.
9. Press **Content** in node-1, change the title to “Get information” and choose “Field Usage” tab. Change Usage in all fields to “edit”.
10. Press **Content** in node-2, change the title to “Show information” and choose “Field Usage” tab. Change Usage in all fields to “show”.
11. Save and test.

Now we have gathered some information and want to create a lead. In “real world” it’s always a good idea to complete your model as much as possible before you start mapping to objects.

12. Go to “Fields and mapping” tab and in “Map to object” field choose lead.
13. In the Custname field choose “Last Name” in Mapping field.
14. In the No_emp field choose “Employees” in Mapping field.
15. In the Email field choose “Email” in Mapping field.

Now we have specified mapping. To activate it we have to change the last node’s type to “Finish”. This will mean that the user has reached the

end and if you have specified mapping, it will be executed at this stage. You can make Finish nodes wherever you want.

16. Change node type in node-2 to Finish.

17. **Save** and test your model - go to your list of leads and see if the lead is created.

If you don't find your lead, it's probably because there are some required fields in your lead object. Add these fields in your model and try again. Remember to check Required in your model or give them a default value (you can do it in settings on the field).

The lead is by default assigned to the end-user, but in Setup tab of your model you can check "Use assignment rules for mapping to case and lead for this questionnaire" and the lead will be assigned in accordance with your assignment rules.

In tutorial 9 you will see how we update an record and create records with relation to other records.

Tutorial 7: Different outcomes

In tutorial 6 we worked with the most common outcome (create and change Salesforce records). This was done with the help of Finish node.

Another outcome is “New Questionnaire” where one Cauzali model initiates another model. Think of a model that classifies problems and specific models that help to solve each kind of these problems.

The last outcome is “URL”. It is used to open a webpage in a new tab or in the same tab. If it’s in the same tab, you’ll leave the model.

Let’s try both “New Questionnaire” and “URL” in one model.

1. From App Launcher  at the top of the page, select Cauzali.
2. Click on Questionnaires tab.
3. Press **New**.
4. Use “MyOutcomeModel” as Name and press **Save**.
5. Press **Add** to make a question node. Write “How do you want to get more info?” in Question field.
6. Press **New** twice.
7. In the first answer node write “Self-study” in Answer field and change Type to “URL”.
8. To set up the url press Content and write “<http://www.salesforce.com>” in Redirect URL field. Leave Redirect Target as “New tab”

If you skip the “http://” the url will be relative and let you navigate to objects, lists, articles and more in Salesforce. For example,

“/lightning/o/Lead/list?filterName=Recent” will open your list of recently viewed leads.

9. In the second answer node write “Let sales rep contact me” in the Answer field and change Type to “New Questionnaire”.
10. Press Content and write “MyCreateLeadModel” in New questionnaire field. Be aware, it is important that you write this name exactly as you did it in the previous model in tutorial 2.

Before you try your model you have to make MyCreateLeadModel active. Otherwise, you will get an error when you try the model.

11. **Save** your model.
12. Open MyCreateLeadModel, set it Active in the top of the page and Save.
13. Open MyOutcomeModel and try two different outcomes.

Tutorial 8: Images

Images are often what make a solution very nice, but often complicated for the designer. CauZali has its own image library to do it simply. In this tutorial, we will first upload some images to the library.

Images can be used 3 places in CauZali: as a header for your model, on a page in your model or use images in a select field.

Start finding 4 images. 3 images of the same size and relative low resolutions. Ex. 400*400 pixel. 1 more wide image ex. 1200*400 pixel.

1. From the App Launcher  at the top of the page, select **Cauzali**.
2. Click the **Image manager** tab.
3. Click the **New** button.
4. Give your first image a name, and upload the image.
5. Press **OK**
6. Repeat 3-4 for the 3 other images.

Now you can see the 4 images ready to be used in Cauzali models.

Let's work on the model from Tutorial 4 where we implemented a select field.

Let's start with the top image for your model. To place a top image we use a configuration which can be used in many models.

7. From the App Launcher  at the top of the page, select **Cauzali**.
8. Click the Configurations tab.

9. Click the **New** button.
10. Write “MyConfig” in name and press **Save**.
11. Click on the **select** button under Top image.
12. Find the wide image you just uploaded, and click on the checkmark.
13. Press **Save** to save your configuration.
14. Open the MyFieldModel from tutorial 4.
15. Choose the Setup tab.
16. In the “Select config” choose “MyConfig”
17. **Save and test** your model - now with a header image.

Try now to insert images on a pages nodes. Use the Image tab on the page nodes content page.

One really nice feature is images combined with select fields. Let try.

18. Go to you Item types from tutorial 4.
19. Choose the related tab.
20. Open each item and select an image with the selection to the right. Use the 3 images with the same size. Press **Save** when you have selected an item.
21. Now open your MyFieldModel and find the Select field you added to the first model.
22. In the field setup change the layout to “Image” and set Columns accross = 3 to indicate that you want 3 image on each row.
23. **Save and test** your model.

Tutorial 9: Use Cauzali on lightning pages, in communities, in flows or in your lightning components

It's possible to implement your CauZali model nearly everywhere in Salesforce where you can place a Lightning Component.

Typical you will place the component on a record page with Lightning App Builder. In later tutorials we will show how to use the component in a Flow.

The component has 4 design parameters:

- Name : The name of your CauZali model.
- Selection :
 - name = used when you type a name in the name field.
 - list = used when you want the user to select a Model
 - cauzali = used on a record page where you have a field called "cauzali" populated with a model name.
- RecordId : Used only when we use CauZali in a Flow.
- ResultId : Used only when we use CauZali in a Flow.

In this tutorial, we work with the most typical use case. Use CauZali to populate some fields on a record.

Note!

In this tutorial, we presume that you know how to configure a Lightning page with Lightning app builder.

Let's start making an ultra-simple model.

1. From the App Launcher  at the top of the page, select Cauzali.
2. Click the Questionnaires tab.
3. Click the **New** button.
4. Use "MyCompModel" as title and press the **Save** button.

5. In the designer press the **Add** two times.
6. Change first node type to Page, and the second node to Finish.
7. In first node add a next node.
8. Select the “Fields and mapping” tab.
9. In “Map to object” select “lead”
10. Make a field of type integer, Name=numEmp, Label=Number of employees and map it to Employees.
11. Make the field editable on the first node.
12. Make the model active.
13. Save your model.

Now let us place the model on a Lead record page. If you don't have a Lead record page then create one and make it active and assigned to your profile.

Make sure you can see the record details and the number of employees field.

In the Lightning app builder under Lightning component/Custom you will find the “Decision questionnaire” component. Drag the component to your page.

Write “MyCompModel” in the name field, and leave the rest. Already now you will see your model.

Save your page.

Open a lead, write a number in your model and click Next.

You should now see the Employee field on the lead is updated.

Tutorial 10: Scoring

In this first version of Cauzali we have a quite simple scoring functionality, but still it is very useful. In every type of Answer node there is a score field where you can specify a number that will be added to the scoring value. The end value of the scoring can be mapped to a field in a record, used in routing condition or used to decide whether or not a field should be hidden or disabled.

1. From the App Launcher  at the top of the page, select Cauzali.
2. Click the Questionnaires tab.
3. Press **New**.
4. Use “MyScoringModel” as name and press **Save**.
5. Add 3 nodes.
6. Make the last node of type “Finish”.
7. In the first node write “Red or Green?” in Question field.
8. Add 2 answer nodes.
9. Set Answer “Red” in first answer node and “Green” in the next.
10. Set type “Next” in both answer nodes.
11. Press **Content** in the “Red” node and choose Config tab.
12. Set “Add to score” to 50 and press **OK**.
13. Press **Content** in the “Green” node and choose Config tab.
14. Set “Add to score” to 100 and press **OK**.
15. In the second node write “Sex?” in Question field.
16. Add 2 answer nodes under the node.
17. Set Answer “Male” in the first answer node and “Female” in the next.
18. Set Type to “Next” in both answer nodes.

19. Press **Content** in the “Male” node and choose Config tab.
20. Set “Add to score” to -10 and press **OK**.
21. Press **Content** on the “Female” node and choose the Config tab.
22. Set “Add to score” to 20 and press **OK**.
23. **Save**.
24. Try your model in the preview window. You can see the scoring value in Result tab.

In the “Fields and mapping” tab you can specify the optional mapping field.

Tutorial 11: Hide and disable fields

Every field, group, description and image can be hidden or disabled depending on the values of the field itself or other fields. Even scoring value can be used as a dependent variable.

From App Launcher  at the top of the page, select Cauzali.

1. Click on Questionnaires tab.
2. Press **New**.
3. Use “MyHideModel” as Name and press **Save**.
4. In Designer press **Add** two times.
5. Create 3 fields in the first node as following:

Usage	Name	Label	Type
edit	I_n1	Number	Integer
edit	I_s1	Text1	String
edit	I_s2	Text2	String

6. Go to settings in I_s1 and go to DISABLED section.
7. Choose “I_n1” in Field, “Less than” in Operant and 10 in Value.
8. Go to settings in I_s2 and go to HIDDEN section.
9. Choose “I_n1” in Field, “Less than” in Operant and 10 in Value.

10. **Save** your model and test. Try with numbers both greater and less than 10.
11. Now go back to your Designer.
12. Create 3 answer nodes under the first node.
13. Set Answer to "Red" and node Type to "Next" in the first node.
14. Press **Content**. Choose config and set Add to score to 5.
15. In the second answer node, set Answer to "Green", Type to "Next" and score to 10.
16. In the third answer node, set Answer to "Blue", Type to "Next" and score to 15.
17. In your last node (node-2) create an editable string field called I_s3 with Label "Text3". Make this field disabled if score equals 5 and hidden if score equals 10.
18. Test your model for all 3 answers.

Implementing the CauZali runtime component.

When you install CauZali you get a lightning component called “Decision questionnaire”. The component is ready to use in Lightning App Builder, Community builder, Flow Builder and in your own lightning Component. The component has 4 design attributes where some are optional depending on the use case. In this article, we give a quick guide on how to implement the component in different environments.

But first some comments about the different Selection types (the most important design attribute).

Following Selection types are implemented today:

1. Name : This is the default and covers most use-cases. You as the admin can choose which CauZali model is placed on the page.
2. List : Give the user the choice for which model to use. Today every active model will be selectable, so be careful with this option. But do you have a few models all working in the same field it can be useful.
3. Cauzali : Can be used on record pages and require a text field on the object named cauzali. The field is supposed to have a value equal to the name of a cauzali model. Suppose you have a call-center working on leads and you have different models depending on the lead.

Implementation in a different environment.

App Page.

Mapping:

If the model map to an object it will always create a new object.

Design attributes:

Name is the name of your model. (If your selection is set to “list” this is the default model)

Selection is typically set to “name”, but “list” is also valid, “cauzali” isn’t valid.

recordId is ignored on App Pages.

resultId is ignored on App Pages.

Record Page.

Mapping:

If the model map to an object and it's placed on a Record Page with the same object it will update the record. If it isn't the same object type the model will create a new record. If the record page is for the lead or case object and the mapping object has a reference to lead or case this will be set automatically.

Design attributes:

Name is the name of your model. (If your selection is set to "list" this is the default model)

Selection is typically set to "name", but "list" and "cauzali" are also valid.

recordId is ignored on App Pages.

resultId is ignored on App Pages.

Community page (where no object is in focus)

Mapping:

If the model map to an object it will always create a new object.

Design attributes:

Name is the name of your model. (If your selection is set to "list" this is the default model)

Selection is typically set to "name", but "list" is also valid but not recommended, "cauzali" isn't valid.

recordId is ignored on App Pages.

resultId is ignored on App Pages.

Community page (where an object is in focus)

Record Page.

Mapping:

If the model map to an object and it's placed on a Record Page with the same object it will update the record. If it isn't the same object type the model will create a new record. If the record page is for the lead or case object and the mapping object has a reference to lead or case this will be set automatically.

Design attributes:

Name is the name of your model. (If your selection is set to “list” this is the default model)

Selection is typically set to “name”, but “list” is also valid but not recommended and “cauzali” are also valid.

recordId has to be set to “{!recordId}”

resultId is ignored on App Pages.

Flow Builder.

The CauZali model has to be placed on a Screen element.

Mapping:

If the recordId is set the mapping work just as described under Record Page, if not set it work as when we place the model on an app page.

Design attributes:

Name is the name of your model.

Selection is set to “name”, but “list” and “cauzali” is also valid but not recommended.

recordId has to be set to a variable with a recordId or blank.

resultId is used as an output variable and can be used to get the recordId on the result record.

Result object.

Sometimes you don't want to map data to an object, but still, want to have data after a user has used your model. For this, you can check the "Create result record" on.

This will add a record to the "Decision result" object where you get some data, which you can use to reporting or documentation. If you want to save even more you can turn on "Create detailed results records", then each variable and each select will be saved.

If you use cauzali in a Flow and want to get information after the CauZali component, you have to turn on "Create result record".