// --------------------------------------------------------------------------------------------------------------------

// <copyright file="InvalidBusinessException.cs" company="">

//

// </copyright>

// <summary>

// The invalid business exception.

// </summary>

// --------------------------------------------------------------------------------------------------------------------

using System;

using System.Runtime.Serialization;

using System.Security.Permissions;

namespace Core.Exception

{

/// <summary>

/// The invalid business exception.

/// </summary>

[Serializable]

public class InvalidBusinessException : System.Exception

{

/// <summary>

/// Gets or sets DuplicateName.

/// </summary>

/// <value>The name of the duplicate.</value>

public bool? DuplicateName { get; set; }

/// <summary>

/// Gets or sets the length of the name invalid.

/// </summary>

/// <value>The length of the name invalid.</value>

public bool? NameInvalidLength { get; set; }

/// <summary>

/// Gets or sets the invalid business type code.

/// </summary>

/// <value>The invalid business type code.</value>

public bool? InvalidBusinessTypeCode { get; set; }

/// <summary>

/// Gets or sets the invalid business ubi.

/// </summary>

/// <value>The invalid business ubi.</value>

public bool? InvalidBusinessUbi { get; set; }

/// <summary>

/// Gets or sets the invalid location ubi.

/// </summary>

/// <value>The invalid location ubi.</value>

public bool? InvalidLocationUbi { get; set; }

/// <summary>

/// Gets or sets the invalid legal entity id.

/// </summary>

/// <value>The invalid legal entity id.</value>

public bool? InvalidLegalEntityId { get; set; }

/// <summary>

/// Gets or sets the invalid prior business id.

/// </summary>

/// <value>The invalid prior business id.</value>

public bool? InvalidPriorBusinessId { get; set; }

/// <summary>

/// Gets or sets the invalid detection id.

/// </summary>

/// <value>The invalid detection id.</value>

public bool? InvalidDetectionId { get; set; }

/// <summary>

/// Gets or sets the invalid first activity date.

/// </summary>

/// <value>The invalid first activity date.</value>

public bool? InvalidFirstActivityDate { get; set; }

/// <summary>

/// Gets or sets the invalid naic.

/// </summary>

/// <value>The invalid naic.</value>

public bool? InvalidNaic { get; set; }

/// <summary>

/// Gets or sets the length of the products and services invalid.

/// </summary>

/// <value>The length of the products and services invalid.</value>

public bool? ProductsAndServicesInvalidLength { get; set; }

/// <summary>

/// Gets or sets the invalid estimated gross income.

/// </summary>

/// <value>The invalid estimated gross income.</value>

public bool? InvalidEstimatedGrossIncome { get; set; }

/// <summary>

/// Gets or sets the invalid square footage.

/// </summary>

/// <value>The invalid square footage.</value>

public bool? InvalidSquareFootage { get; set; }

/// <summary>

/// Initializes a new instance of the <see cref="InvalidBusinessException"/> class.

/// </summary>

public InvalidBusinessException()

{

}

/// <summary>

/// Initializes a new instance of the <see cref="InvalidBusinessException"/> class.

/// </summary>

/// <param name="message">

/// The message.

/// </param>

public InvalidBusinessException(string message) : base(message)

{

}

/// <summary>

/// Initializes a new instance of the <see cref="InvalidBusinessException"/> class.

/// </summary>

/// <param name="message">The message.</param>

/// <param name="exception">The exception.</param>

public InvalidBusinessException(String message, System.Exception exception)

: base(message, exception)

{ }

/// <summary>

/// Initializes a new instance of the <see cref="InvalidBusinessException"/> class.

/// </summary>

/// <param name="info">The info.</param>

/// <param name="context">The context.</param>

/// <exception cref="T:System.ArgumentNullException">

/// The <paramref name="info"/> parameter is null.

/// </exception>

/// <exception cref="T:System.Runtime.Serialization.SerializationException">

/// The class name is null or <see cref="P:System.Exception.HResult"/> is zero (0).

/// </exception>

protected InvalidBusinessException(SerializationInfo info, StreamingContext context) : base(info, context)

{

}

/// <summary>

/// When overridden in a derived class, sets the <see cref="T:System.Runtime.Serialization.SerializationInfo"/> with information about the exception.

/// </summary>

/// <param name="info">

/// The <see cref="T:System.Runtime.Serialization.SerializationInfo"/> that holds the serialized object data about the exception being thrown.

/// </param>

/// <param name="context">

/// The <see cref="T:System.Runtime.Serialization.StreamingContext"/> that contains contextual information about the source or destination.

/// </param>

/// <exception cref="T:System.ArgumentNullException">

/// The <paramref name="info"/> parameter is a null reference (Nothing in Visual Basic).

/// </exception>

/// <PermissionSet>

/// <IPermission class="System.Security.Permissions.FileIOPermission, mscorlib, Version=2.0.3600.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" version="1" Read="\*AllFiles\*" PathDiscovery="\*AllFiles\*"/>

/// <IPermission class="System.Security.Permissions.SecurityPermission, mscorlib, Version=2.0.3600.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" version="1" Flags="SerializationFormatter"/>

/// </PermissionSet>

[SecurityPermission(SecurityAction.Demand, Flags = SecurityPermissionFlag.SerializationFormatter)]

public override void GetObjectData(SerializationInfo info, StreamingContext context)

{

base.GetObjectData(info, context);

info.AddValue("DuplicateName", DuplicateName);

}

}

}