

Truck

1. Apex Classes

TruckController.cls

```
public with sharing class TruckController {

    @AuraEnabled

    public static List<Truck__c> getTrucksByBrand(String brand) {

        return [SELECT Id,Truck_Id__c,Name,Brand__c, Cost_of_Booking__c,
Availability_of_Truck__c,Maximum_Load__c FROM Truck__c WHERE Brand__c = :brand
AND Availability_of_Truck__c = true];

    }

}
```

MyOrder.cls

```
public with sharing class MyOrder {

    @AuraEnabled(cacheable=true)

    public static List<Truck_Booking__c> main() {

        return [SELECT Id,Name,(SELECT Id FROM Paymentss__r LIMIT
1),Truck_Name__r.Brand__c,Customer_s_Load__c,Date_Of_Booking__c,Drop_Address_
__c,Payment_Type__c,Pickup_Address__c,Status__c,Truck_Name__c FROM
Truck_Booking__c WHERE CreatedById = :UserInfo.getUserId() ORDER BY
Date_of_Booking__c DESC];

    }

}
```

2. Parent Component

home1.html

<!--

@description :

@author : ChangeMeIn@UserSettingsUnder.SFDoc

@group :

@last modified on : 11-06-2025

@last modified by : ChangeMeIn@UserSettingsUnder.SFDoc

-->

<template>

<template if:false={selectedBrand}>

<lightning-card lwc:if={s1}>

<div class="slds-grid slds-wrap">

<div

class="slds-col slds-size_1-of-2 slds-grid slds-grid_vertical slds-grid_align-center slds-grid_vertical-align-center">

<h1 class="slds-text-heading_medium">Quick Actions</h1>

<lightning-button label="Book a Truck" onclick={handleClick}
name="createBooking" variant="brand"

class="slds-p-top_small">

</lightning-button>

<lightning-button variant="brand" label="My Orders" onclick={handleClick}
name="myOrders"></lightning-button>

</div>

<div class="slds-col slds-size_1-of-2">

<lightning-card title="Truck Brands">

<lightning-carousel>

<template for:each={brands} for:item="brand">

<lightning-carousel-image key={brand.value} src={brand.image}
header={brand.label}

```

        description="Click to view trucks" onclick={handleBrandClick}
        data-brand={brand.value}>
    </lightning-carousel-image>
</template>
</lightning-carousel>
</lightning-card>
</div>
</div>
</lightning-card>
<template if:true={s}>
    <lightning-card title="Brands Available">
        <div class="slds-grid slds-wrap a">

            <div class="slds-col slds-size_6-of-12 slds-p-around_small brand-cards">
                <h3>Mahindra</h3>
                <img src={mah} onclick={handleBrandClick} data-brand="Mahindra">
            </div>

            <div class="slds-col slds-size_6-of-12 slds-p-around_small brand-cards">
                <h3>TataMotors</h3><img src={tata} onclick={handleBrandClick} data-
brand="TataMotors">
            </div>

            <div class="slds-col slds-size_6-of-12 slds-p-around_small brand-cards">
                <h3>Ashok Leyland</h3><img src={ashok} onclick={handleBrandClick}
data-brand="AshokLeyland">
            </div>

            <div class="slds-col slds-size_6-of-12 slds-p-around_small brand-cards">
                <h3>Eicher</h3><img src={eicher} onclick={handleBrandClick} data-
brand="Eicher">
            </div>

```

```

        </div>

    </lightning-card>

</template>

</template>

<!-- Conditionally render the list component -->

<template if:true={selectedBrand}>

    <c-booking1 brand={selectedBrand} onback={handleBack}></c-booking1>

</template>

</template>

```

home1.js

```

import {
    __ LightningElement,
    __ track
} from 'lwc';

import mahindraLogo from '@salesforce/resourceUrl/Mahindra';
import TataLogo from '@salesforce/resourceUrl/TataMotors';
import AshokLogo from '@salesforce/resourceUrl/AshokLeyland';
import EicherLogo from '@salesforce/resourceUrl/Eicher';

import {
    __ NavigationMixin
} from 'lightning/navigation';

-

export default class Home1 extends NavigationMixin(LightningElement) {
    __ @track selectedBrand;
    __ mah = mahindraLogo;

```

```

    tata = TataLogo;

    ashok = AshokLogo;

    eicher = EicherLogo;

    s

    s1 = true

    brands = [{

        label: 'Tata Motor',

        value: 'TataMotors',

        image: this.tata

    },

    {

        label: 'Ashok Leyland',

        value: 'AshokLeyland',

        image: this.ashok

    },

    {

        label: 'Mahindra',

        value: 'Mahindra',

        image: this.mah

    },

    {

        label: 'Eicher',

        value: 'Eicher',

        image: this.eicher

    }

    ];

    -

    handleBrandClick(event) {

```

```

    this.selectedBrand = event.currentTarget.dataset.brand;
  }
  handleBack() {
    this.selectedBrand = null;
  }
  handleClick(e) {
    let act = e.target.name
    if (act == 'createBooking') {
      this.s = true
      this.s1 = false
    } else if (act == 'myOrders') {
      this[NavigationMixin.Navigate]({
        type: 'standard_webPage',
        attributes: {
          url: '/truckkkkkkkk'
        }
      })
    }
  }
}

```

home1.css

```

.a img{
  width: 100%;
  height: 300px;
  border-radius: 10px;
}

```

```
.brand-cards {  
  position: relative  
}
```

```
.brand-cards h3 {  
  display: inline-block;  
  position: absolute;  
  bottom: 20px;  
  left: 150px;  
  background-color: rgba(255,255,255,0.7);  
  padding: 5px;  
  border-radius: 5px  
}
```

3. Child Component

booking1.html

<!--

@description :

@author : ChangeMeIn@UserSettingsUnder.SFDoc

@group :

@last modified on : 11-06-2025

@last modified by : ChangeMeIn@UserSettingsUnder.SFDoc

-->

<template>

<lightning-card title={brandTitle} lwc:if={showTrucks}>

<lightning-button label="Previous" onclick={goBack} icon-name="utility:back" class="slds-m-bottom_small">

</lightning-button>

<template if:true={trucks}>

<div class="slds-grid slds-wrap">

<template for:each={trucks} for:item="truck">

<div class="slds-col slds-size_1-of-2 slds-p-around_small" key={truck.Id}>

<div class="slds-grid slds-wrap">

<div class="slds-col slds-size_1-of-2">

<h3>Truck Name: {truck.Name}</h3>

<h3>Cost of Booking: {truck.Cost_of_Booking_c}</h3>

<h3>Maximum Load: {truck.Maximum_Load_c}</h3>

</div>

<div class="slds-col slds-size_1-of-2">

</div>

</div>

<div class="slds-grid slds-grid_align-end">

<div class="slds-col">

<lightning-button variant="brand" label="Add to Cart" onclick={handleClick} name="add"

data-id={truck.Id}></lightning-button>

</div>

<div class="slds-col slds-p-left_x-small">

<lightning-button variant="brand" label="Book" onclick={handleClick} name="book"


```
_____ data-id={truck.Id}></lightning-button>
_____
_____ </div>
_____
_____ </div>
_____
_____ </div>
_____
_____ </template>
_____
_____ </div>
_____
_____ </template>
_____
_____ <template if:true={error}>
_____ <p>Error loading trucks: {error.message}</p>
_____
_____ </template>
_____
_____ </lightning-card>
_____
_____ <lightning-card title="My Details" lwc:if={bookcon}>
_____
_____ <template lwc:if={recordId}>
_____
_____ <lightning-record-edit-form object-api-name="Contact" record-id={recordId}
onsuccess={handleSuccess}>
_____
_____ <lightning-messages></lightning-messages>
_____
_____ <lightning-input-field field-name='FirstName'></lightning-input-field>
_____
_____ <lightning-input-field field-name='LastName'></lightning-input-field>
_____
_____ <lightning-input-field field-name='Email'></lightning-input-field>
_____
_____ <lightning-input-field field-name='Phone'></lightning-input-field>
_____
_____ <lightning-button variant="brand" label="Back" onclick={handleClick}
name="back"></lightning-button>
_____
_____ <lightning-button variant="brand" label="Next" name="save"
type="submit">
_____
_____ </lightning-button>
_____
_____ </lightning-record-edit-form>
_____
_____ </template>
_____
_____ </lightning-card>
_____
_____ <lightning-card lwc:if={s2} title="Booking Details">
```

```

    <lightning-record-edit-form object-api-name="Truck_Booking__c"
onsuccess={handleBookingSuccess}>

    <lightning-input-field field-name="Date_Of_Booking__c"></lightning-input-
field>

    <lightning-input-field field-name="Truck_Name__c" value={selectedTruckId}
class="slds-hide">

    </lightning-input-field>

    <lightning-input-field field-name="Customer_s_Load__c"></lightning-input-
field>

    <lightning-input-field field-name="Payment_Type__c"></lightning-input-field>

    <lightning-input-field field-name="Pickup_State__c"></lightning-input-field>

    <lightning-input-field field-name="Pickup_City__c"></lightning-input-field>

    <lightning-input-field field-name="Pickup_Area__c"></lightning-input-field>

    <lightning-input-field field-name="Drop_State__c"></lightning-input-field>

    <lightning-input-field field-name="Drop_City__c"></lightning-input-field>

    <lightning-input-field field-name="Drop_Area__c"></lightning-input-field>

    <lightning-button variant="brand" label="Back" onclick={handleClick}
name="back1"></lightning-button>

    <lightning-button variant="brand" label="Save" type="submit"
name="save1"></lightning-button>

</lightning-record-edit-form>

</lightning-card>

<!-- <lightning-card lwc:if={s3}> -->

<lightning-card >

    <div class="slds-grid slds-wrap">

        <div

            class="slds-col slds-size_1-of-2 slds-p-around_medium slds-grid slds-
grid_vertical-align-center slds-grid_align-center">

            <img src={tImg}>

        </div>

```

```

<div class="slds-col slds-size 1-of-2">
  <div class="slds-m-top_xx-large slds-text-align_center">
    <h1>
      <b>
        Booking confirmed! We've successfully created your order and it's now
        visible in your my
        orders page. We're excited to serve you!
      </b>
    </h1>
  </div>
  <div class="slds-grid slds-grid_align-end slds-p-around_large">
    <lightning-button variant="brand" label="My Orders"
    onclick={handleClick} name="ord">
      </lightning-button>
    </div>
  </div>
</div>
</lightning-card>
</template>

```

booking1.js

```

import {
  LightningElement,
  api,
  wire
} from 'lwc';

```

```

import getTrucksByBrand from
 '@salesforce/apex/TruckController.getTrucksByBrand';

import Mahindra from '@salesforce/resourceUrl/Mahindra';

import Tata from '@salesforce/resourceUrl/TataMotors';

import Ashok_Leyland from '@salesforce/resourceUrl/AshokLeyland';

import Eicher from '@salesforce/resourceUrl/Eicher';

import truckImg from '@salesforce/resourceUrl/happyTruck';

import {
    getRecord,
    getFieldValue
} from 'lightning/uiRecordApi';

import USER_ID from '@salesforce/user/Id';

import CONTACT_ID from '@salesforce/schema/User.ContactId';

import {
    ShowToastEvent
} from 'lightning/platformShowToastEvent';

import {
    NavigationMixin
} from 'lightning/navigation';

export default class Booking1 extends NavigationMixin(LightningElement) {

    @api brand

    trucks = []

    brandTitle = ''

    brandImage

    tImg = truckImg

    showTrucks = true

    bookcon

    recordId

```

```
selectedTruckId
detailcon = true
s2 = false
s3
@wire(getRecord, {
    recordId: USER_ID,
    fields: [CONTACT_ID]
})
wired({
    data,
    error
}) {
    if (data) {
        this.recordId = getFieldValue(data, CONTACT_ID);
    }
}
connectedCallback() {
    this.brandTitle = `${this.brand} Trucks`;
    this.setBrandImage(this.brand);
    this.fetchTrucks();
}
setBrandImage(brand) {
    switch (brand) {
        case 'Mahindra':
            this.brandImage = Mahindra;
            break;
        case 'Tata':
        case 'TataMotors':
```

```

        this.brandImage = Tata;

        break;
// case 'Ashok_Leyland':
case 'AshokLeyland':

    this.brandImage = Ashok_Leyland;

    break;

case 'Eicher':

    this.brandImage = Eicher;

    break;

default:

    this.brandImage = "";

    }
}

fetchTrucks() {

    getTrucksByBrand({

        brand: this.brand

    })

    .then(result => {

        this.trucks = result;

        this.error = undefined;

    })

}

handleClick(event) {

    const action = event.target.name;

    const truckId = event.target.dataset.id;

    if (action == 'book') {

        this.selectedTruckId = truckId

        this.bookcon = true

```

```
        this.showTrucks = false
    } else if (action == 'back') {
        this.showTrucks = true
        this.bookcon = false
    } else if (action == 'back1') {
        this.bookcon = true
        this.s2 = false
    } else if (action == 'ord') {
        this[NavigationMixin.Navigate]({
            type: 'standard__webPage',
            attributes: {
                url: '/truckkkkkkk'
            }
        })
    }
}

goBack() {
    const backEvent = new CustomEvent('back');
    this.dispatchEvent(backEvent);
}

handleSuccess() {
    this.showTrucks = false
    this.bookcon = false
    this.s2 = true
}

handleBookingSuccess() {
    this.s2 = false
    this.s3 = true
}
```

```

const toastEvent = new ShowToastEvent({
    title: 'Success',
    message: 'Truck Booked Successfully',
    variant: 'success'
})

this.dispatchEvent(toastEvent);
}
}

```

4. For My Orders

myOrder.html

```
<!--
```

```
@description    :
```

```
@author        : ChangeMeIn@UserSettingsUnder.SFDoc
```

```
@group         :
```

```
@last modified on : 11-05-2025
```

```
@last modified by : ChangeMeIn@UserSettingsUnder.SFDoc
```

```
-->
```

```
<template>
```

```
<lightning-card title="My Orders">
```

```
<div class="slds-grid slds-wrap slds-p-around_medium">
```

```
<template for:each={bookings} for:item="book">
```

```
<div class="slds-col slds-size_1-of-1 slds-p-around_small slds-box slds-m-
bottom_medium" key={book.Id}>
```

```
<div class="slds-grid slds-wrap">
```

```
<div class="slds-col slds-size_1-of-3 slds-p-around_small">
```

```
<template if:true={book.brandLogo}>
```



```

        <img src={book.brandLogo} />

    </template>

</div>

<div class="slds-col slds-size_1-of-3 slds-p-around_small">

    <h1><b>Booking Name:</b> {book.Name}</h1>

    <h1><b>Brand:</b> {book.brandName}</h1>

    <h1><b>Date of Delivery:</b> {book.fDate}</h1>

    <h1><b>Pickup Address:</b> {book.Pickup_Address__c}</h1>

    <h1><b>Drop Address:</b> {book.Drop_Address__c}</h1>

    <h1><b>Load:</b> {book.Customer_s_Load__c}</h1>

    <h1><b>Delivery Type:</b> {book.Delivery_Type__c}</h1>

    <h1><b>Status:</b> {book.Status__c}</h1>

</div>

<div class="slds-col slds-size_1-of-3 slds-p-around_small">

    <div class="slds-grid slds-grid_vertical-align-center slds-grid_align-
center slds-wrap">

        <template if:true={book.isCompleted}>

            <div class="slds-col slds-size_1-of-2 slds-p-around_medium">

                <lightning-button label="Feedback" onclick={handleClick}
variant="brand" name="a">

                    </lightning-button>

                </div>

            </template>

            <div class="slds-col slds-size_1-of-2 slds-p-around_medium">

                <lightning-button label="View" onclick={handleClick}
variant="brand" name="b" data-id={book.Id}>

                    </lightning-button>

                </div>

            <div class="slds-col slds-size_1-of-2 slds-p-around_medium">

```

```

        <lightning-button label="Payment Details" onclick={handleClick}
variant="brand" name="c" data-id={book.paymentId}>

        </lightning-button>

    </div>

    <div class="slds-col slds-size_1-of-2 slds-p-around_medium">

        <lightning-button label="Invoice" onclick={handleClick}
variant="brand" name="d">

        </lightning-button>

    </div>

</div>

</div>

</div>

<!-- <div class="slds-col slds-size_1-of-3">

    <h1>Hiiiiii</h1>

</div> -->

</div>

</div>

</template>

</div>

</lightning-card>

</template>

```

myOrder.js

```

import {
    LightningElement,
    wire
} from 'lwc';

import mahindraLogo from '@salesforce/resourceUrl/Mahindra';
import TataLogo from '@salesforce/resourceUrl/TataMotors';
import AshokLogo from '@salesforce/resourceUrl/AshokLeyland';

```

```

import EicherLogo from '@salesforce/resourceUrl/Eicher';

import cls from '@salesforce/apex/MyOrder.main';

import {
    NavigationMixin
} from 'lightning/navigation';

export default class MyOrder extends NavigationMixin(LightningElement) {

    brandLogos = {

        'Mahindra': mahindraLogo,

        'Tata_Motors': TataLogo,

        'Ashok_Leyland': AshokLogo,

        'Eicher': EicherLogo

    }

    bookings = []

    @wire(cls) wired({

        data,

        error

    }) {

        if (data) {

            this.bookings = data.map(book => ({

                ...book,

                brandLogo: this.brandLogos[book.Truck_Name__r.Brand__c] || null,

                brandName: book.Truck_Name__r.Brand__c || 'Unknown',

                fDate: new Date(book.Date_Of_Booking__c).toLocaleDateString('en-IN'),

                isCompleted: book.Status__c === 'Completed',

                paymentId: book.Paymentss__r?.[0].Id

            }));

        }

    }

}

```

```
handleClick(e) {  
  let n = e.target.name  
  if (n == 'a') {  
    this[NavigationMixin.Navigate]({  
      type: 'standard__webPage',  
      attributes: {  
        url: 'https://forms.gle/ZXajUCoRYjvZqHjS6'  
      }  
    })  
  } else if (n == 'b') {  
    this[NavigationMixin.Navigate]({  
      type: 'standard__recordPage',  
      attributes: {  
        recordId: e.target.dataset.id,  
        objectApiName: 'Truck_Booking__c',  
        actionName: 'view'  
      }  
    })  
  } else if (n == 'c') {  
    this[NavigationMixin.Navigate]({  
      type: 'standard__recordPage',  
      attributes: {  
        recordId: e.target.dataset.id,  
        objectApiName: 'Payments__c',  
        actionName: 'view'  
      }  
    })  
  } else if (n == 'd') {
```

}

}

}