



Apex Trigger

Course Overview

BY VIJAY KUMAR





Introduction

Welcome to "**Master Apex Trigger with Real-Time Challenges and Solutions: A Digital Course**".

With this thorough course, we are pleased to guide you through the complicated world of "**Apex Triggers**". if whether you're an expert developer looking to **improve your skills** or someone who is eager to get into Salesforce, **this course** will provide you with the understanding and **hands-on practice** needed to master Apex Trigger.

In this course I have covered complete syllabus of Salesforce Apex Trigger with real-time project example step-by-step for beginner to advanced label.

Where you will be learn **advanced technical scenarios** based on live demo projects example of Apex Trigger in Salesforce.

This course is intended for developers who want to learn how to automated business processes with Apex triggers in Salesforce.

It is suitable for developers who have experience with JavaScript and web development, and want to learn Apex Trigger Development in Salesforce.

It is also suitable for experienced Salesforce developers who want to learn how to use the **Apex Trigger** to build more efficient automated business processes with Apex triggers in Salesforce.

I have **taken care** of who are brand new to start Salesforce Career as a developer and want to **grow their Career** in Salesforce application and development.



What You Will Learn

- ✦ *What is Apex Trigger, Context Variables, DML Statements, Trigger Execution, and Type of Data in Apex Trigger Salesforce*
- ✦ ***Best Practice of Apex Trigger, Apex Class, and Apex Test Class in Salesforce***
- ✦ ***Order of execution, and Exception Handling in Salesforce***
- ✦ *Upon Contact record creation if LeadSource Field is having value as '**Web**' or '**Phone Inquiry**' then Populate Level as '**Primary**'*
- ✦ *When an Account record is created then Create a **related Contact** automatically in Salesforce*
- ✦ *When an Account record is created then Create a **related Opportunity** automatically in Salesforce*
- ✦ *When an Account record is created then **assigned the Billing Address** fields value to **Shipping Address** fields value in Salesforce*



What You Will Learn

- ✦ Update the **related Account Phone** if Contact Record is Created in Salesforce
- ✦ **Don't allow** to change **Amount**, if Opportunity Old Amount not equal to New Amount in Salesforce
- ✦ If Account Active fields selected "Yes" and **Related Opportunities** Stage has "**Closed Lost**" then display error message in Salesforce
- ✦ **Do not allow** create **duplicate record** of Contact if associated Account has same Phone number in Salesforce
- ✦ Whenever new contact is created, we will be **updated correspondence** account field of Status
- ✦ Before creating a Lead, we will **check Email**, if email **already exist**, then we will display error message, do not allow create duplicate Lead
- ✦ **Update phone of all Contact**, associated with a particular Account have the same phone number, if **Account is updated** in Salesforce



What You Will Learn

- ✦ How to prevent **recursive trigger** while updating Contact record through Apex Trigger in Salesforce.
- ✦ Do **not allow delete** record, if Opportunity Stage has "Closed Won" in Salesforce.
- ✦ Whenever Opportunity is **deleted the corresponding Account** and Contact should be automatically deleted.
- ✦ Trigger on custom object to **prevent delete record**, if Parent object has **Child record**, then we don't allow to delete Parent record.
- ✦ Write a Trigger on custom object to Update the **Last Modified Date** in the Parent Object When the Records is Updated or Inserted in Salesforce.
- ✦ Write an Apex trigger to Add the Contact **First Name** and **Last Name** to **Associated Account** Custom Field, Whenever the Contact record is Inserted or Updated in Salesforce.
- ✦ **Before delete Account**, we will check **related Opportunity Stage**, if Opportunity Stage has "Closed Own", then do not allow delete Account record.



What You Will Learn

- ✦ **Do not allow** delete Account, if any of **related Contact** associated with Account in Salesforce.
- ✦ **Update all Email** of Contact **related list** to the **associated** Account if Account is updated in Salesforce.
- ✦ Update **Parent** field when **Child record** is Updated or Inserted in Salesforce.

