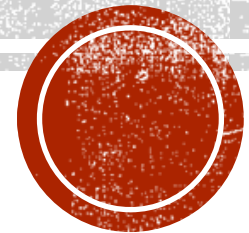


TP N°5

Accès à une base de données SQL SERVER
Avec Entity Framework Core (Approche Code First)
SITE E-COMMERCE

Enseignants :

Malek Zribi & Lilia Ayadi



PROJET BASÉ SUR L'APPROCHE CODE FIRST

- Dans ce TP, nous allons créer un site de vente en ligne de vêtements classés par des catégories. Pour cela, on va suivre les étapes suivantes :
- Ouvrez Visual Studio 2022 et cliquez sur **Créer un nouveau projet** , de type **Application web ASP .Net Core (Modèle-Vue-Contrôleur)**.
- On va maintenant ajouter les classes suivantes :
 - Categorie
 - Commande
 - Panier
 - Produit



CODE DE LA CLASSE PRODUIT

- Le code de la classe « Produit » est le suivant :

```
public class Produit
{
    public int ProduitId { get; set; }

    [Required]
    [StringLength(50, MinimumLength = 2)]
    public string Nom { get; set; }

    [Required]
    [Display(Name = "Prix en Dinars :")]
    public float Prix { get; set; }

    [Required(ErrorMessage = "Erreur categorie !")]
    [Display(Name = "Catégorie")]
    public int CategorieId { get; set; }
    public virtual Categorie Categorie { get; set; }

    [Required]
    [Display(Name = "Quantité en unité :")]
    public int Quantite { get; set; }

    [Required]
    [Display(Name = "Image :")]
    public string Image { get; set; } }
```



CODE DE LA CLASSE CATEGORIE

- Le code de la classe « Catégorie » qui représente la catégorie d'un « Produit » et qui aura l'une des valeurs suivantes « Homme, Femme, Enfants » est le suivant :

```
public class Categorie
{
    public int CategorieId { get; set; }
    [Required]
    [Display(Name = "Nom")]
    public string Nom { get; set; }
    public virtual ICollection<Produit> Produits { get; set; }
}
```



CODE DES CLASSE PANIER ET COMMANDE

- Le code de la classe « Panier » est le suivant :

```
public class Panier
{
    public int PanierId { get; set; }
}
```

- Le code de la classe « Commande » est le suivant :

```
public class Commande
{
    public int CommandeId { get; set; }
    public int PanierId { get; set; }
    public virtual Panier Panier { get; set; }
}
```

- Créer Votre base de donnée.
- Implémenter la classe ProduitRepository, le contrôleur ProduitController et créer les Views correspondantes en se basant sur les TP N°3 et 4.



FORMULAIRES DE MAJ DE PRODUIT

Index

[Create New](#)

Nom	Prix en Dinars :	Categorie	Quantité en unité :	Image :
Ensemble	200	2	603de996-f9b3-4179-851a-4c56a640255f_cloth_4.jpg	Edit Details Delete
Jean	75	20	bfb9b6ce-8c5d-4fa9-81a9-e31a8a75310c_Jean1.png	Edit Details Delete
Robe Courte	150	10	31e33ee6-fabf-4163-93d9-a7e5769ab788_cloth_8.jpg	Edit Details Delete
Veste	210	20	42028c65-6fdd-41a3-9235-f2ef9a2797b4_Veste1.jpg	Edit Details Delete

Nouveau

Produit

Nom

Prix en Dinars :

Catégorie

Quantité en unité :

Image

Choisir un fichier

Aucun ...tionné

:

Choose File...

Create

[Back to List](#)

Details

Produit

Nom

Jean

Prix en Dinars :

75

Catégorie

1

Quantité en unité :

20

Image



[Edit](#) | [Back to List](#)



GESTION DES CATÉGORIES

- La table « Catégorie » sera remplie manuellement par les informations suivantes :

Nombre maximal de lignes : 1000		
	Categoriell	Nom
▶	1	Homme
	2	Femme
	3	Enfant
★	NULL	NULL

- Ajouter la classe `CatégorieRepository`, contenant les différentes méthodes de mise à jour ainsi que la méthode suivante :

```
public Catégorie GetProduitsByCateg(string cat)
{
    return context.Categories.Include("Produits")
        .Single(g => g.Nom == cat);
}
```



PAGE D'ACCUEIL DE NOTRE SITE

- La page d'accueil de notre site aura la forme suivante :

[Site Vente en Ligne](#) [Home](#) [Accueil](#) [Mon Panier](#) [Catégorie ▼](#)

[Register](#) [Login](#)



Vêtements hommes Placard

Les hommes n'ont pas besoin d'une grande garde-robe pour paraître au top...

[Détails »](#)



Vêtements femmes Placard

Qui a dit que la mode est éphémère ? Car il n'y a rien d'éphémère...

[Détails »](#)



Vêtements enfants Placard

Les parents et la mode enfants ont une chose en commun...

[Détails »](#)



PAGE D'ACCUEIL DE NOTRE SITE

- Le click sur le bouton « Détails » ou le choix d'une catégorie de la liste déroulante, nous envoie vers une page relative à la liste des produits d'une catégorie donnée, représentée sous forme d'objet de type card utilisé dans le TP N°2 :

[Site Vente en Ligne](#) [Home](#) [Accueil](#) [Mon Panier](#) [Catégorie ▼](#)

[Register](#) [Login](#)

Liste des produits de la catégorie : Homme

Désignation : Jean

Prix :75



[Détails »](#) [Ajouter au panier »](#)

Désignation : Veste

Prix :210



[Détails »](#) [Ajouter au panier »](#)

PAGE D'ACCUEIL DE NOTRE SITE

- Pour ce faire, dans le « HomeController », on va ajouter la méthode d'action suivante :

```
public ActionResult DetailsCategorie(string cat)
{
    var produitListe = categorieRepository.GetProduitsByCateg(cat);
    return View(produitListe);
}
```

- Et on va créer la vue correspondante avec le code suivant :

```
@model Site_VenteEnLigne.Models.Categorie
@{
    ViewBag.Title = "Produit";
}
<h1>Liste des produits de la catégorie : @Model.Nom</h1>
<br />
<br />
<div class="card-group">
    @foreach (var produit in Model.Produits)
    {
        <div class="card m-3" style="min-width: 18rem; max-width:30.5%;">
            <div class="card-header">
                <h5><b>Désignation : </b> @produit.Nom</h5>
                <h5><b>Prix : </b>@produit.Prix</h5>
            </div>

            <center>
            <div class="caption">
                <center>
                    <a href="/Produit/DetailsCat/@produit.ProduitId" class="btn btn-default" role="button">Details »</a>
                    @Html.ActionLink("Ajouter au panier »", "AjouterProduit", "Panier", new { id = @produit.ProduitId }, "")
                </center>
            </div>
        </div>
    }
</div>
```



PAGE D'ACCUEIL DE NOTRE SITE

- Pour ce faire, dans le « HomeController », on va ajouter la méthode d'action suivante :

```
public ActionResult DetailsCategorie(string cat)
{
    var produitListe = categorieRepository.GetProduitsByCateg(cat);
    return View(produitListe);
}
```

- Et on va créer la vue correspondante avec le code suivant :

```
@model Site_VenteEnLigne.Models.Categorie
@{
    ViewBag.Title = "Produit";
}
<h1>Liste des produits de la catégorie : @Model.Nom</h1>
<br />
<br />
<div class="card-group">
    @foreach (var produit in Model.Produits)
    {
        <div class="card m-3" style="min-width: 18rem; max-width:30.5%;">
            <div class="card-header">
                <h5><b>Désignation : </b> @produit.Nom</h5>
                <h5><b>Prix : </b>@produit.Prix</h5>
            </div>

            <center>
            <div class="caption">
                <center>
                    <a href="/Produit/DetailsCat/@produit.ProduitId" class="btn btn-default" role="button">Details >></a>
                    @Html.ActionLink("Ajouter au panier >>", "AjouterProduit", "Panier", new { id = @produit.ProduitId }, "")
                </center>
            </div>
        </div>
    }
</div>
```



AJOUT D'UN ÉLÉMENT AU PANIER

- On va commencer par l'ajout d'une nouvelle classe qui permettra de gérer la liste des produits sélectionnés. On va alors ajouter sous le dossier « Models », un dossier « Help » qui va inclure les deux classes suivantes : « Item » et « ListCart »

```
public class Item
{
    public int quantite { get; set; }
    private int _ProduitId;
    public Produit _produit = null;
    public Produit Prod
    {
        get { return _produit; }
        set { _produit = value; }
    }
    public string Description
    {
        get { return _produit.Nom; }
    }
    public float UnitPrice
    {
        get { return _produit.Prix; }
    }
    public float TotalPrice
    {
        get { return _produit.Prix * quantite; }
    }
    public Item(Produit p)
    {
        this.Prod = p;
    }
    public bool Equals(Item item)
    {
        return item.Prod.ProduitId == this.Prod.ProduitId;
    }
}
```



CODE DE LA CLASSE LISTECART

```
public class ListeCart
{
    public List<Item> Items { get; private set; }

    public static readonly ListeCart Instance;

    static ListeCart()
    {
        Instance = new ListeCart();
        Instance.Items = new List<Item>();
    }

    protected ListeCart() { }
    public void AddItem(Produit prod)
    {
        Boolean iswhat = false;
        // Create a new item to add to the cart
        foreach (Item a in Items)
        {
            if (a.Prod.ProduitId == prod.ProduitId)
            {
                a.quantite++;
                iswhat = true;
                return;
            }
        }
        if (iswhat == false)
        {
            Item newItem = new Item(prod);
            newItem.quantite = 1;
            Items.Add(newItem);
        }
    }
    public void setToNull()
    {}
}
```

```
public void SetLessOneItem(Produit produit)
{
    foreach (Item a in Items)
    {
        if (a.Prod.ProduitId == produit.ProduitId)
        {
            if (a.quantite <= 0)
            {
                RemoveItem(a.Prod);
                return;
            }
            else
            {
                a.quantite--;
                return;
            }
        }
    }
}

public void SetItemQuantity(Produit produit, int quantity)
{
    if (quantity == 0)
    {
        RemoveItem(produit);
        return;
    }

    foreach (Item a in Items)
    {
        if (a.Prod.ProduitId == produit.ProduitId)
        {
            a.quantite = quantity;
            return;
        }
    }
}
```



CODE DE LA CLASSE LISTCART

```
public void RemoveItem(Produit produit)
{
    Item t = null;

    foreach (Item a in Items)
    {
        if (a.Prod.ProduitId == produit.ProduitId)
        {
            t = a;
        }
    }

    if (t != null)
    {
        Items.Remove(t);
    }
}

public float GetSubTotal()
{
    float subTotal = 0;
    foreach (Item i in Items)
        subTotal += i.TotalPrice;
    return (float)subTotal;
}
}
```



CRÉATION DU CONTRÔLEUR PANIER

- Créer un contrôleur vide « PanierController », puis ajouter le code suivant :

```
public class PanierController : Controller
```

```
{
    readonly ProduitRepository produitRepository;
    public PanierController(ProduitRepository produitRepository)
    {
        this.produitRepository = produitRepository;
    }
    public ActionResult Index()
    {
        ViewBag.Liste = ListeCart.Instance.Items;
        ViewBag.total = ListeCart.Instance.GetSubTotal();
        return View();
    }
    public ActionResult AjouterProduit(int id)
    {
        Produit pp = produitRepository.GetById(id);
        ListeCart.Instance.AddItem(pp);
        ViewBag.Liste = ListeCart.Instance.Items;
        ViewBag.total = ListeCart.Instance.GetSubTotal();
        return View();
    }
}
```

```
[HttpPost]
```

```
public ActionResult PlusProduit(int id)
{
    Produit pp = produitRepository.GetById(id);
    ListeCart.Instance.AddItem(pp);
    Item trouve = null;
    foreach (Item a in ListeCart.Instance.Items)
    {
        if (a.Prod.ProduitId == pp.ProduitId)
            trouve = a;
    }
    var results = new {
        ct = 1,
        Total = ListeCart.Instance.GetSubTotal(),
        Quatite = trouve.quantite,
        TotalRow = trouve.TotalPrice
    };
    return Json(results);
}
```



CRÉATION DU CONTRÔLEUR PANIER (SUITE)

- Créer un contrôleur vide « PanierController », puis ajouter le code suivant :

```
[HttpPost]
public ActionResult MoinsProduit(int id)
{
    Produit pp = produitRepository.GetById(id);
    ListeCart.Instance.SetLessOneItem(pp);
    Item trouve = null;
    foreach (Item a in ListeCart.Instance.Items)
    {
        if (a.Prod.ProduitId == pp.ProduitId)
            trouve = a;
    }
    if (trouve != null)
    {
        var results = new
        {
            Total = ListeCart.Instance.GetSubTotal(),
            Quatite = trouve.quantite,
            TotalRow = trouve.TotalPrice,
            ct = 1
        };
        return Json(results);
    }
    else
    {
        var results = new
        {
            ct = 0
        };

        return Json(results);
    }
    return null;
}
```

```
public ActionResult CheckOut()
{
    return View();
}

[HttpPost]
public ActionResult CheckOut(FormCollection collection)
{
    ListeCart.Instance.Items.Clear();

    ViewBag.Message = "Commande effectuée avec succès";

    return View();
}

[HttpPost]
public ActionResult SupprimerProduit(int id)
{
    Produit pp = produitRepository.GetById(id);
    ListeCart.Instance.RemoveItem(pp);
    var results = new
    {
        Total = ListeCart.Instance.GetSubTotal(),
    };

    return Json(results);
}
```



CODE DU FICHIER « MONSCRIPT.JS »

- Avant de créer les views, on va créer un dossier « Scripts » dans lequel, on va ajouter un nouvel élément nommé « Monscript.js » dont le code est le suivant :

```
<script type="text/javascript">
    $(function () {
        // Document.ready -> link up remove event handler
        $(".PlusProduit").click(function () {
            // Get the id from the link
            var recordtoupdate = $(this).attr("data-id");
            if (recordtoupdate != '') {
                // Perform the ajax post
                $.post("/Panier/PlusProduit", { "id": recordtoupdate },
                    function (data) {

                        if (data.ct == "1") {

                            $('#totalapayer').text(data.Total);
                            $("#quantite_" + recordtoupdate).text(data.Quatite);
                            $("#rquantite_" + recordtoupdate).text(data.Quatite);
                            $("#total_" + recordtoupdate).text(data.TotalRow);

                        }

                    });
            }
        });
    });
};
```



CODE DU FICHIER « MONSCRIPT.JS »

```
// Document.ready -> link up remove event handler
$(".MinusProduit").click(function () {
    // Get the id from the link
    var recordtoupdate = $(this).attr("data-id");
    if (recordtoupdate != '') {
        // Perform the ajax post
        $.post("/Panier/MoinsProduit", { "id": recordtoupdate },
            function (data) {
                if (data.ct == "1") {
                    $('#totalapayer').text(data.Total);
                    $("#quantite_" + recordtoupdate).text(data.Quatite);
                    $("#rquantite_" + recordtoupdate).text(data.Quatite);
                    $("#total_" + recordtoupdate).text(data.TotalRow);
                }
                else if (data.ct == "0") {
                    $("#row-" + recordtoupdate).fadeOut('slow');
                }
            });
    }
});

// Document.ready -> link up remove event handler
$(".RemoveLink").click(function () {
    // Get the id from the link
    var recordtoupdate = $(this).attr("data-id");
    if (recordtoupdate != '') {
        // Perform the ajax post
        $.post("/Panier/SupprimerProduit", { "id": recordtoupdate },
            function (data) {

                $("#row-" + recordtoupdate).fadeOut('slow');
                $('#totalapayer').text(data.Total);
            });
    }
});
</script>
```



CRÉATION DES VIEWS

- On va créer la vue correspondante à l'action Index. Pour cela, on va créer une vue vide « Index », puis on va ajouter le code suivant :

```
@{
    ViewBag.Title = "Index";
}
<script src="~/Scripts/MonsScript.js"
    type="text/javascript"></script>
<h2>Mon panier</h2>

<div>
    @if (ViewBag.Liste.Count > 0)
    {
        <table class="table table-hover">
            <tr class="carousel">
                <th>
                    Image
                </th>
                <th class="danger">
                    Infos
                </th>
            </tr>
            @foreach (var item in ViewBag.Liste)
            {
                <tr id="row-@item.Prod.ProduitId">
                    <td>
                        
                    </td>
                    <td>
                        <h5>Nom : @item.Prod.Nom </h5><hr />
                        <h5>Quantité : <span id="quantite_@item.Prod.ProduitId">@item.quantite</span></h5><hr />
                    </td>
                </tr>
            }
        </table>
    }
}
```



VIEW « INDEX » (SUITE)

```
<a href="#" class="PlusProduit" data-id="@item.Prod.ProduitId">
     </a>
    <a href="#" class="MinusProduit" data-id="@item.Prod.ProduitId">
         </a>
    <a href="#" class="RemoveLink" data-id="@item.Prod.ProduitId">
         </a>
    <div>
        <hr />
        <h5>Prix de <span id="rquantite_@item.Prod.ProduitId">@item.quantite</span> article(s) : <span
id="total_@item.Prod.ProduitId">@item.TotalPrice</span></h5><hr />
    </div>
</td>
</tr>
}

</table>
<div>
    <strong>Prix total à payer : </strong>
    <div id="totalapayer">@ViewBag.total</div>
</div>
@(ViewBag.total > 0)
{
    <p class="button">
        <a href="~/Panier/Checkout" class="btn btn-success" role="button">Checkout »</a>
    </p>
}
}
else
{
    <p>Votre panier est vide</p>
}
</div>
```



CRÉATION DES VIEWS

- On va maintenant créer une vue vide « AjouterProduit », puis on va ajouter le code suivant :

```
@{
    ViewBag.Title = "AjouterProduit";
}

<script src="~/Scripts/MonScript.js"
    type="text/javascript"></script>
<h2>Mon panier</h2>
<div>
    @if (ViewBag.Liste.Count > 0)
    {
        <table class="table table-hover">
            <tr class="carousel">
                <th>
                    Image
                </th>
                <th class="danger">
                    Infos
                </th>
            </tr>
            @foreach (var item in ViewBag.Liste)
            {
                <tr id="row-@item.Prod.ProduitId">
                    <td>  </td>
                    <td>
                        <h5>Nom : @item.Prod.Nom </h5><hr />
                        <h5>Quantité : <span id="quantite_@item.Prod.ProduitId">@item.quantite</span></h5><hr />
                        <a href="#" class="PlusProduit" data-id="@item.Prod.ProduitId">
                             </a>
                        <a href="#" class="MinusProduit" data-id="@item.Prod.ProduitId">
                             </a>
                        <a href="#" class="RemoveLink" data-id="@item.Prod.ProduitId">
                             </a>
                    </td>
                </tr>
            }
        </table>
    }
}
```



VIEW « AJOUTERPRODUIT » (SUITE)

```
<div>
    <hr />
    <h5>Prix de <span id="rquantite_@item.Prod.ProduitId">@item.quantite</span> article(s) : <span
id="total_@item.Prod.ProduitId">@item.TotalPrice</span></h5><hr />
    </div>
    </td>
</tr>
}
</table>
<div>
    <strong>Prix total à payer : </strong>
    <div id="totalapayer">@ViewBag.total</div>
</div>
@(ViewBag.total > 0)
{
    <p class="button">
        <a href="~/Panier/Checkout" class="btn btn-success" role="button">Checkout »</a>
    </p>
}

}
else
{
    <p>Votre panier est vide</p>
}
</div>
```



MON PANIER

- Le panier sera représentée sous la forme suivante :

[Site Vente en Ligne](#) [Home](#) [Accueil](#) [Mon Panier](#) [Catégorie](#) ▼

[Register](#) [Login](#)

Mon panier

Image



Infos

Nom : Jean

Quantité : 2



Prix de 2 article(s) : 150

Prix total à payer :

150

True

[Checkout »](#)

