Deloitte.

Salesforce manual
"how to integrate Google maps within a Visualforce page"

Introduction

Showing a map alongside an address is really useful for your end users. In this tutorial you will learn how to extend Salesforce by including a Google Map within a Visualforce page and then displaying this on a page layout.

This tutorial is intended to be a gentle introduction to Visualforce for Salesforce administrators with no past experience developing in Visualforce. In this tutorial we will cover not only how to use Google Maps within Visualforce, but all of the steps needed from beginning to the end and everything in between.

What are we going to build?

We are going to create a Visualforce page to display a Google Map displaying the address of an account in Salesforce. This will be displayed on the page layout.

What tools do I need?

Everything you need to get going is within Salesforce. You don't need any tools for this tutorial, all you need is somewhere to do your coding!

Unless you are the administrator, you may need to ask for additional permissions from your system administrator to allow you to customise your Salesforce org. It advise to do the following within a sandbox or use a developer org so not to interfere with your production system.

If you would like to have your own Salesforce org where you can safely learn and customise, then go to http://developer.force.com and register for a Developer Edition org. These orgs are intended for people to learn and develop new applications for the platform.

Let's get started!

Now it's time to start building our Visualforce page which will contain our Google Map. In this section you will introduced to the developer console, how to create a Visualforce page and how to include it on an account page layout.
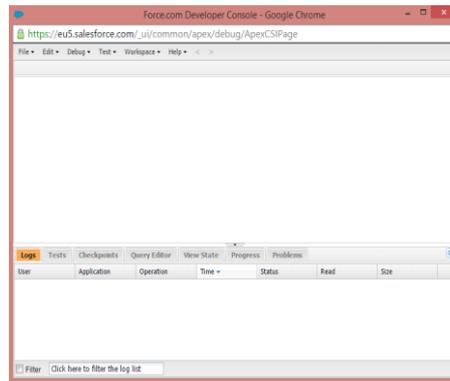
Opening the developer console

The developer console is where we will be creating our first Visualforce page.

To open the developer console follow these steps:

- Click your name at the top right of the screen.
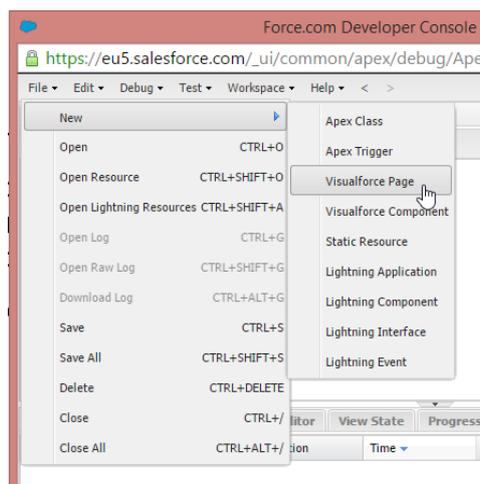- Click developer console.
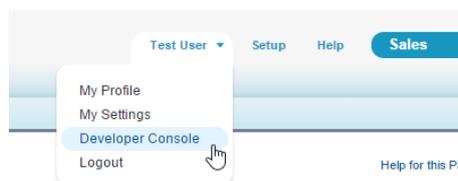
Clicking on developer console will open a new window which should look something similar to the following:
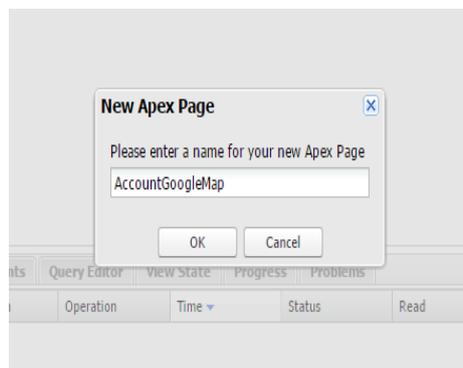

Creating our first Visualforce page

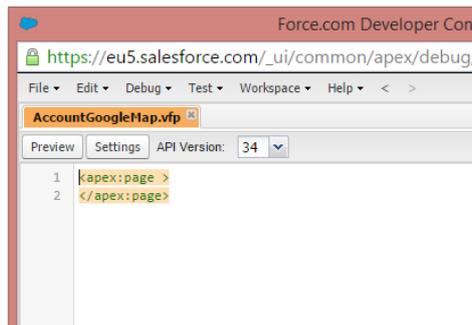First we need to create a blank Visualforce page for us to modify. To do this follow these instructions:

- Click File
- Hover over New
- Click Visualforce Page.

Salesforce will now ask you for a name you want to use for the page. Provide a useful name such as the following:

You should now see your first Visualforce page ready to be edited within the developer console:

If you see the above in your developer console, great! You've successfully created your first Visualforce page!

Introducing our Google Map

Now we can bring in our Google Map into your new Visualforce page.

Copy and paste the following code and replace the existing code within your new Visualforce page which you created in the previous step.

```
<apex:page standardcontroller="Account">

<apex:map width="600px" height="400px" mapType="roadmap"
center="{!Account.BillingStreet},{!Account.BillingCity},{!Account.BillingState}"
zoomlevel="16">
</apex:map>

</apex:page>
```
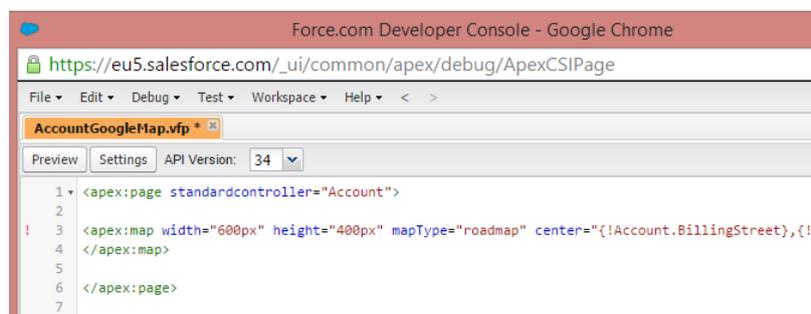
After copying and pasting in the code, save it by pressing CTRL+S on your keyboard or by clicking File > Save.

The developer console should look similar to the following:

What does it all mean?

The page starts and finishes with a Visualforce tag called `apex:page`. This just tells Salesforce that is dealing with a Visualforce page.

On the opening tag you will also notice there is an attribute called "standardcontroller" with a value of "Account". Standard controllers are available for the majority of objects in Salesforce, such as opportunities and contacts. They allow us to retrieve data from the account currently in context. In this tutorial the context is the account for which our Visualforce map will be viewed with.

Within our page we define our Google Map using the following code.

```
<apex:map width="600px" height="400px" mapType="roadmap"
center="{!Account.BillingStreet},{!Account.BillingCity},{!Account.BillingState}"
zoomlevel="16">
</apex:map>
```

The map has 3 main attributes which are of interest to us; mapType, center and zoomlevel.

- mapType – you can change the map type to satellite, roadmap or hybrid.
- center – this is the position on the map for which to centre what the user sees.
- zoomlevel – controls how zoomed in we should be when viewing the map.

Displaying your Visualforce page on an account page layout

We're nearly finished in our quest of displaying a Google Map on the account page layout. All we have to do now is to edit the account page layout and add it where we want to have it displayed.

To do this edit one of the page layouts for the account object in Salesforce so you see the following screen (Setup > Customize > Accounts > Page Layouts):

When you are in the page layout editor click on Visualforce Pages from the components list within the top toolbar.
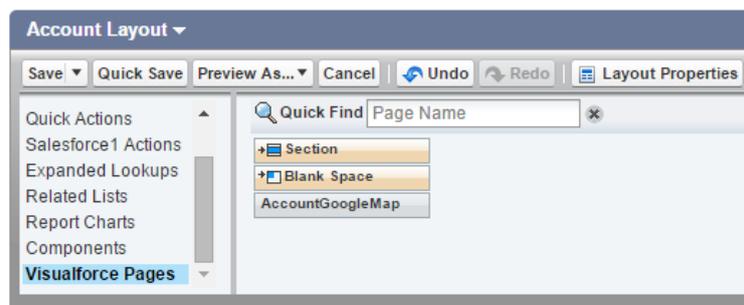
If you don't see the Visualforce page available in the list above then most likely you have missed out the following from the beginning of the code to be copied.

```
<apex:page standardcontroller="account">
```

Without the above attribute Salesforce will not know that this page can be used with an account. Go back to the developer console and recheck you have copied the code correctly.
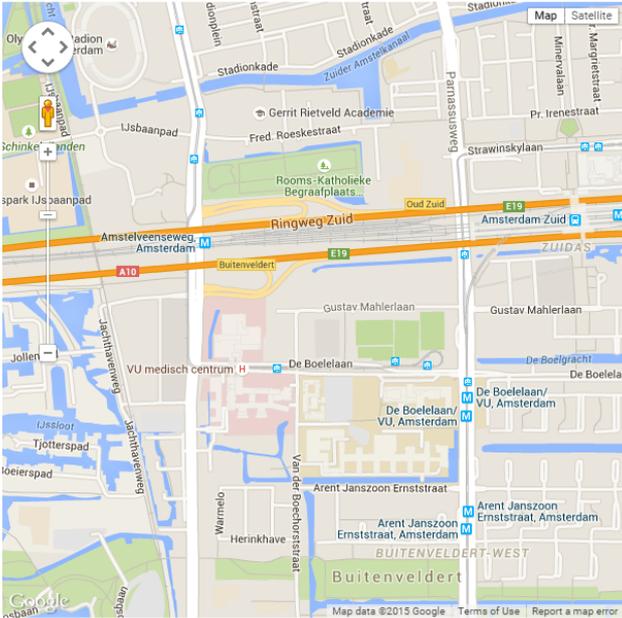
Test drive

Now we have added the Visualforce page to an account layout we can now go and preview it. Navigate to an account in Salesforce and you should see your Visualforce page with the Google Map loaded:

If you see the larger map on your page layout you have successfully added the Visualforce page. The smaller map is the one Salesforce automatically adds next to standard address fields in Salesforce.

If you don't, check that you have added the Visualforce page to the correct page layout and try again.

Adding a map marker – show the user where the account exactly is!

You might have noticed when we previewed our Google Map there was no map marker to display to the user exactly where the accounts billing address really is.

To add a map marker we need to tell Salesforce where to place it and this is done between the opening (`<apex:map>`) and closing (`</apex:map>`) map tags.

```
<apex:page standardcontroller="Account">


<apex:map width="600px" height="400px" mapType="roadmap"
center="{!Account.BillingStreet},{!Account.BillingCity},{!Account.BillingState}">
    <apex:mapMarker title="{!Account.Name}"
position="{!Account.BillingStreet},{!Account.BillingCity},{!Account.BillingState}" />
</apex:map>


</apex:page>
```

So what is going on in the above code? We have introduced a new Visualforce tag called `apex:mapMarker`. Map markers instruct the map to drop a marker at a given position on the map.

In the above example you can see there are 2 attributes; title and position. The title attribute will contain text which will be displayed to the user when they hover over the map marker. Whereas, position instructs the map where to drop the marker. In our case, we will reuse the same location as the centre of the map which is the address of the account.

Conclusion

We hope this tutorial has given you inspiration to learn more about Visualforce and how you can use it to customise your own Salesforce org.

If you want to find out more check out the documentation at Salesforce:

https://developer.salesforce.com/docs/atlas.en-us.pages.meta/pages/pages_compref_map.htm