|  |
| --- |
| ClassLibrary建立 |
| |  | | --- | | **建立檔案** | | 檔案 =>專案 =>新增 =>類別庫(.NET Framework) | |

|  |
| --- |
| ClassLibrary宣告使用 |
| |  | | --- | | **建立檔案** | | 1. ClassLibrary專案先重建方案(可建立D: \OD\_ClassLibrary\OD\_ClassLibrary\bin\Debug\OD\_ClassLibrary.dll)  2.需要使用OD\_ClassLibrary的專案中  參考 =>加入參考 => 瀏覽 =>D: \OD\_ClassLibrary\OD\_ClassLibrary\bin\Debug\OD\_ClassLibrary.dll  3.在程式中宣告  using OD\_ClassLibrary;  public Class1 class1 = new Class1(); // Class1 是OD\_ClassLibrary中 namespace的名稱 | |

|  |
| --- |
| ClassLibrary專案內碼更新 |
| |  | | --- | | **建立檔案** | | 1. ClassLibrary專案先重建方案(可建立D: \OD\_ClassLibrary\OD\_ClassLibrary\bin\Debug\OD\_ClassLibrary.dll)  2.需要使用OD\_ClassLibrary的專案中  參考 =>加入參考 => 瀏覽 =>D: \OD\_ClassLibrary\OD\_ClassLibrary\bin\Debug\OD\_ClassLibrary.dll | |

|  |
| --- |
| OD\_ClassLibrary |
| |  | | --- | | **List頁數**   1. **宣告使用Numberofpages<T>(List<T> data,int number,int now\_page)**   **(1). List<T> data 需要設定的資料清單**  **(2). Number 每頁顯示資料筆數**  **(3). Now\_page 目前顯示頁數**   1. **T型別是Numberofpages<T>設定** 2. **data.Skip(0).Take(10) 顯示1-10的資料** | | public List<T> Numberofpages <T>(List<T> data,int number,int now\_page) where T :new ()  {  List<T> return\_list = new List<T>();  return\_list = data.Skip(number\*(now\_page-1)).Take(number+number\*(now\_page-1)).ToList();  if (return\_list.Count < number)  {  for(int a = return\_list.Count; a < number; a++)  {  return\_list.Add(new T());  }  }  return return\_list;  } | | **DateTable 轉成List** | | public List<T> DataTableToListWithJson<T>(DataTable dt)  {  string json = JsonConvert.SerializeObject(dt); // 將 DataTable 轉為 JSON  return JsonConvert.DeserializeObject<List<T>>(json); // 將 JSON 轉為 T 的列表  } | | **SQL資料查詢FUNCTION** | | public DataTable SQL\_select\_exce(string sqlstring)  {  string connectionString = "連資料庫路徑";  SqlDataReader reader;  DataTable data = new DataTable();  using (SqlConnection conn = new SqlConnection(connectionString))  {  conn.Open();  using (SqlCommand cmd = new SqlCommand(sqlstring, conn))  {  reader = cmd.ExecuteReader();//讀取回傳的資料  data.Load(reader);  }  }  return data;  } | |

|  |
| --- |
| Model |
| |  | | --- | | **變數宣告** | | public class ProductionDetail  {  public string productiondetail\_id { get; set; }//自動增長的編號  public string inout\_id { get; set; }//進出貨  public string product\_id { get; set; }//商品姓名  public int number { get; set; }//庫存數量  public int price { get; set; }//售貨價格  public DateTime? dr\_date { get; set; }//創建日期  public DateTime? dt\_date { get; set; }//更新日期  }  public class Production\_model  {  public string type\_id { get; set; }//類型編號  public string product\_name { get; set; } //產品名稱  public string product\_id { get; set; } //產品編號  public List<SelectListItem> select\_type\_id { get; set; }  public List<SelectListItem> select\_inout\_id { get; set; }  public List<SelectListItem> select\_product { get; set; }  public page pd\_page { get; set; }//頁數  } | | **呼叫SQL預設資料設定** | | public Production\_model Set\_Production\_model()  {  Production\_model return\_value = new Production\_model();  return\_value.select\_type\_id = Types\_list();  return\_value.select\_product = ALLProduct\_list();  return\_value.select\_inout\_id = new List<SelectListItem>();  return\_value.select\_inout\_id.Add(new SelectListItem { Text = "入庫", Value = "0" });  return\_value.select\_inout\_id.Add(new SelectListItem { Text = "報銷", Value = "1" });  return return\_value;  } | | public List<SelectListItem> Types\_list()  {  string sql = "SELECT type\_id AS Value, name AS Text FROM API.dbo.Types WITH(NOLOCK)";  return class1.DataTableToListWithJson<SelectListItem>(class1.SQL\_select\_exce(sql));  }  public List<SelectListItem> ALLProduct\_list()  {  string sql = $@"SELECT product\_id AS Value, name AS Text FROM orders.dbo.Production WITH(NOLOCK)";  return class1.DataTableToListWithJson<SelectListItem>(class1.SQL\_select\_exce(sql));  } | | **入庫/報銷歷史清單** | | public List<ProductionDetail> ProductionDetail\_history\_list(string inout\_id)  {  List<ProductionDetail> return\_list = new List<ProductionDetail>();  string sql = string.Empty;  if (!string.IsNullOrEmpty(inout\_id))  {  sql = $@"and inout\_id ={inout\_id}";  }  sql = $@"select  a.productiondetail\_id,  case a.inout\_id  when 0 then '入庫'  when 1 then '報銷'  end as inout\_id,  b.name as product\_id,  a.number,  a.price  from orders.dbo.ProductionDetail as a with(nolock)  left join orders.dbo.Production as b on a.product\_id=b.product\_id  where 1 = 1" +sql;  return\_list = class1.DataTableToListWithJson<ProductionDetail>(class1.SQL\_select\_exce(sql));  return return\_list;  } | | **呼叫SQL資料刪除** | | public void delete\_ProductionDetail(OD\_ViewModels ajs)  {  OD\_ViewModels ViewModels = new OD\_ViewModels();  OD\_Models Models = new OD\_Models();  if (!String.IsNullOrEmpty(ajs.production\_model.productiondetail\_id))  {  sql\_list = $@" delete from orders.dbo.ProductionDetail WHERE productiondetail\_id ='{alls. productiondetail\_id }' ";  class1. SQL\_delete (sql\_list);  }  } | |

|  |
| --- |
| Controller |
| |  | | --- | | **ProductionDetail\_history\_Index 內碼** | | //商品進出貨明細歷史紀錄  public ActionResult ProductionDetail\_history\_Index()  {  OD\_ViewModels ViewModels = new OD\_ViewModels();  OD\_Models Models = new OD\_Models();  ViewModels.production\_model = Models.Set\_Production\_model();  ViewModels.ProductionDetail\_list = Models.ProductionDetail\_history\_list(string.Empty);  #region 頁數  ViewModels.production\_model.pd\_page = new page();  ViewModels.production\_model.pd\_page.now\_page = 1;  ViewModels.production\_model.pd\_page.numbers = 5;  decimal pt = decimal.Parse(ViewModels.ProductionDetail\_list.Count.ToString()) /  decimal.Parse(ViewModels.production\_model.pd\_page.numbers.ToString());  ViewModels.production\_model.pd\_page.end\_page = (int)Math.Ceiling(pt);  #endregion  ViewModels.ProductionDetail\_list = class1.Numberofpages<ProductionDetail>(ViewModels.ProductionDetail\_list,  ViewModels.production\_model.pd\_page.numbers, ViewModels.production\_model.pd\_page.now\_page);  return View(ViewModels);  }  [HttpPost]  public ActionResult ProductionDetail\_history\_Index(OD\_ViewModels ajs)  {  OD\_ViewModels ViewModels = new OD\_ViewModels();  OD\_Models Models = new OD\_Models();  ViewModels.production\_model = Models.Set\_Production\_model();  ViewModels.ProductionDetail\_list = Models.ProductionDetail\_history\_list(string.Empty);  #region 頁數  ViewModels.production\_model.pd\_page = ajs.production\_model.pd\_page;  decimal pt = decimal.Parse(ViewModels.ProductionDetail\_list.Count.ToString()) /  decimal.Parse(ViewModels.production\_model.pd\_page.numbers.ToString());  ViewModels.production\_model.pd\_page.end\_page = (int)Math.Ceiling(pt);  #endregion  ViewModels.ProductionDetail\_list = class1.Numberofpages<ProductionDetail>(ViewModels.ProductionDetail\_list,  ViewModels.production\_model.pd\_page.numbers, ViewModels.production\_model.pd\_page.now\_page);  return View(ViewModels);  } | | **ProductionDetail\_delect 單筆資料刪除** | | [HttpPost]  public ActionResult ProductionDetail\_delect(OD\_ViewModels ajs)  {  OD\_ViewModels ViewModels = new OD\_ViewModels();  OD\_Models Models = new OD\_Models();  if (!String.IsNullOrEmpty(ajs.production\_model.productiondetail\_id))  {  Models.delete\_ProductionDetail(ajs.production\_model.productiondetail\_id);  }  return RedirectToAction("ProductionDetail\_history\_Index", ajs); ;  } | |

|  |
| --- |
| View |
| |  | | --- | | **ProductionDetail\_history\_Index.cshtml** | | @using orders.Models.ViewModels  @model OD\_ViewModels  <style>  table.custom-table {  width: 100%; /\* 設定表格寬度為 80% \*/  height: 50px; /\* 設定表格高度為 200px \*/  border: 1px black;  }  th {  border: 1px solid black; /\* 表頭藍色邊框 \*/  padding: 10px;  }  td {  border: 1px solid black; /\* 單元格黑色邊框 \*/  padding: 10px;  }  </style>  @using (Html.BeginForm("ProductionDetail\_history\_Index", "orders", FormMethod.Post, new { id = "ProductionDetail\_from" }))  {  @Html.HiddenFor(x => x.production\_model.pd\_page.now\_page)  @Html.HiddenFor(x => x.production\_model.pd\_page.end\_page)  @Html.HiddenFor(x => x.production\_model.pd\_page.numbers)  }  @using (Html.BeginForm("ProductionDetail\_delect", "orders", FormMethod.Post, new { id = "ProductionDetail\_delect\_from" }))  {  @Html.HiddenFor(x=>x.production\_model.productiondetail\_id)  @Html.HiddenFor(x => x.production\_model.pd\_page.now\_page)  @Html.HiddenFor(x => x.production\_model.pd\_page.end\_page)  @Html.HiddenFor(x => x.production\_model.pd\_page.numbers)  }  <div class="jumbotron">  <h2>入庫/報銷申請歷史清單</h2>  </div>  <div class="jumbotron">  <table class="custom-table">  <tr>  <th>  <h4>編號</h4>  </th>  <th>  <h4>請選入庫/報銷</h4>  </th>  <th>  <h4>商品名稱</h4>  </th>  <th>  <h4>數量</h4>  </th>  <th>  <h4>價格</h4>  </th>  <th>  <h4>刪除</h4>  </th>  </tr>  @for (int pd = 0; pd < Model.ProductionDetail\_list.Count; pd++)  {  <tr>  <th>  <h4> @Model.ProductionDetail\_list[pd].productiondetail\_id</h4>  </th>  <th>  <h4>@Model.ProductionDetail\_list[pd].inout\_id</h4>  </th>  <th>  <h4> @Model.ProductionDetail\_list[pd].product\_id</h4>  </th>  <th>  <h4> @Model.ProductionDetail\_list[pd].number</h4>  </th>  <th>  <h4> @Model.ProductionDetail\_list[pd].price</h4>  </th>  <th>  <h4><button onclick="delect(@Model.ProductionDetail\_list[pd].productiondetail\_id)">刪除</button> </h4>  </th>  </tr>  }  <tr>  <th colspan="6">  <div class="ts-pagination is-fluid">  <h4>  <a class="item is-back" onclick="changePage('up')">上一頁</a>  @for (int a = 1; a <= Model.production\_model.pd\_page.end\_page; a++)  {  if ((Model.production\_model.pd\_page.now\_page - a) < 5)  {  <a class="item" onclick="changePage(@a)"> @a </a>  }  else { break; }  }  <a class="item is-next" onclick="changePage('down')">下一頁</a>  </h4>  </div>  </th>  </tr>  </table>  </div>  <script>  function changePage(page) {  var now\_page = $("#production\_model\_pd\_page\_now\_page").val();  if (!isNaN(Number(page))) {  now\_page = page;  } else if (page == "up") {  now\_page = Number(now\_page) - 1;  } else if (page == "down") {  now\_page = Number(now\_page) + 1;  }  if (now\_page <= @Model.production\_model.pd\_page.end\_page&& now\_page >= 1) {  $("#production\_model\_pd\_page\_now\_page").val(now\_page);  from();  }  }  function delect(productiondetail\_id) {  if (productiondetail\_id != null && productiondetail\_id != '') {  $('#production\_model\_productiondetail\_id').val(productiondetail\_id);  var form = document.getElementById('ProductionDetail\_delect\_from');  form.submit();  }  }  function from() {  var form = document.getElementById("ProductionDetail\_from");  form.submit(); // 提交表單  }  </script> | |