

How to Create a PDF Quotes in Seconds by Flows

Step1: Create an Apex Class in the Developer Console:

1. **Open the Developer Console:**
 - In Salesforce, click on your **Avatar/Profile** at the top right corner.
 - Select **Developer Console** from the dropdown menu. This will open a new window.
2. **Create a New Apex Class:**
 - In the Developer Console, go to **File > New > Apex Class**.
 - A popup will appear asking for a name for your Apex class. Enter the desired name (e.g., generateQuotePdfDocument) and click **OK**.

Paste the Given below Code

```
Public class generateQuotePdfDocument{

    @InvocableMethod

    public static void CreateQuote(List<Id> quoteIds)

    {

        createQuoteFutureMethod(quoteIds);

    }

    @future(callout=true)

    public static void createQuoteFutureMethod (List<Id> quoteIds) {

        //Initialize the quote url

        String quoteUrl = '/quote/quoteTemplateDataViewer.apexp?';

        //Get the Quote Template Id from Custom Settings

        String quoteTemplateId = Label.Quote_TemplateId;
```

```
System.debug('quoteTemplateId ::' +quoteTemplateId);
```

```
//List variable to get all the Quote Documents
```

```
List<QuoteDocument> lstQuoteDoc = new List<QuoteDocument>();
```

```
System.debug('quoteIds ::' +quoteIds);
```

```
if(!quoteIds.isEmpty() && quoteIds.size() > 0) {
```

```
    for(Id quoteId :quoteIds) {
```

```
        //Construct the quote URL to generate PDF
```

```
        quoteUrl += 'id=' + quoteId;
```

```
        quoteUrl += '&headerHeight=100&footerHeight=75';
```

```
        quoteUrl += '&summlid=' + quoteTemplateId;
```

```
        //call the quote url
```

```
        PageReference pageRef = new PageReference(quoteUrl);
```

```
        //get the quotePdf
```

```
        Blob quoteBlob;
```

```
        if(Test.isRunningTest()) {
```

```
            quoteBlob = Blob.valueOf('Generate Pdf');
```

```
        } else {
```

```
            quoteBlob = pageRef.getContentAsPDF();
```

```
        }
```

```

        //initialize the QuoteDocument to hold the quote pdf for insertion

        QuoteDocument quoteDoc = new QuoteDocument();

        quoteDoc.Document = quoteBlob;

        quoteDoc.QuoteId = quoteId;

        lstQuoteDoc.add(quoteDoc);

    }

}

System.debug('lstQuoteDoc ::' +lstQuoteDoc);

if(!lstQuoteDoc.isEmpty() && lstQuoteDoc.size() > 0) {

    Database.insert(lstQuoteDoc);

}

}

}

```

Step 2: To get the **Quote Template Id** from Custom Labels in Salesforce, follow these steps:

1. Create Custom Labels in Salesforce:

1. **Go to Setup:**
 - Log in to Salesforce and navigate to **Setup**.
2. **Search for Custom Labels:**
 - In the **Quick Find Box**, type **Custom Labels** and click on it under the **User Interface** section.
3. **Create a New Custom Label:**
 - Click on the **New Custom Label** button at the top of the page.
4. **Define the Custom Label:**
 - **Label:** Enter a descriptive **Label Name** (e.g., Quote TemplateId).
 - **Name:** Salesforce automatically generates the **API Name** (e.g., Quote_TemplateId), which you will reference in Apex.

- **Value:** Enter the actual value for this label (e.g., the **Quote Template Id** like 00Xxx000000Xyz).
- **Language:** Choose the language if you're working with multi-language support.
- **Protected Component:** You can make this label protected if needed (only accessible to certain packages or users).

5. Save the Label:

After filling out the necessary details, click Save.

Step 3: How to get the Id of Template.

To get the **Id of a Quote Template** or any template (like email templates, document templates, etc.) in Salesforce, you have several options depending on your scenario. Here are the steps for each method:

1. From the Salesforce User Interface (UI):

If you're looking to manually retrieve the **Id** of a template, follow these steps:

- Go to the Template List:**
 - For **Quote Templates**:
 - Go to **Setup** and search for **Quote Templates**.
 - Click on **Quote Templates** to see a list of templates.
 - For **Email Templates** or other types of templates, search for the specific template type in the **Quick Find** box in **Setup**.
- Open the Template:**
 - Click on the name of the template whose Id you want to retrieve. The template's detail page will open.
- Retrieve the Id from the URL:**
 - Look at the browser's URL. The URL will contain the **Id** of the template, usually after the domain part and a "/lightning/r/" or "/lightning/o/" segment.

For example:

`https://yourInstance.salesforce.com/00XXXXXXXXXXXXXXXXXXXX`

Here, 00XXXXXXXXXXXXXXXXXXXX is the **Template Id**.

Note -> Here's an example: Only pick the highlighted part in green, ignoring the first two characters. Remember, only the highlighted ID should be used in your custom labels.

XXXXXXXXXXXXX lightning/setup/QuoteTemplateEditor/page?address=%2Fquotetemplate%2Fmlid%3D0EHBg0000001W5t%26retURL%3D%252Fui%252Fquote%252FQuoteTemplateLayouts%253FretURL%253D%25252Fsetup%25252Fhome%2526appLayout%253Dsetup%2526tour%253D%2526isdt%253Dp1%2526sfcdIFrameOrigin%253Dhttps%25253A%25252F%25252Fsmgsal

1cfd3d588331657f98bb08f5090bf60ac796a04c776a9b80%25261tn_app_id%253D06m0K000001m2x7QAA%2526clc%253D1

Step 4: Here are the steps to create a Record-Triggered Flow for the Quote object in Salesforce:

1. Navigate to Flow Builder

- Go to **Setup**.
- In the **Quick Find** box, search for **Flows**.
- Click **Flows** and then click **New Flow**.

2. Choose Flow Type

- In the flow type selection screen, select **Record-Triggered Flow**.
- Click **Create**.

3. Configure the Start Element

- Choose the **Object** as **Quote**.
- Select the **Trigger**:
 - **Trigger the Flow When**: Choose **A record is created or updated** (or any other depending on your use case).
- Specify **Conditions** (if needed), such as triggering only when specific fields are updated.
- Choose the **Run the Flow** setting:
 - **Actions and Related Records**: For **after save** flows (if you want to perform actions like creating related records or sending emails or Apex Actions).

4. Add the Apex Action to the Flow

- In the **Flow Builder**, after setting up the **Start** element, drag an **Action** element from the toolbox onto the canvas.
- Search for **Apex** actions, and look for `generateQuotePdfDocument`.

Select it and configure any required **input variables** (if needed by the Apex method).

Step 5: To implement the button and actions for generating a PDF and adding the custom field in Salesforce, follow these steps:

1. Create a Custom Action (Button) for 'Create PDF'

Step 1.1: Create a New Action

1. Navigate to **Setup**.
2. In the **Quick Find** box, search for **Object Manager**, then select the **Quote** object.
3. In the **Buttons, Links, and Actions** section, click **New Action**.
 - **Action Type**: Choose **Update a Record**.
 - **Label**: Set the label to **Create PDF**.
 - **Success Message**: Set the success message to **A PDF has been generated**.
4. Click **Save**.

Step 1.2: Customize the Action Layout

1. Once the action is created, you will be directed to the layout editor for the action.
2. Drag and drop fields that you want to include in this action. Ex. **Do you want to create PDF?**
3. Click **Save**.

2. Add a Custom Field 'Do you want to create PDF?'

Step 2.1: Create a Custom Field

1. Navigate to **Setup**.
2. Go to **Object Manager** and select the **Quote** object.
3. In the **Fields & Relationships** section, click **New**.
4. Choose the field type as **Checkbox**.
5. Set the field label to **Do you want to create PDF?** and set the default value to **Unchecked**.
6. Click **Next**, set field-level security, and click **Save**.

3. Add the Action Button to the Layout

Step 4.1: Add the Action Button to the Quote Layout

1. Go to **Page Layouts** in the **Object Manager** for **Quote**.
2. Open the **Page Layout** where you want to add the **Create PDF** action.
3. In the **Mobile & Lightning Actions** section, drag the **Create PDF** action into the layout where you want it to appear (e.g., under the **Salesforce Mobile and Lightning Experience Actions** section).
4. Save the page layout.

When clicked, it will generate a PDF and display the success message: **A PDF has been generated.**

Abhishek Singh