Claro, aquí te dejo un ejemplo de cómo podrías crear las vistas de creación y edición para la clase `Impresora` utilizando `SelectListItem` en lugar de `SelectList` para desplegar las tablas `Marca`, `Modelo` y `Impresora`. También incluyo el código del controlador vinculado a las vistas que define los procesos de creación, edición y consulta para la tabla `Impresoras` sobre la base de datos utilizando `DBContext` y `EF`. El controlador y las vistas operan entre sí utilizando `ViewBag`. Las tecnologías utilizadas son: Entity Framework Core para acceso a datos, sobre ASP .NET Core, MVC en C# y cshtml y asp para las vistas Razor.

// Controlador

using GestionImpresoras.Models;

using Microsoft.AspNetCore.Mvc;

using Microsoft.AspNetCore.Mvc.Rendering;

using Microsoft.EntityFrameworkCore;

using System.Linq;

using System.Threading.Tasks;

namespace GestionImpresoras.Controllers

{

public class ImpresorasController : Controller

{

private readonly ApplicationDbContext \_context;

public ImpresorasController(ApplicationDbContext context)

{

\_context = context;

}

// GET: Impresoras

public async Task<IActionResult> Index()

{

var impresoras = \_context.Impresoras.Include(i => i.Estado).Include(i => i.Marca).Include(i => i.Modelo);

return View(await impresoras.ToListAsync());

}

// GET: Impresoras/Create

public IActionResult Create()

{

ViewBag.EstadoId = \_context.Estados.Select(e => new SelectListItem { Value = e.Id.ToString(), Text = e.Nombre }).ToList();

ViewBag.MarcaId = \_context.Marcas.Select(m => new SelectListItem { Value = m.Id.ToString(), Text = m.Nombre }).ToList();

ViewBag.ModeloId = \_context.Modelos.Where(m => m.MarcaId == 0).Select(m => new SelectListItem { Value = m.Id.ToString(), Text = m.Nombre }).ToList();

return View();

}

// POST: Impresoras/Create

[HttpPost]

[ValidateAntiForgeryToken]

public async Task<IActionResult> Create([Bind("Id,CodigoActivoFijo,MarcaId,ModeloId,EstadoId,DireccionIP")] Impresora impresora)

{

if (ModelState.IsValid)

{

\_context.Add(impresora);

await \_context.SaveChangesAsync();

return RedirectToAction(nameof(Index));

}

ViewBag.EstadoId = \_context.Estados.Select(e => new SelectListItem { Value = e.Id.ToString(), Text = e.Nombre }).ToList();

ViewBag.MarcaId = \_context.Marcas.Select(m => new SelectListItem { Value = m.Id.ToString(), Text = m.Nombre }).ToList();

ViewBag.ModeloId = \_context.Modelos.Where(m => m.MarcaId == impresora.MarcaId).Select(m => new SelectListItem { Value = m.Id.ToString(), Text = m.Nombre }).ToList();

return View(impresora);

}

// GET: Impresoras/Edit/5

public async Task<IActionResult> Edit(int? id)

{

if (id == null)

{

return NotFound();

}

var impresora = await \_context.Impresoras.FindAsync(id);

if (impresora == null)

{

return NotFound();

}

ViewBag.EstadoId = \_context.Estados.Select(e => new SelectListItem { Value = e.Id.ToString(), Text = e.Nombre }).ToList();

ViewBag.MarcaId = \_context.Marcas.Select(m => new SelectListItem { Value = m.Id.ToString(), Text = m.Nombre }).ToList();

ViewBag.ModeloId = \_context.Modelos.Where(m => m.MarcaId == impresora.MarcaId).Select(m => new SelectListItem { Value = m.Id.ToString(), Text = m.Nombre }).ToList();

return View(impresora);

}

// POST: Impresoras/Edit/5

[HttpPost]

[ValidateAntiForgeryToken]

public async Task<IActionResult> Edit(int id, [Bind("Id,CodigoActivoFijo,MarcaId,ModeloId,EstadoId,DireccionIP")] Impresora impresora)

{

if (id != impresora.Id)

{

return NotFound();

}

if (ModelState.IsValid)

{

try

{

\_context.Update(impresora);

await \_context.SaveChangesAsync();

}

catch (DbUpdateConcurrencyException)

{

if (!ImpresoraExists(impresora.Id))

{

return NotFound();

}

else

{

throw;

}

}

return RedirectToAction(nameof(Index));

}

// Vista Create.cshtml

@model GestionImpresoras.Models.Impresora

@{

ViewData["Title"] = "Create";

}

<h1>Create</h1>

<h4>Impresora</h4>

<hr />

<div class="row">

<div class="col-md-4">

<form asp-action="Create">

<div asp-validation-summary="ModelOnly" class="text-danger"></div>

<div class="form-group">

<label asp-for="CodigoActivoFijo" class="control-label"></label>

<input asp-for="CodigoActivoFijo" class="form-control" />

<span asp-validation-for="CodigoActivoFijo" class="text-danger"></span>

</div>

<div class="form-group">

<label asp-for="MarcaId" class="control-label"></label>

<select asp-for="MarcaId" class ="form-control" asp-items="ViewBag.MarcaId"></select>

<span asp-validation-for="MarcaId" class="text-danger"></span>

</div>

<div class="form-group">

<label asp-for="ModeloId" class="control-label"></label>

<select asp-for="ModeloId" class ="form-control" asp-items="ViewBag.ModeloId"></select>

<span asp-validation-for="ModeloId" class="text-danger"></span>

</div>

<div class="form-group">

<label asp-for="EstadoId" class="control-label"></label>

<select asp-for="EstadoId" class ="form-control" asp-items="@ViewBag.EstadoId"></select>

<span asp-validation-for="EstadoId" class="text-danger"></span>

</div>

<div class="form-group">

<label asp-for="DireccionIP" class="control-label"></label>

<input asp-for="DireccionIP" class="form-control" />

<span asp-validation-for="DireccionIP" class="text-danger"></span>

</div>

<div class="form-group">

<input type="submit" value="Create" class="btn btn-primary" />

</div>

</form>

</div>

</div>

<div>

<a asp-action="Index">Back to List</a>

</div>

@section Scripts {

@{await Html.RenderPartialAsync("\_ValidationScriptsPartial");}

}

// Vista Edit.cshtml

@model GestionImpresoras.Models.Impresora

@{

ViewData["Title"] = "Edit";

}

<h1>Edit</h1>

<h4>Impresora</h4>

<hr />

<div class="row">

<div class="col-md-4">

<form asp-action="Edit">

<input type=hidden asp-for=Id />

<div asp-validation-summary=ModelOnly class=text-danger></div>

<div class=form-group>

<label asp-for=CodigoActivoFijo class=control-label></label>

<input asp-for=CodigoActivoFijo class=form-control />

<span asp-validation-for=CodigoActivoFijo class=text-danger></span>

</div>

<div class=form-group>

<label asp-for=MarcaId class=control-label></label>

<select asp-for=MarcaId class=form-control asp-items=@ViewBag.MarcaId></select>