Jermaine Micah M. Rianzares BSIT 4-1

PUP STC

WhatsNext Vision Motors

Shaping the Future of Mobility with Innovation and Excellence

WhatsNext Vision Motors, a pioneering force in the automotive industry, is dedicated to transforming the mobility sector through innovative technology and customer-focused solutions. This Salesforce project aims to enhance customer experience, streamline the ordering process, and strengthen operational efficiency across the organization.

Project Overview

This Salesforce implementation improves the customer ordering process by automatically suggesting the nearest dealer based on the customer's address. This feature enhances convenience and reduces the effort required for customers when placing vehicle orders.

The system also prevents order placement for vehicles that are out of stock. This ensures that customers can only place orders for available units, reducing confusion and boosting order accuracy.

A scheduled process is included to automatically update bulk order records. If a vehicle is out of stock, the order status becomes Pending. If the vehicle is available, the order status becomes Confirmed. This automation promotes transparency and accurate communication with customers.

Overall, this project enhances efficiency, reduces manual workload, minimizes errors, and improves customer satisfaction.

Requirements

1. Salesforce CRM Implementation

- Store and manage vehicle details, stock availability, and dealer information.
- Track customer orders, test drives, and service requests.
- Automatically assign orders to the nearest dealer based on customer location.

2. Process Automation

- Prevent order placement for out-of-stock vehicles.
- Auto-assign orders to the nearest dealer.
- Send automated email reminders for scheduled test drives.

3. Apex and Triggers

- Implement Apex triggers for stock validation and dealer assignment.
- Use a trigger handler framework for modular and maintainable code.

4. Batch Jobs

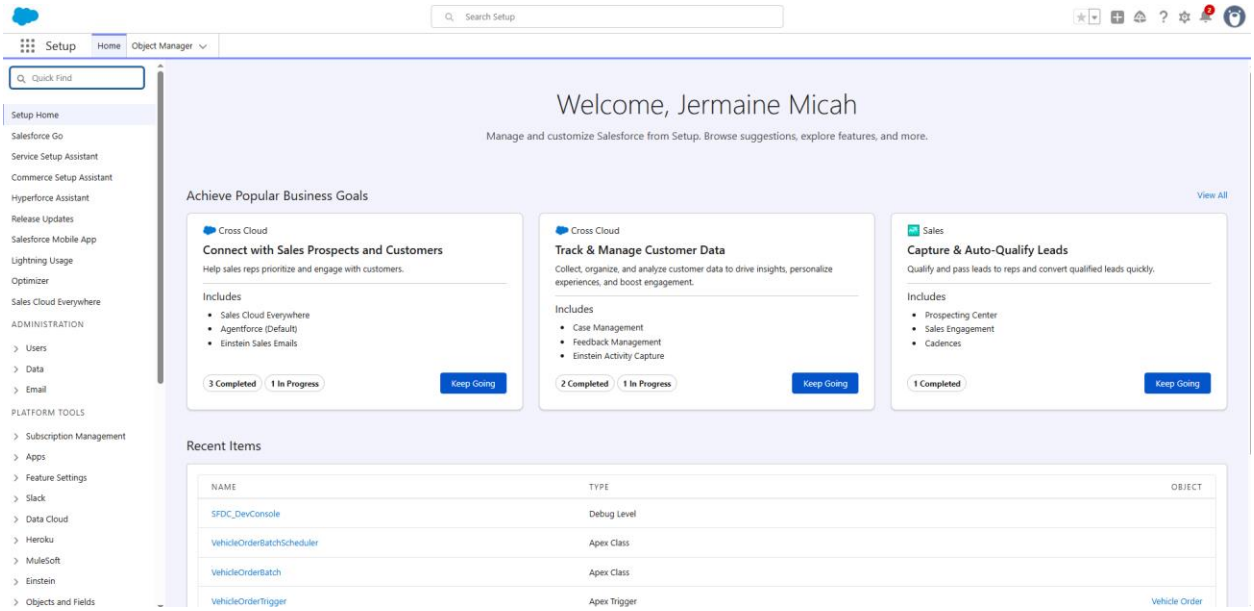
- Create a Batch Apex job to periodically check and update stock availability.
- Send scheduled email notifications for stock replenishment and order processing.

What You'll Learn

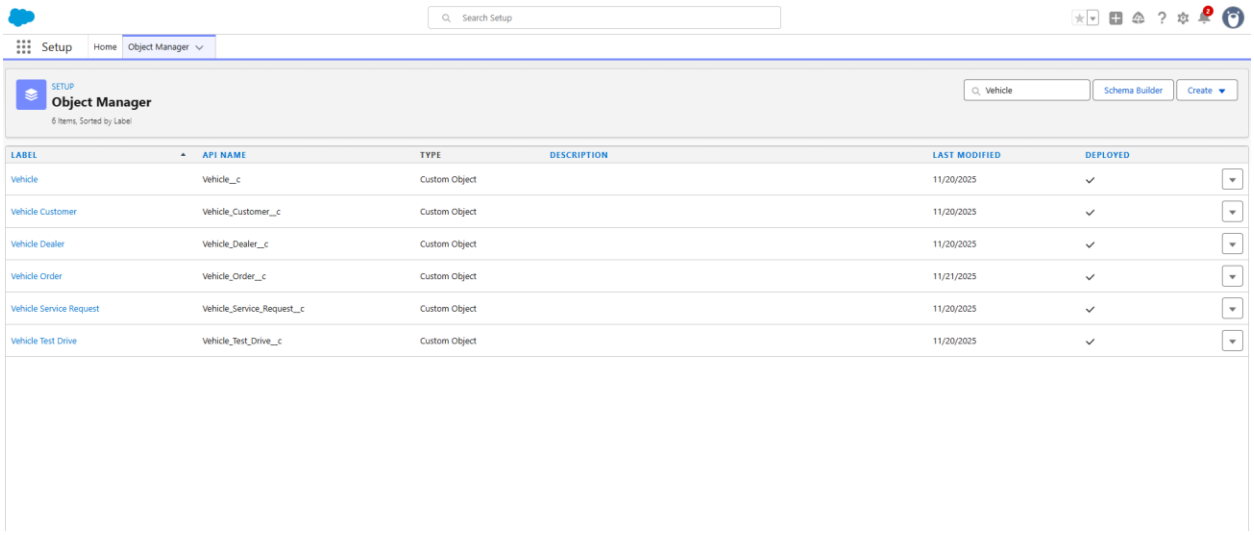
- Data Modelling
- Fields and Relationships
- Lightning App Builder
- Record-Triggered Flows
- Apex and Apex Triggers
- Batch Apex
- Scheduled Apex

Project Progress

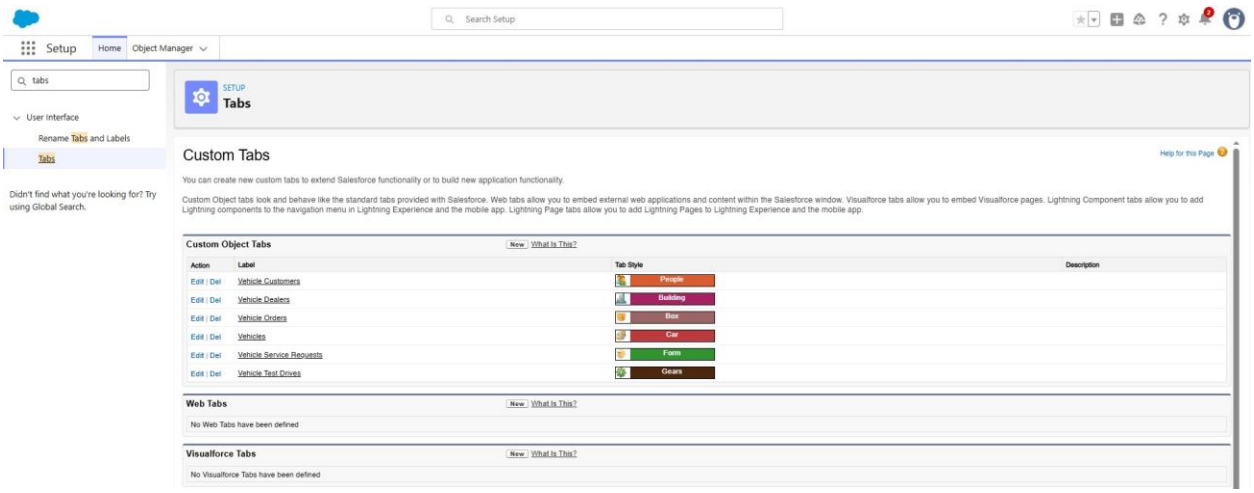
Salesforce Credentials Creation



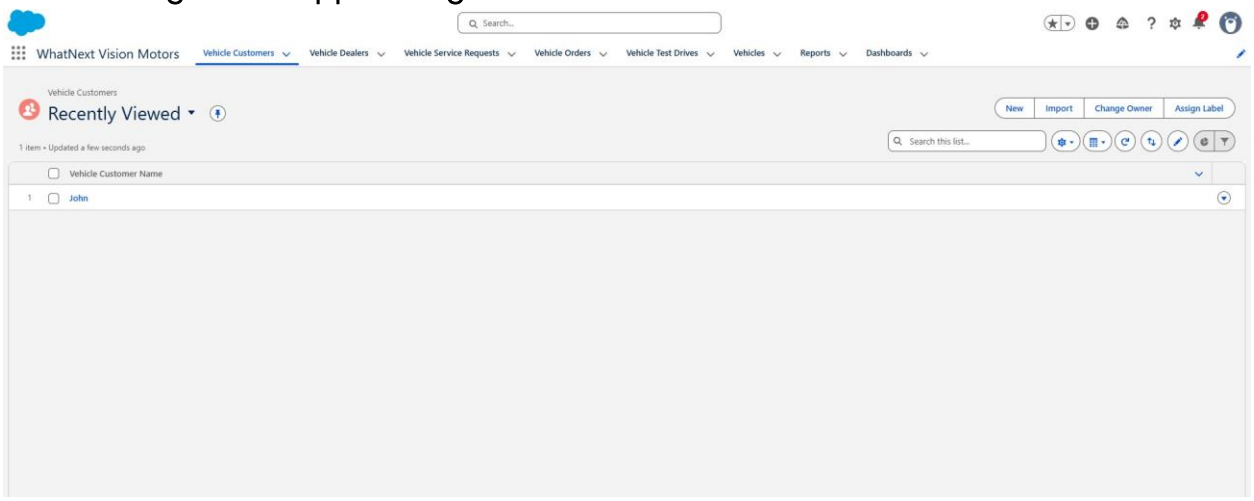
Data Management-Objects



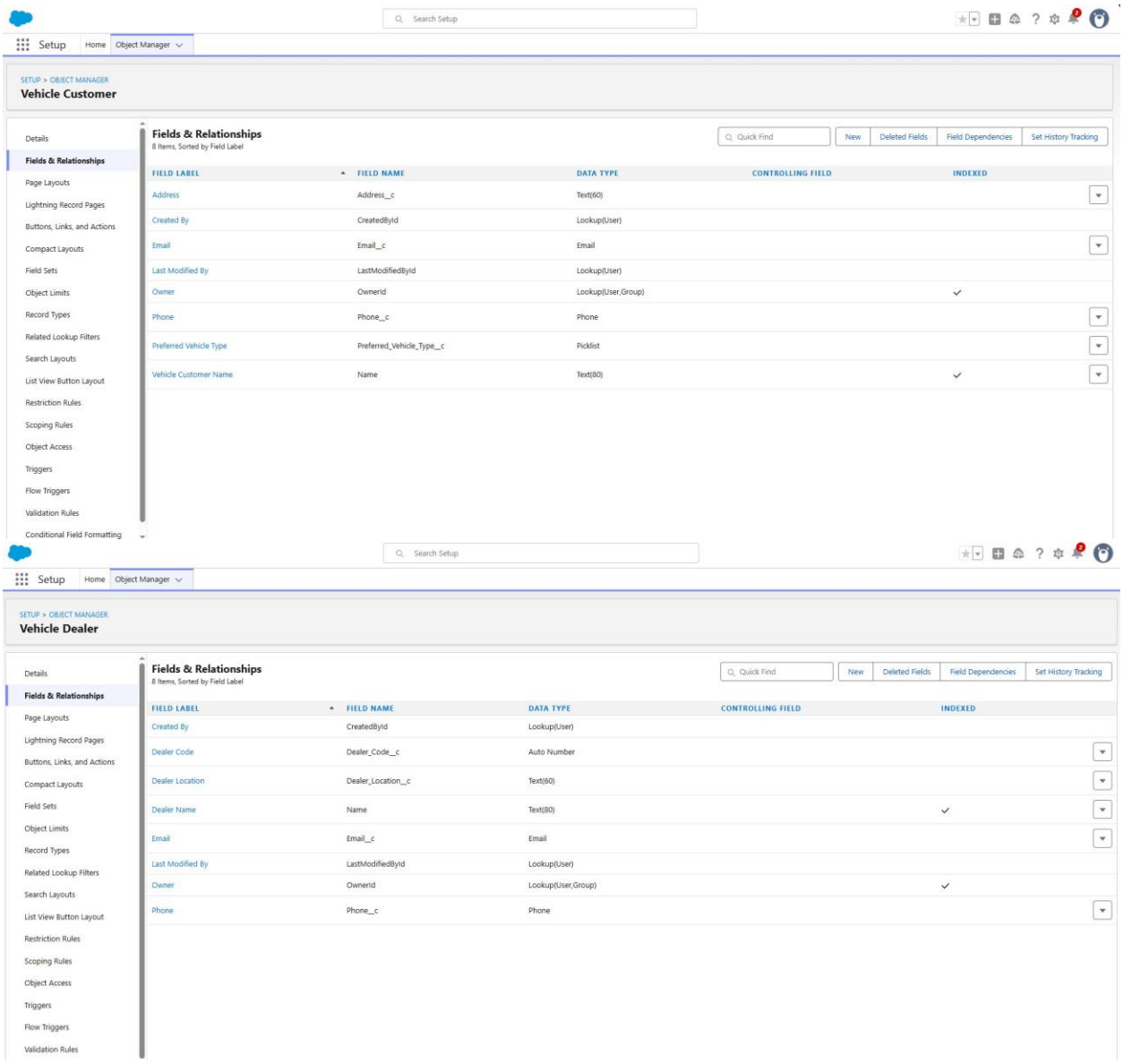
Data Management-Tabs



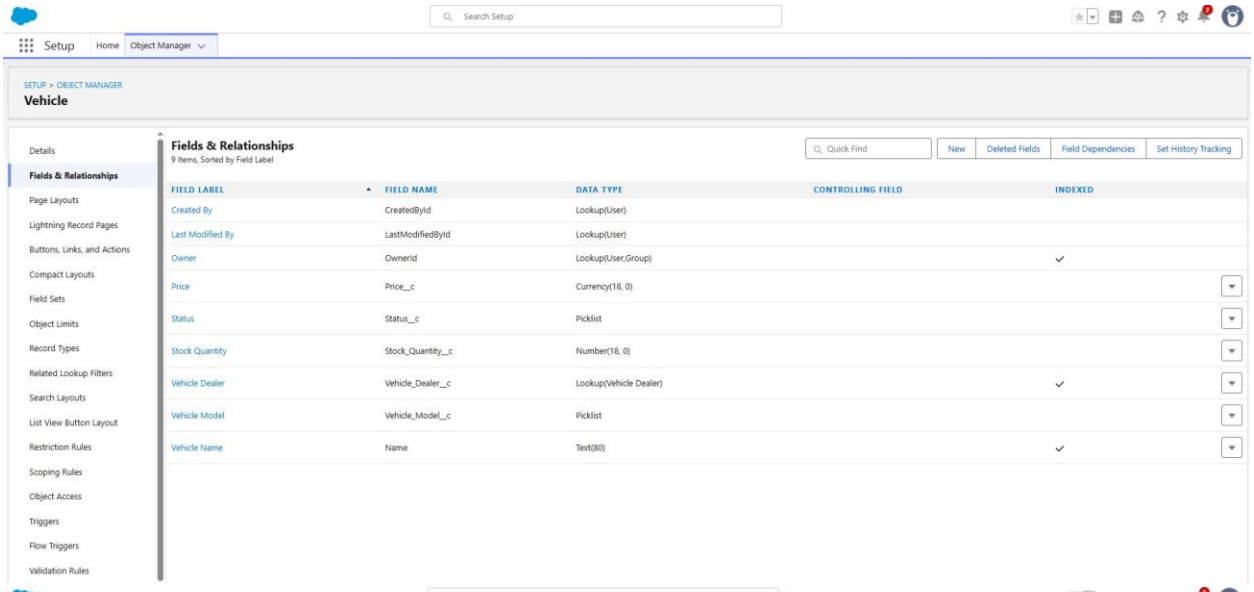
Data Management-App Manager

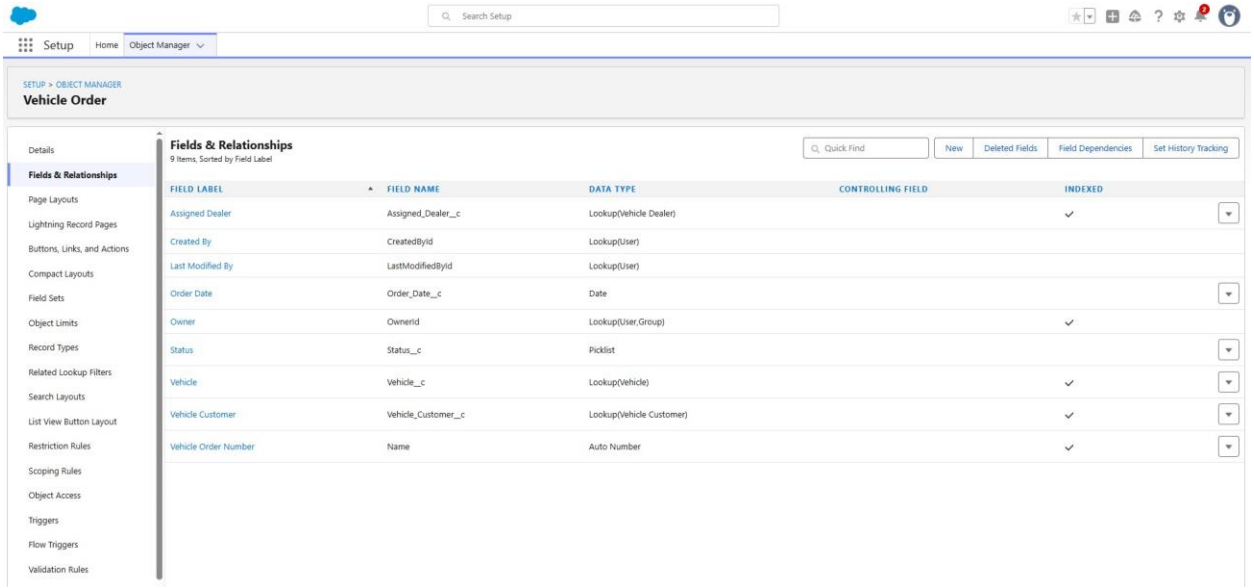


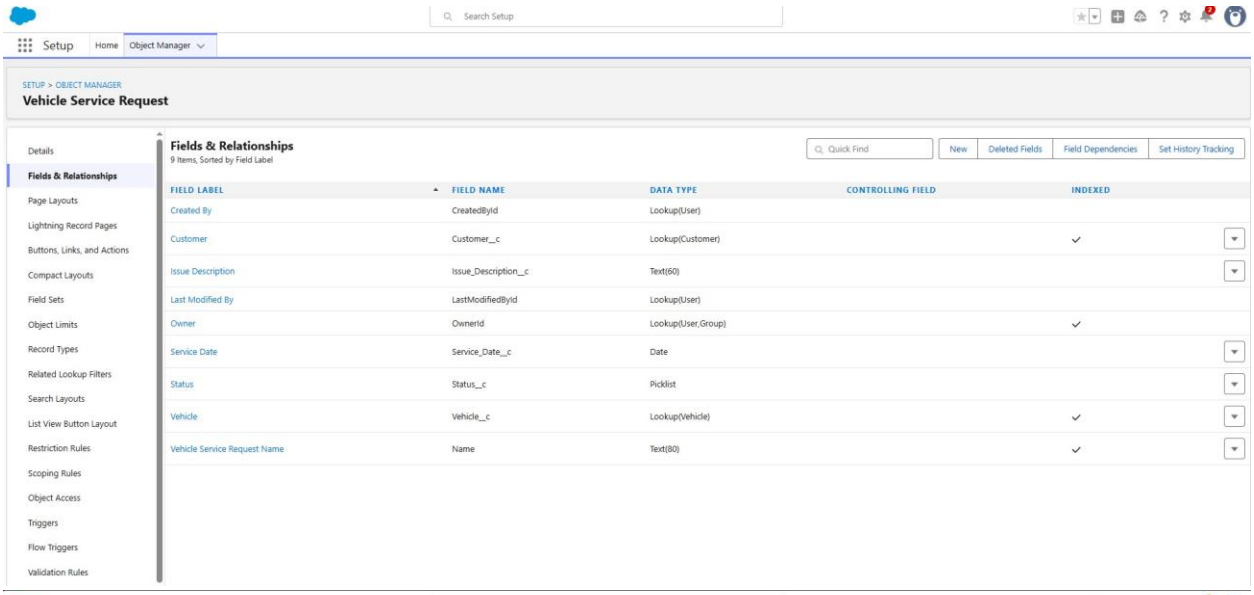
Data Management-Fields

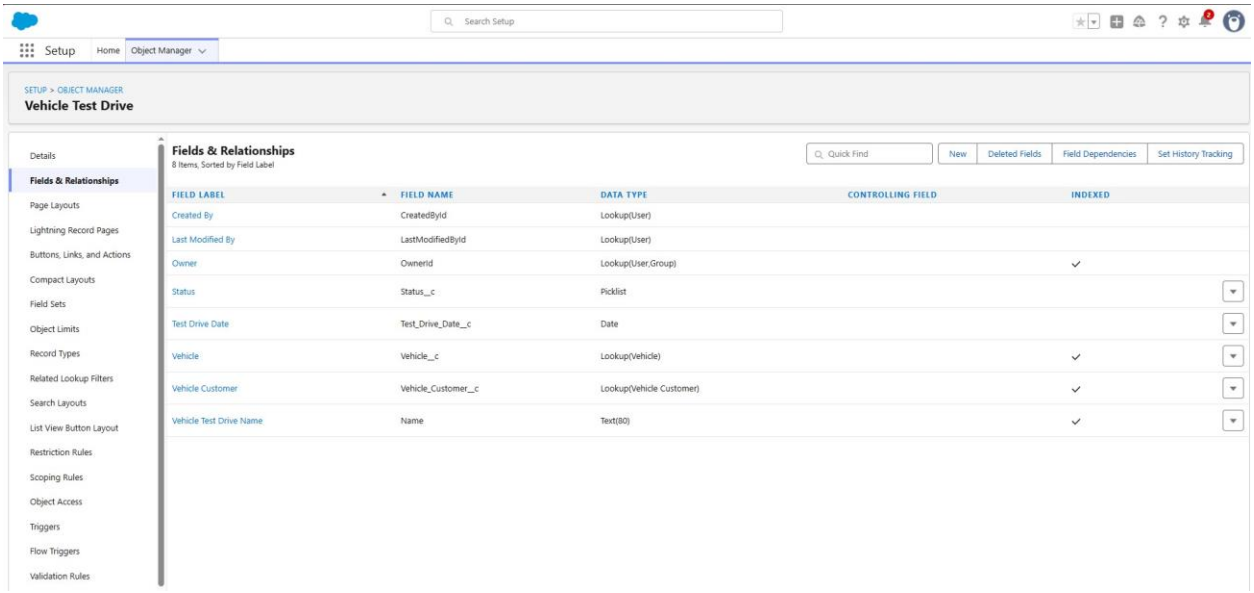

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Vehicle Customer
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers
Validation Rules
Conditional Field Formatting
Fields & Relationships
8 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Address
Address__c
Text(60)
Created By
CreatedById
Lookup(User)
Email
Email__c
Email
Last Modified By
LastModifiedById
Lookup(User)
Owner
OwnerId
Lookup(User,Group)
Phone
Phone__c
Phone
Preferred Vehicle Type
Preferred_Vehicle_Type__c
Picklist
Vehicle Customer Name
Name
Text(80)

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Vehicle Dealer
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers
Validation Rules
Fields & Relationships
8 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Created By
CreatedById
Lookup(User)
Dealer Code
Dealer_Code__c
Auto Number
Dealer Location
Dealer_Location__c
Text(60)
Dealer Name
Name
Text(80)
Email
Email__c
Email
Last Modified By
LastModifiedById
Lookup(User)
Owner
OwnerId
Lookup(User,Group)
Phone
Phone__c
Phone

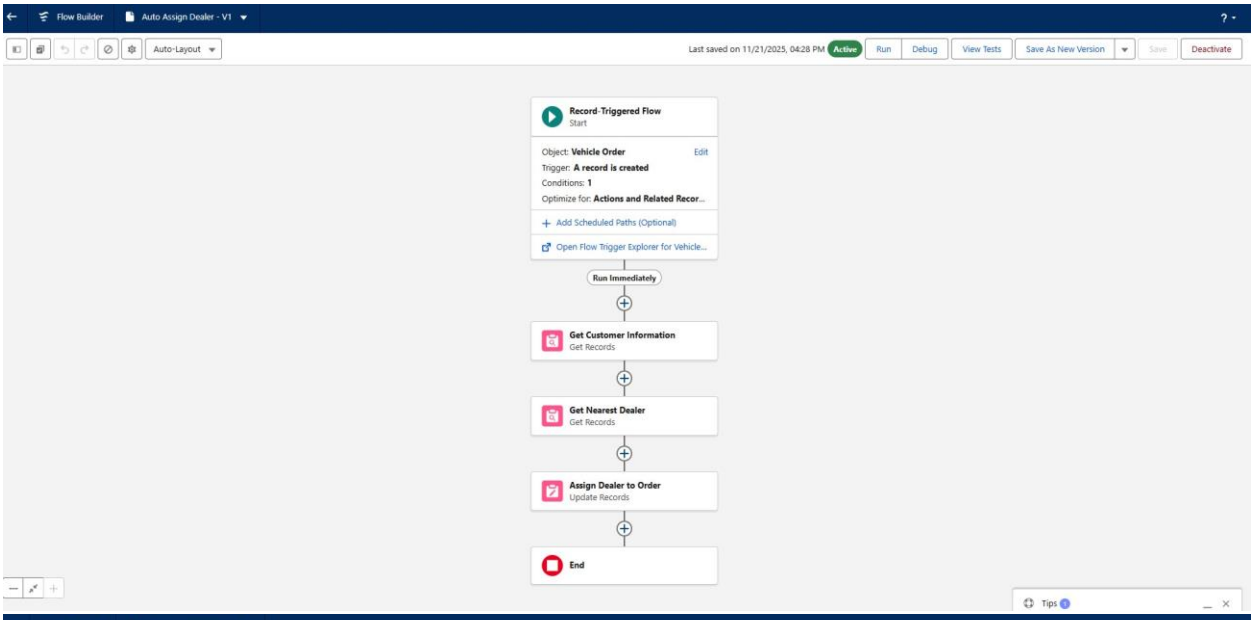

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Vehicle
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers
Validation Rules
Fields & Relationships
9 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Created By
CreatedById
Lookup(User)
Last Modified By
LastModifiedById
Lookup(User)
Owner
OwnerId
Lookup(User,Group)
Price
Price__c
Currency(18, 0)
Status
Status__c
Picklist
Stock Quantity
Stock_Quantity__c
Number(18, 0)
Vehicle Dealer
Vehicle_Dealer__c
Lookup(Vehicle Dealer)
Vehicle Model
Vehicle_Model__c
Picklist
Vehicle Name
Name
Text(80)

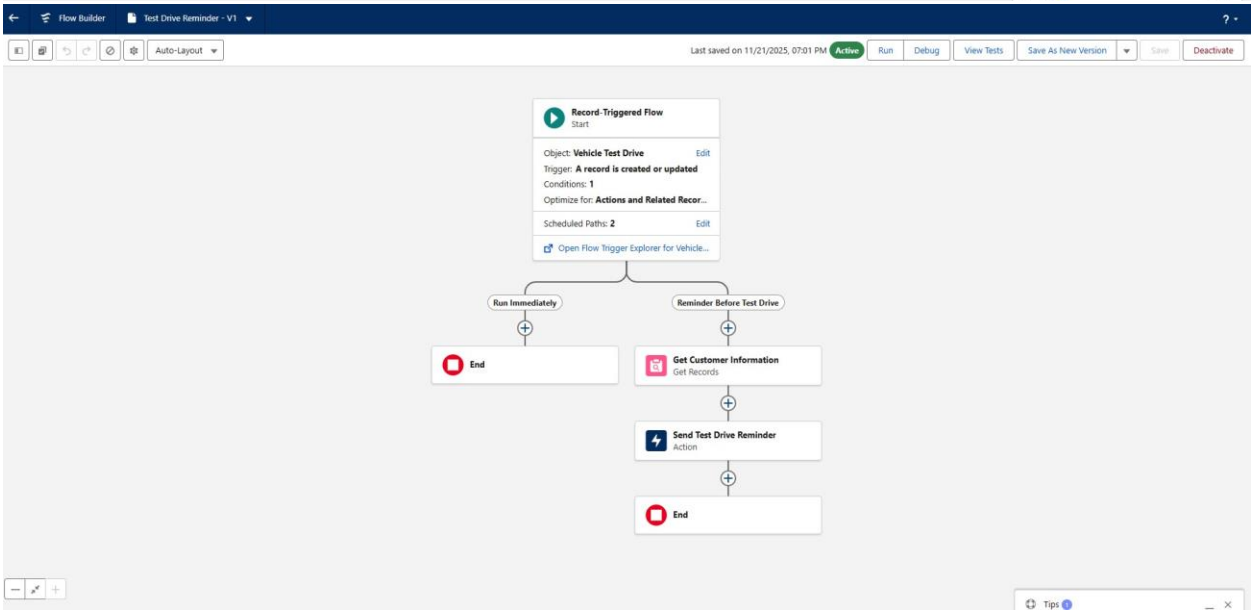

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Vehicle Order
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers
Validation Rules
Fields & Relationships
9 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Assigned Dealer
Assigned_Dealer__c
Lookup(Vehicle Dealer)
Created By
CreatedById
Lookup(User)
Last Modified By
LastModifiedById
Lookup(User)
Order Date
Order_Date__c
Date
Owner
OwnerId
Lookup(User,Group)
Status
Status__c
Picklist
Vehicle
Vehicle__c
Lookup(Vehicle)
Vehicle Customer
Vehicle_Customer__c
Lookup(Vehicle Customer)
Vehicle Order Number
Name
Auto Number

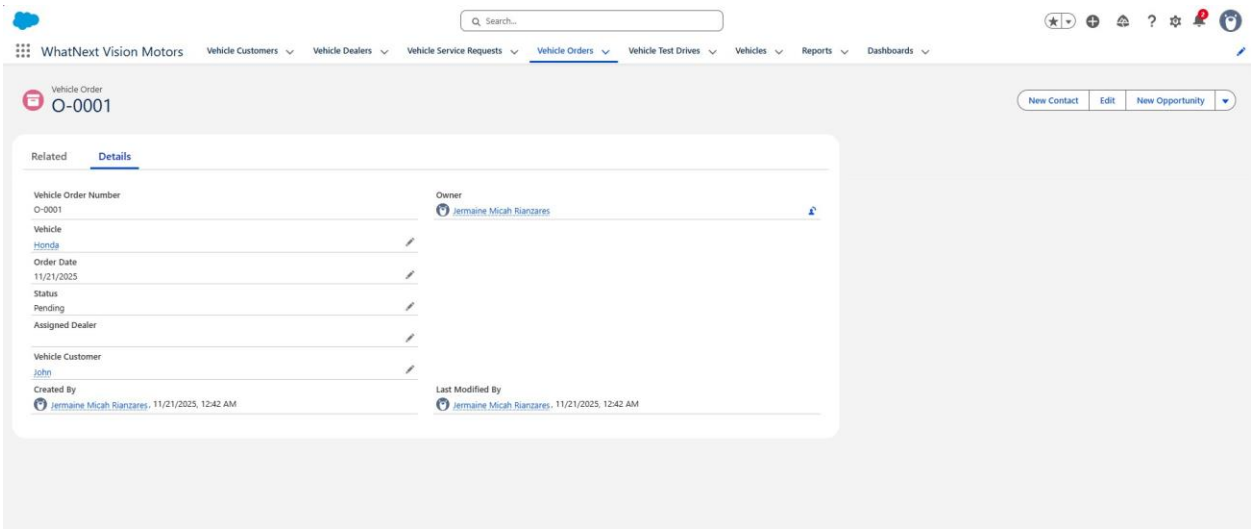

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Vehicle Service Request
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers
Validation Rules
Fields & Relationships
9 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Created By
CreatedById
Lookup(User)
Customer
Customer__c
Lookup(Customer)
Issue Description
Issue_Description__c
Text(60)
Last Modified By
LastModifiedById
Lookup(User)
Owner
OwnerId
Lookup(User,Group)
Service Date
Service_Date__c
Date
Status
Status__c
Picklist
Vehicle
Vehicle__c
Lookup(Vehicle)
Vehicle Service Request Name
Name
Text(80)


Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Vehicle Test Drive
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers
Validation Rules
Fields & Relationships
8 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Created By
CreatedById
Lookup(User)
Last Modified By
LastModifiedById
Lookup(User)
Owner
OwnerId
Lookup(User,Group)
Status
Status__c
Picklist
Test Drive Date
Test_Drive_Date__c
Date
Vehicle
Vehicle__c
Lookup(Vehicle)
Vehicle Customer
Vehicle_Customer__c
Lookup(Vehicle Customer)
Vehicle Test Drive Name
Name
Text(80)

Automation

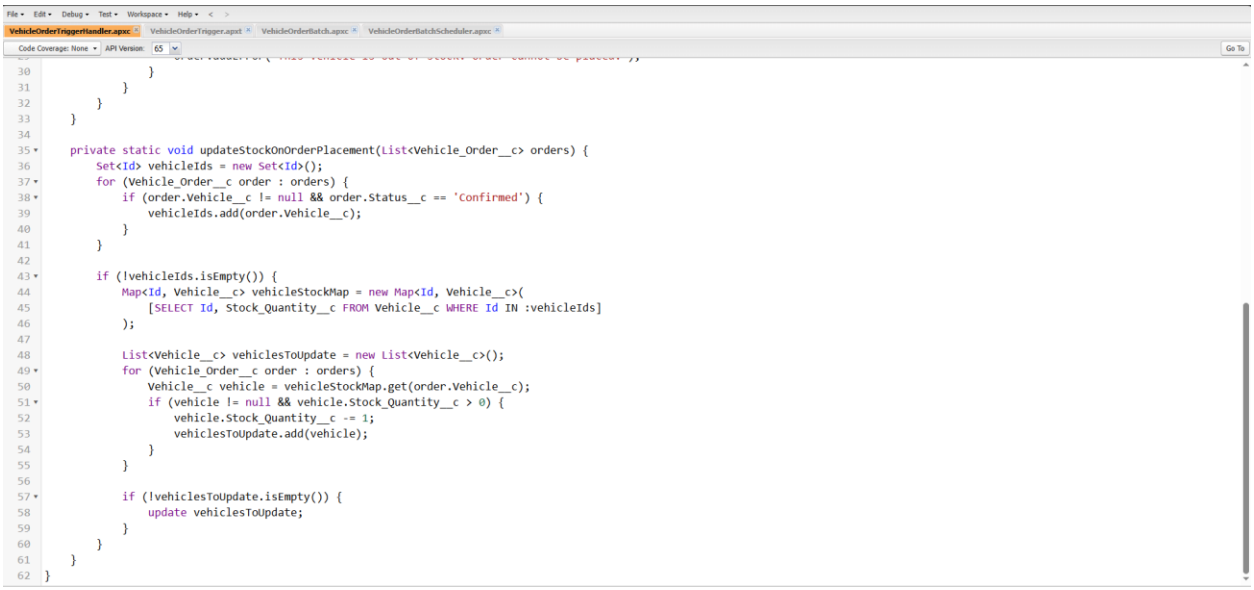

Flow Builder
Auto Assign Dealer - V1
Auto-Layout
Last saved on 11/21/2025, 04:28 PM
Active
Run
Debug
View Tests
Save As New Version
Save
Deactivate
Record-Triggered Flow
Start
Object: Vehicle Order
Edit
Trigger: A record is created
Conditions: 1
Optimize for: Actions and Related Recor...
Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Vehicle...
Run Immediately
Get Customer Information
Get Records
Get Nearest Dealer
Get Records
Assign Dealer to Order
Update Records
End
Tips


Flow Builder
Test Drive Reminder - V1
Auto-Layout
Last saved on 11/21/2025, 07:01 PM
Active
Run
Debug
View Tests
Save As New Version
Save
Deactivate
Record-Triggered Flow
Start
Object: Vehicle Test Drive
Edit
Trigger: A record is created or updated
Conditions: 1
Optimize for: Actions and Related Recor...
Scheduled Paths: 2
Edit
Open Flow Trigger Explorer for Vehicle...
Run Immediately
Reminder Before Test Drive
End
Get Customer Information
Get Records
Send Test Drive Reminder
Action
End
Tips

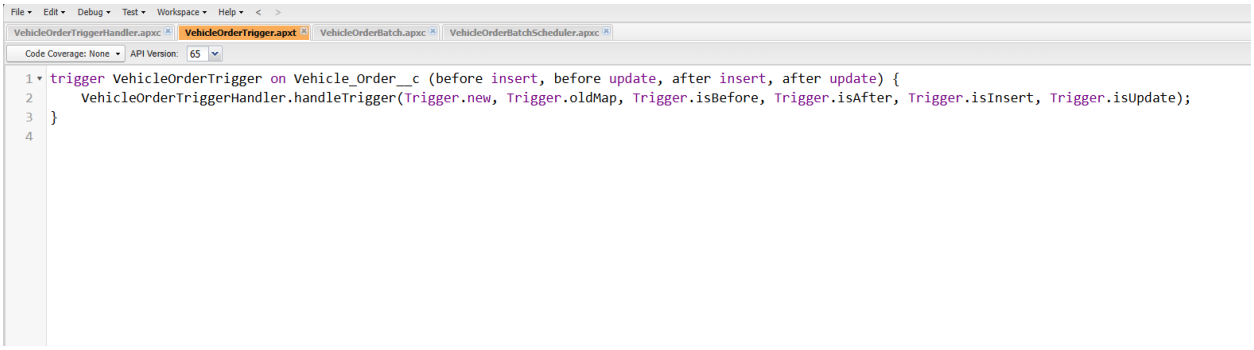


Apex and Batch Class

```apex
public class VehicleOrderTriggerHandler {

    public static void handleTrigger(List<Vehicle_Order__c> newOrders, Map<Id, Vehicle_Order__c> oldOrders, Boolean isBefore, Boolean isAfter, Boolean isInsert, Boolean isUpdate) {
        if (isBefore && (isInsert || isUpdate)) {
            preventOrderIfOutOfStock(newOrders);
        }

        if (isAfter && (isInsert || isUpdate)) {
            updateStockOnOrderPlacement(newOrders);
        }
    }

    private static void preventOrderIfOutOfStock(List<Vehicle_Order__c> orders) {
        Set<Id> vehicleIds = new Set<Id>();
        for (Vehicle_Order__c order : orders) {
            if (order.Vehicle__c != null) {
                vehicleIds.add(order.Vehicle__c);
            }
        }

        if (!vehicleIds.isEmpty()) {
            Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>(
                [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
            );

            for (Vehicle_Order__c order : orders) {
                Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
                if (vehicle != null && vehicle.Stock_Quantity__c <= 0) {
                    order.addError('This vehicle is out of stock. Order cannot be placed.');
                }
            }
        }
    }
```



```
                }
            }
        }
    }

    private static void updateStockOnOrderPlacement(List<Vehicle_Order__c> orders) {
        Set<Id> vehicleIds = new Set<Id>();
        for (Vehicle_Order__c order : orders) {
            if (order.Vehicle__c != null && order.Status__c == 'Confirmed') {
                vehicleIds.add(order.Vehicle__c);
            }
        }

        if (!vehicleIds.isEmpty()) {
            Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>(
                [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
            );

            List<Vehicle__c> vehiclesToUpdate = new List<Vehicle__c>();
            for (Vehicle_Order__c order : orders) {
                Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
                if (vehicle != null && vehicle.Stock_Quantity__c > 0) {
                    vehicle.Stock_Quantity__c -= 1;
                    vehiclesToUpdate.add(vehicle);
                }
            }

            if (!vehiclesToUpdate.isEmpty()) {
                update vehiclesToUpdate;
            }
        }
    }
}
```



```
trigger VehicleOrderTrigger on Vehicle_Order__c (before insert, before update, after insert, after update) {
    VehicleOrderTriggerHandler.handleTrigger(Trigger.new, Trigger.oldMap, Trigger.isBefore, Trigger.isAfter, Trigger.isInsert, Trigger.isUpdate);
}
```

File Edit Debug Test Workspace Help < >

VehicleOrderTriggerHandler.apxc | VehicleOrderTrigger.apxt | VehicleOrderBatch.apxc | VehicleOrderBatchScheduler.apxc

Code Coverage: None | API Version: 65

```apex
global class VehicleOrderBatch implements Database.Batchable<sObject> {

    global Database.QueryLocator start(Database.BatchableContext bc) {
        return Database.getQueryLocator([
            SELECT Id, Status__c, Vehicle__c FROM Vehicle_Order__c WHERE Status__c = 'Pending'
        ]);
    }

    global void execute(Database.BatchableContext bc, List<Vehicle_Order__c> orderList) {
        Set<Id> vehicleIds = new Set<Id>();
        for (Vehicle_Order__c order : orderList) {
            if (order.Vehicle__c != null) {
                vehicleIds.add(order.Vehicle__c);
            }
        }

        if (!vehicleIds.isEmpty()) {
            Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>(
                [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
            );

            List<Vehicle_Order__c> ordersToUpdate = new List<Vehicle_Order__c>();
            List<Vehicle__c> vehiclesToUpdate = new List<Vehicle__c>();

            for (Vehicle_Order__c order : orderList) {
                Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
                if (vehicle != null && vehicle.Stock_Quantity__c > 0) {
                    order.Status__c = 'Confirmed';
                    vehicle.Stock_Quantity__c -= 1;
                    ordersToUpdate.add(order);
                    vehiclesToUpdate.add(vehicle);
                }
            }
```

File Edit Debug Test Workspace Help < >

VehicleOrderTriggerHandler.apxc | VehicleOrderTrigger.apxt | VehicleOrderBatch.apxc | VehicleOrderBatchScheduler.apxc

Code Coverage: None | API Version: 65

```
        for (Vehicle_Order__c order : orderList) {
            if (order.Vehicle__c != null) {
                vehicleIds.add(order.Vehicle__c);
            }
        }

        if (!vehicleIds.isEmpty()) {
            Map<Id, Vehicle__c> vehicleStockMap = new Map<Id, Vehicle__c>(
                [SELECT Id, Stock_Quantity__c FROM Vehicle__c WHERE Id IN :vehicleIds]
            );

            List<Vehicle_Order__c> ordersToUpdate = new List<Vehicle_Order__c>();
            List<Vehicle__c> vehiclesToUpdate = new List<Vehicle__c>();

            for (Vehicle_Order__c order : orderList) {
                Vehicle__c vehicle = vehicleStockMap.get(order.Vehicle__c);
                if (vehicle != null && vehicle.Stock_Quantity__c > 0) {
                    order.Status__c = 'Confirmed';
                    vehicle.Stock_Quantity__c -= 1;
                    ordersToUpdate.add(order);
                    vehiclesToUpdate.add(vehicle);
                }
            }

            if (!ordersToUpdate.isEmpty()) update ordersToUpdate;
            if (!vehiclesToUpdate.isEmpty()) update vehiclesToUpdate;
        }
    }

    global void finish(Database.BatchableContext bc) {
        System.debug('Vehicle order batch job completed.');
    }
}
```

File Edit Debug Test Workspace Help < >

VehicleOrderTriggerHandler.apxc | VehicleOrderTrigger.apxt | VehicleOrderBatch.apxc | VehicleOrderBatchScheduler.apxc

Code Coverage: None | API Version: 65

```
global class VehicleOrderBatchScheduler implements Schedulable {
    global void execute(SchedulableContext sc) {
        VehicleOrderBatch batchJob = new VehicleOrderBatch();
        Database.executeBatch(batchJob, 50); // 50 = batch size
    }
}
```