

08:12



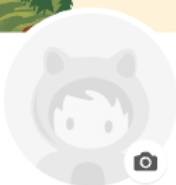
4G



19%



salesforce.com/trailblazer/prof



Kuruba Vani

Administrator at Salesforce

Andhra Pradesh, India

[Tell us about yourself! Add a short bio.](#)

[salesforce.com/trailblazer/kurubavani](#)

[Add Hire Me Button](#)

Certifications



Add Certifications

Learn how to add additional certifications, including Salesforce, Slack, MuleSoft, Tableau, and Accredited Professional to your profile.

[Learn More](#)

9 Superbadges



Superbadge

Business Administration Specialist

Completed October 31, 2023

Brighten up a new business unit with the Salesforce tools the team needs to succeed.



Superbadge

User Authentication Settings Superbadge Unit

Completed October 30, 2023

Bring user authentication settings up to standard to secure your org.



Superbadge

Security Governance Specialist Superbadge

Completed October 30, 2023

Complete the capstone assessment to earn the Security Governance Specialist Superbadge.

[Show More](#)

Skills

Skills you developed while earning Trailhead badges

Trailhead



69

Badges

45,875

Points

3

Trails



Earn 31 more badges and 6,125 more points to reach [Ranger](#) rank.

[Go to Trailhead](#)

Community Tags



Add your community tags here

Questions & Answers

0

Answers

0

Best Answers

0

Questions

[Go to Community Feed](#)

Connections



Trailblazing is better together

Make your first connection by following a Trailblazer or group.

0

Followers

0

Following

0

Groups

Files

[View Files](#)

errors.

- If a DML statement in Apex spawned the trigger, any error rolls back the entire operation. However, the runtime engine still processes every record in the operation to compile a comprehensive list of errors.

Triggers and Callouts

Apex allows you to make calls to and integrate your Apex code with external Web services. Apex calls to external Web services are referred to as callouts. For example, you can make a callout to a stock quote service to get the latest quotes. When making a callout from a trigger, the callout must be done asynchronously so that the trigger process doesn't block you from working while waiting for the external service's response. The asynchronous callout is made in a background process, and the response is received when the external service returns it.

To make a callout from a trigger, call a class method that executes asynchronously. Such a method is called a future method and is annotated with `@future(callout=true)`. This example class contains the future method that makes the callout.



The example uses a hypothetical endpoint URL for illustration purposes only. You can't run this example unless you change the endpoint to a valid URL and add a remote site in Salesforce for your endpoint.

```
1 public class CalloutClass {
2     @future(callout=true)
3     public static void makeCallout() {
4         HttpRequest request = new HttpRequest();
5         // Set the endpoint URL.
6         String endpoint = 'http://yourHost/yourService';
7         request.setEndPoint(endpoint);
8         // Set the HTTP verb to GET.
9         request.setMethod('GET');
10        // Send the HTTP request and get the response.
11        HttpResponse response = new HTTP().send(request);
12    }
13 }
```

[Copy](#)

This example shows the trigger that calls the method in the class to make a callout asynchronously.

```
1 trigger CalloutTrigger on Account (before insert, before update) {
2     CalloutClass.makeCallout();
3 }
```

[Copy](#)

This section offers only an overview of callouts and is not intended to cover callouts in detail. For more information, see [Invoking Callouts Using Apex](#) in the Apex Developer Guide.

Resources

- [Apex Developer Guide: Triggers](#)
- [Apex Developer Guide: Invoking Callouts Using Apex](#)
- [Trailhead: Apex Integration Services](#)

ASSESSMENT COMPLETE!

+500 points



Apex Triggers

100%

[Retake this Challenge](#)

[View more modules](#)