Лабораторная работа №19
Настройка отображения данных в веб-приложении

1 Цель работы

1.1 Научиться реализовывать настройку интерфейса пользователя для отображения, фильтрации и сортировки данных БД в веб приложении ASP.Net.

2Литература

2.1 Руководство по ASP.NET Core и C#. metanit.com – Текст : электронный // metanit.com, 2025. – URL: https://metanit.com/sharp/aspnet6/ - гл.12-13

3Подготовка к работе

3.1 Повторить теоретический материал (см.п.2).

3.2 Изучить описание практической работы.

4Основное оборудование

4.1 Персональный компьютер.

5Задание

Задание выполнять в проекте ЛР №17-18 на странице Сеансы.

5.1 Поиск

5.1.1 Добавить в класс страницы просмотра Сеансы свойство FilmTitle, привязываемое к элементам страницы (чтобы не передавать через параметры в метод Get):

```
[BindProperty(SupportsGet = true)]
public string FilmTitle { get; set; }
```

5.1.2 Добавить в верхнюю часть страницы Сеансы форму для указания данных поиска, при нажатии на кнопку Enter выполнялась:

```
<form method="get">
    <input type="text"
           name="filmTitle"
           value="@Model.FilmTitle"
           placeholder="Поиск..."
           onchange="this.form.submit()"
           class="form-control" />
 </form>
```

5.1.3 Реализовать поиск по части названия фильма в методе OnGetAsync().

5.2 Сортировка

5.2.1 Добавить в класс страницы просмотра Сеансы свойство SortColumn, привязываемое к элементам страницы.

5.2.2 Добавить в форму выпадающий список для сортировки данных по возрастанию/убыванию названия фильма:

```
<select class="form-control"
        name="sortColumn"
        onchange="this.form.submit()">
```

<option value="price" selected="@(Model.SortColumn == "price")">По цене (дешевые)</option>
<option value="price_desc" selected="@(Model. SortColumn == "price_desc")">По цене (дорогие)</option>
</select>

5.2.3 Реализовать сортировку по выбранному варианту в методе OnGetAsync().

5.3 Фильтрация

5.3.1 Добавить в класс страницы просмотра списка Сеансы свойство Hall, привязываемое к элементам страницы.

5.3.2 Добавить в начало метода OnGetAsync() код для получения списка всех кинозалов и добавить :

ViewData["Halls"] = new SelectList(_context.Залы, "Id", "Информация");

5.3.3 Добавить в форму выпадающий список для вывода списка залов:

```
<select class="form-control" asp-items="ViewBag.Halls"
    asp-for="Hall"
    name="hall"
    onchange="this.form.submit()">
</select>
```

5.3.4 Реализовать фильтрацию по выбранному варианту в методе OnGetAsync().

5.4 Пагинация

5.4.1 Добавить в класс страницы просмотра списка Сеансы свойство PageIndex, привязываемое к элементам, и TotalPages.

5.4.2 Добавить в нижнюю часть страницы кнопки-гиперссылки для переключения между страницами:

```
@if (Model.PageIndex > 1)
{
  <a asp-page="./Index" asp-route-pageIndex="@(Model.PageIndex-1)">Назад</a>
}
@if (Model.PageIndex < Model.TotalPages)
{
  <a asp-page="./Index" asp-route-pageIndex="@(Model.PageIndex+1)">Вперед</a>
}
```

5.4.3 Реализовать пагинацию (переход между выбранными страницами) в методе OnGetAsync().

5.4.4 Реализовать вывод текущего номера страницы между кнопками Назад и Вперед.

6Порядок выполнения работы

6.1 Выполнить все задания из п.5.

6.2 Ответить на контрольные вопросы.

7Содержание отчета

7.1 Титульный лист

7.2 Цель работы

8Контрольные вопросы