@page "/hotel-room/create"

@using Models;

@using Business.Repository.IRepository;

@inject IHotelRoomRepository HotelRoomRepository

@inject NavigationManager NavigationManager

<div class="row mt-2 mb-5">

<h3 class="card-title text-info mb-3 ml-3">Create Hotel Room</h3>

<div class="col-md-12">

<div class="card">

<div class="card-body">

<**EditForm** **Model**="@HotelRoomModel" **OnValidSubmit**="HandleHotelRoomUpsert">

<**DataAnnotationsValidator**/>

<div class="form-group">

<label>Name</label>

<**InputText** @bind-Value="HotelRoomModel.Name" class="form-control" />

<**ValidationMessage** **For**="@(()=>HotelRoomModel.Name)" />

</div>

<div class="form-group">

<label>Occupancy</label>

<**InputNumber** @bind-Value="HotelRoomModel.Occupancy" class="form-control" />

<**ValidationMessage** **For**="@(()=>HotelRoomModel.Occupancy)" />

</div>

<div class="form-group">

<label>Rate</label>

<**InputNumber** @bind-Value="HotelRoomModel.RegularRate" class="form-control" />

<**ValidationMessage** **For**="@(()=>HotelRoomModel.RegularRate)" />

</div>

<div class="form-group">

<label>Sq ft.</label>

<**InputText** @bind-Value="HotelRoomModel.SqFt" class="form-control" />

</div>

<div class="form-group">

<label>Details</label>

<**InputTextArea** @bind-Value="HotelRoomModel.Details" class="form-control" />

</div>

<div class="form-group">

<button class="btn btn-primary">@Title</button>

<**NavLink** href="Hotel-room" class="btn btn-secondary">Back to Index</**NavLink**>

</div>

</**EditForm**>

</div>

</div>

</div>

</div>

@code {

private HotelRoomDTO HotelRoomModel { get; set; } = new HotelRoomDTO();

private string Title { get; set; } = "Create";

private async Task HandleHotelRoomUpsert()

{

var roomDetailsByName = await HotelRoomRepository.IsRoomUnique(HotelRoomModel.Name);

if (roomDetailsByName !=null)

{

//there is duplicate room show error

return;

}

var createdResult = await HotelRoomRepository.CreateHotelRoom(HotelRoomModel);

NavigationManager.NavigateTo("hotel-room");

}

}

@page "/hotel-room"

@using Models;

@using Business.Repository.IRepository;

@inject IHotelRoomRepository HotelRoomRepository

<div class="row mt-4">

<div class="col-8">

<h4 class="card-title text-info"> Hotel Rooms</h4>

</div>

<div class="col-3 offset-1">

<**NavLink** class="btn btn-info form-control text-white"

href="Hotel-room/create">Add New Room

</**NavLink**>

</div>

</div>

<div class="row mt-4">

<div class="col-12">

<table class="table table-bordered table-hover">

<thead>

<tr>

<th>Name</th>

<th>Occupancy</th>

<th>Rate</th>

<th>Sqft</th>

<th>For Buttons</th>

</tr>

</thead>

<tbody>

@if (HotelRooms.Any())

{

@foreach (var room in HotelRooms)

{

<tr>

<td>@room.Name</td>

<td>@room.Occupancy</td>

<td>@room.RegularRate.ToString("c")</td>

<td>@room.SqFt</td>

<td></td>

</tr>

}

}

else

{

<tr>

<td colspan="5">No Records of Rooms</td>

</tr>

}

</tbody>

</table>

</div>

</div>

@code {

private IEnumerable<HotelRoomDTO> HotelRooms { get; set; } = new List<HotelRoomDTO>();

protected override async Task OnInitializedAsync()

{

HotelRooms = await HotelRoomRepository.GetAllHotelRooms();

}

}