

# ASP.NET Core MVC 6.0

Education and Training Solutions 2023

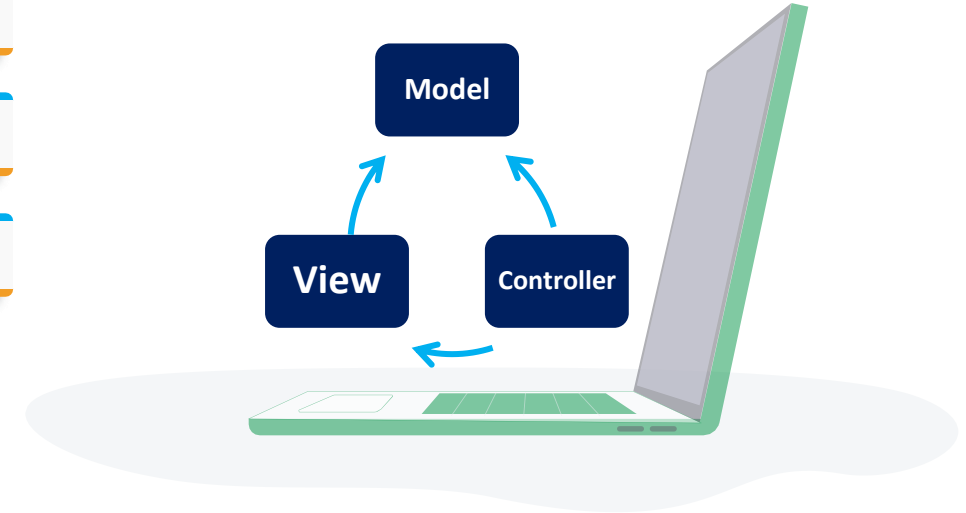


## Chapter 1

1 Search

2 DataTables

3 Reports





Search

Create **Search** Action in Admin Controller

```
[HttpGet]
public IActionResult Search()
{
    var modelContext =
        _context.ProductCustomers.Include(p =>
            p.Customer).Include(p => p.Product).ToList();
    return View(modelContext);
}
```



Create **Search** Action in Admin Controller

```
[HttpPost]
public async Task<IActionResult> Search(DateTime?
stratDate, DateTime? endDate)
{

var modelContext =
_context.ProductCustomers.Include(p =>
p.Customer).Include(p => p.Product);

if (stratDate == null && endDate == null)
{
return View(modelContext);
}
```



Create **Search** Action in Admin Controller

```
}  
else if (stratDate == null && endDate != null)  
{  
  
var result = await modelContext.Where(x =>  
x.DateFrom.Value.Date == endDate).ToListAsync();  
return View(result);  
}  
  
else if (stratDate != null && endDate == null)  
{
```



Create **Search** Action in Admin Controller

```
var result = await modelContext.Where(x =>
x.DateFrom.Value.Date == stratDate).ToListAsync();
return View(result);
}
else
{
var result = await modelContext.Where(x => x.DateFrom
>= stratDate && x.DateFrom <= endDate).ToListAsync();
return View(result);
}
}
```



## Create a Search view

×

Add Razor View

View name

Search

Template

Empty (without model) ▾

Model class

▾

Data context class

ModelContext (TheLearningHub\_Resturant.Models) ▾

Options

☐ Create as a partial view

☐ Reference script libraries

☒ Use a layout page

~/Views/Shared/\_AdminLayout.cshtml

...

(Leave empty if it is set in a Razor \_viewstart file)

Add

Cancel



In the Search view create a form that searches between two dates and a table to show the result

```
@model IEnumerable<ProductCustomer>
@{
    ViewData["Title"] = "Search";
    Layout = "~/Views/Shared/_AdminLayout.cshtml";
}
<div class="row">
<div class="col-6">
<form action="Search" method="post">
<input type="date" class="form-control"
name="stratDate" placeholder="Start Date">
```



In the Search view create a form that searches between two dates and a table to show the result

```
<input type="date" class="form-control"
name="endDate" placeholder="End Date">

<button type="submit" class="btn btn-
dark">Search</button>

</form>
</div>
</div>
<div class="row">
<div class="col-10">
```



In the Search view create a form that searches between two dates and a table to show the result

```
<table class="table example">
<thead>
<tr>
<th>Customer First Name </th>
<th>Customer Last Name </th>
<th>Product Name</th>
<th>Quantity</th>
<th>Price</th>
<th>Date From</th>
<th>Date To</th>
</tr>
</thead>
```



In the Search view create a form that searches between two dates and a table to show the result

```
<tbody>
@foreach (var item in Model)
{
    <tr>
    <td>
        @Html.DisplayFor(modelItem => item.Customer.Fname)
    </td>
    <td>
        @Html.DisplayFor(modelItem => item.Customer.Lname)
    </td>
}
```



In the Search view create a form that searches between two dates and a table to show the result

```
<td>
@Html.DisplayFor(modelItem => item.Product.Nameee)
</td>
<td>
@Html.DisplayFor(modelItem => item.Quantity)
</td>
<td>
@(item.Quantity * item.Product.Price)
</td>
```



In the Search view create a form that searches between two dates and a table to show the result

```
<td>
@Html.DisplayFor(modelItem => item.DateFrom)
</td>
<td>
@Html.DisplayFor(modelItem => item.DateTo)
</td>
</tr>
}
</tbody>
</table>
</div>
</div>
```



## The Result

Search

Customer First Name	Customer Last Name	Product Name	Quantity	Price	Date From	Date To
dana	ahmad	chicken	10.00	150	3/1/2022 12:18:00 AM	3/3/2022 12:18:08 AM
Bayan	Mohammad	Burger	5.00	30	5/4/2023 12:40:11 PM	5/6/2023 12:40:16 PM

# DataTables



**DataTables** is a plug-in for the jQuery Javascript library. It's a powerful tool that adds a lot of advanced features to any HTML table such as Pagination, Instant search, and Multi-column ordering

In **JoinTable** view add the attribute Id = “example1” in the start tag of the table

```
<div class="table-responsive">
  <table id="example1" class="display expandable-table" style="width:100%">
    <thead>
      <tr>
        <th>Name</th>
        <th>Image</th>
        <th>Category</th>
        <th>Product</th>
        <th>Price</th>
      </tr>
    </thead>
    <tbody>
      @foreach (var item in Model)
      {
```

Add the following link to the top of the Join Table view

```
<link  
href="https://cdn.datatables.net/1.11.1/css/jquery.dataTables.min.css" rel="stylesheet" />  
<link  
href="https://cdn.datatables.net/buttons/2.0.0/css/buttons.dataTables.min.css" rel="stylesheet" />
```



Add the following Scripts to the end of the JoinTable view

```
<script src="https://code.jquery.com/jquery-3.5.1.js"></script>
<script
src="https://cdn.datatables.net/1.11.1/js/jquery.dataTables.min.js" defer></script>

<script
src="https://cdn.datatables.net/buttons/2.0.0/js/dataTables.buttons.min.js" defer></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
```



Add the following Scripts to the end of the JoinTable view

```
<script  
src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1  
.53/pdfmake.min.js"></script>  
<script  
src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1  
.53/vfs_fonts.js"></script>  
<script  
src="https://cdn.datatables.net/buttons/2.0.0/js/button  
s.html5.min.js" defer></script>
```



Add the following Scripts to the end of the JoinTable view



```
<script>
$(document).ready(function () {
$('#example1').DataTable({
  dom: 'Bfrtip',
  buttons: [
    'copyHtml5',
    'excelHtml5',
    'csvHtml5',
    'pdfHtml5'
  ]
}); });
</script>
```



## The Result

### JoinTables

#### Customers

Copy	Excel	CSV	PDF	Search:	<input type="text"/>
First Name	Last Name	Prodcut Name	Price	Category Name	Image
dana	ahmad	chicken	15	Fast Food	
Bayan	Mohammad	Burger	6	Fast Food	

Showing 1 to 2 of 2 entries

Previous **1** Next

# Reports



## Exercise

Create a View called Report that Contains what you learned in the previous two lectures.

Hint: Search, Tuple, DataTable, Join

Create **Report** action in the Admin controller.

```
[HttpGet]
public IActionResult Report()
{
    var customer = _context.Customers.ToList();
    var procustomers =
        _context.ProductCustomers.ToList();
    var products = _context.Products.ToList();
    var catecoreys = _context.Categories.ToList();
    var multiTable = from c in customer
        join pc in procustomers on c.Id equals pc.CustomerId
```



Create **Report** action in the Admin controller.

```
join p in products on pc.ProductId equals p.Id
join cat in categorieys on p.CategoryId equals cat.Id
select new JoinTables { Product = p, Customer = c,
ProductCustomer = pc, Category = cat };

var modelContext =
_context.ProductCustomers.Include(p =>
p.Customer).Include(p => p.Product).ToList();
ViewBag.TotalQuantity = modelContext.Sum(x =>
x.Quantity);
}
```



Create **Report** action in the Admin controller.

```
ViewBag.TotalPrice = modelContext.Sum(x =>  
x.Product.Price * x.Quantity);  
var model3 = Tuple.Create<IEnumerable<JoinTables>,  
IEnumerable<ProductCustomer>>(multiTable,  
modelContext);  
return View(model3);
```



Create **Report** action in the Admin controller.

```
[HttpPost]
public async Task<IActionResult> Report(DateTime?
stratDate, DateTime? endDate)
{
    var customer = _context.Customers.ToList();
    var procustomers =
        _context.ProductCustomers.ToList();
    var products = _context.Products.ToList();
    var categoreys = _context.Categories.ToList();
    var multiTable = from c in customer
```



Create **Report** action in the Admin controller.

```
join pc in procustomers on c.Id equals pc.CustomerId
join p in products on pc.ProductId equals p.Id
join cat in categorieys on p.CategoryId equals cat.Id
select new JoinTables { Product = p, Customer = c,
ProductCustomer = pc, Category = cat };
var modelContext =
_context.ProductCustomers.Include(p =>
p.Customer).Include(p => p.Product);

if (stratDate == null && endDate == null)
{
```



Create **Report** action in the Admin controller.

```
ViewBag.TotalQuantity = modelContext.Sum(x =>
x.Quantity);
ViewBag.TotalPrice = modelContext.Sum(x =>
x.Product.Price * x.Quantity);
var model3 = Tuple.Create<IEnumerable<JoinTables>,
IEnumerable<ProductCustomer>>(multiTable, await
modelContext.ToListAsync());
return View(model3);

}
else if (stratDate == null && endDate != null)
{
```



Create **Report** action in the Admin controller.

```
ViewBag.TotalQuantity = modelContext.Where(x =>
x.DateFrom.Value.Date == endDate).Sum(x =>
x.Quantity);
ViewBag.TotalPrice = modelContext.Where(x =>
x.DateFrom.Value.Date == endDate).Sum(x =>
x.Product.Price * x.Quantity);
var model3 = Tuple.Create<IEnumerable<JoinTables>,
IEnumerable<ProductCustomer>>(multiTable, await
modelContext.Where(x => x.DateFrom.Value.Date ==
endDate).ToListAsync());
return View(model3);
}
```





Create **Report** action in the Admin controller.

```
else if (stratDate != null && endDate == null)
{
    ViewBag.TotalQuantity = modelContext.Where(x =>
        x.DateFrom.Value.Date == stratDate).Sum(x =>
        x.Quantity);
    ViewBag.TotalPrice = modelContext.Where(x =>
        x.DateFrom.Value.Date == stratDate).Sum(x =>
        x.Product.Price * x.Quantity);
    var model3 = Tuple.Create<IEnumerable<JoinTables>,
        IEnumerable<ProductCustomer>>(multiTable, await
        modelContext.Where(x => x.DateFrom.Value.Date ==
```



Create **Report** action in the Admin controller.

```
stratDate).ToListAsync());  
return View(model3);  
}  
else  
{  
    ViewBag.TotalQuantity = modelContext.Where(x =>  
        x.DateFrom >= stratDate && x.DateFrom <=  
        endDate).Sum(x => x.Quantity);  
    ViewBag.TotalPrice = modelContext.Where(x =>  
        x.DateFrom >= stratDate && x.DateFrom <=  
        endDate).Sum(x => x.Product.Price * x.Quantity);  
}
```



Create **Report** action in the Admin controller.

```
var model3 = Tuple.Create<IEnumerable<JoinTables>,  
IEnumerable<ProductCustomer>>(multiTable, await  
modelContext.Where(x => x.DateFrom >= stratDate &&  
x.DateFrom <= endDate).ToListAsync());  
return View(model3);  
}
```



## Create a **Report** View

×

### Add Razor View

View name

Template

Model class

Data context class

Options

☐ Create as a partial view

☐ Reference script libraries

☒ Use a layout page

...

(Leave empty if it is set in a Razor \_viewstart file)

Add Cancel

Add the following to the report view

```
@model Tuple<IEnumerable<JoinTables>,  
IEnumerable<ProductCustomer>>  
  
@{  
    ViewData["Title"] = "Index";  
    Layout = "~/Views/Shared/_AdminLayout.cshtml";  
}  
<link  
href="https://cdn.datatables.net/1.11.1/css/jquery.d  
ataTables.min.css" rel="stylesheet" />  
<link
```



Add the following to the report view

```
href="https://cdn.datatables.net/buttons/2.0.0/css/b
uttons.dataTables.min.css" rel="stylesheet" />
<div class="container">

<div class="p-3 mb-2 bg-dark text-white text-center
font-weight-bold">Report 1</div>

<div class="row">
<div class="col-6">
<form action="Report" method="post">
<input type="date" class="form-control"
```



Add the following to the report view

```
name="stratDate" placeholder="Start Date">
<input type="date" class="form-control"
name="endDate" placeholder="End Date">
<button type="submit" class="btn btn-
dark">Search</button>
</form>
</div>
</div>
<div class="row">
<div class="col-10">
<table class="table example">
<thead>
```



Add the following to the report view

```
<tr>  
<th>  
Customer First Name  
</th>  
<th>  
Customer Last Name  
</th>  
<th>  
Product Name  
</th>  
<th>  
Quantity
```





Add the following to the report view

```
</th>  
<th>  
Price  
</th>  
<th>  
Date From  
</th>  
<th>  
Date To  
</th>  
</tr>  
</thead>
```



Add the following to the report view

```
<tbody>
@foreach (var item in Model.Item2)
{
<tr>
<td> @Html.DisplayFor(modelItem =>
item.Customer.Fname)</td>
<td> @Html.DisplayFor(modelItem =>
item.Customer.Lname)</td>
<td> @Html.DisplayFor(modelItem =>
item.Product.Nameee)</td>
<td> @Html.DisplayFor(modelItem =>
item.Quantity)</td>
```



Add the following to the report view

```
<td> @(item.Quantity * item.Product.Price)</td>
<td> @Html.DisplayFor(modelItem =>
item.DateFrom)</td>
<td> @Html.DisplayFor(modelItem => item.DateTo)</td>
</tr>
}
</tbody>
<tfoot>
<tr>
<td>Total Quantity</td>
<td>@ViewBag.TotalQuantity</td>
<td></td>
```



Add the following to the report view

```
<td></td>
<td></td>
</tr>
<tr>
<td>Total Price</td>
<td>@ViewBag.TotalPrice</td>
<td></td>
<td></td>
<td></td>
</tr>
</tfoot>
</table>
```



Add the following to the report view

```
</div>
</div>
<div class="p-3 mb-2 bg-dark text-white text-center
font-weight-bold">Report 2</div>
<div class="row">
<div class="col-10">
<table class="table example">
<thead>
<tr>
<th>Customer Name</th>
<th>Product Name</th>
<th>Category Name</th>
```



Add the following to the report view

```
<th>Product Price</th>
</tr>
</thead>
<tbody>
@foreach (var item in Model.Item1)
{
<tr>
<td>@item.Customer.Fname</td>
<td>@item.Product.Sale</td>
<td>@item.Category.CategoryName</td>
<td>@item.Product.Price</td>
</tr>}

```



Add the following to the report view

```
</tbody>
</table>
</div>
</div>
</div>
<script src="https://code.jquery.com/jquery-
3.5.1.js"></script>
<script
src="https://cdn.datatables.net/1.11.1/js/jquery.dat
aTables.min.js" defer></script>
```



Add the following to the report view

```
<script  
src="https://cdn.datatables.net/buttons/2.0.0/js/dat  
aTables.buttons.min.js" defer></script>  
<script  
src="https://cdnjs.cloudflare.com/ajax/libs/jzip/3.  
1.3/jzip.min.js"></script>  
<script  
src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/  
0.1.53/pdfmake.min.js"></script>  
<script  
src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/  
0.1.53/vfs_fonts.js"></script>
```





Add the following to the report view

```
<script  
src="https://cdn.datatables.net/buttons/2.0.0/js/but  
tons.html5.min.js" defer></script>  
<script>  
$(document).ready(function () {  
$('.example').DataTable({  
dom: 'Bfrtip',  
buttons: [  
'copyHtml5',  
'excelHtml5',  
'csvHtml5',  
'pdfHtml5']});});</script>
```



The Result

Report 1

mm/dd/yyyy

mm/dd/yyyy

Search

Copy

Excel

CSV

PDF

Search:

Customer First Name	Customer Last Name	Product Name	Quantity	Price	Date From	Date To
Bayan	Mohammad	Burger	5.00	30	5/4/2023 12:40:11 PM	5/6/2023 12:40:16 PM
dana	ahmad	chicken	10.00	150	3/1/2022 12:18:00 AM	3/3/2022 12:18:08 AM
Total Quantity		15				
Total Price		180				

Showing 1 to 2 of 2 entries

Previous

1

Next

Report 2

Copy

Excel

CSV

PDF

Search:

Customer Name	Product Name	Category Name	Product Price
Bayan	5	Fast Food	6
dana	1	Fast Food	15