Namespace Exchanger.Domain

Classes

[CasaDeCambio](#)

[Cliente](#)

[HistorialTransaccion](#)

[MedioDePago](#)

[Moneda](#)

[TasaDeCambio](#)

[Token](#)

[Token.TokenBuilder](#)

[Transaccion](#)

[TransaccionFactory](#)

[TransaccionMoneda](#)

Class CasaDeCambio

Namespace: [Exchanger](.).[Domain](.)

Assembly: Exchanger.Domain.dll

```
public class CasaDeCambio
```

Inheritance

[object](.) ← CasaDeCambio

Inherited Members

[object.Equals(object)](.) , [object.Equals(object, object)](.) , [object.GetHashCode()](.) , [object.GetType()](.) ,
[object.MemberwiseClone()](.) , [object.ReferenceEquals(object, object)](.) , [object.ToString()](.)

Methods

GetInstance()

```
public static CasaDeCambio GetInstance()
```

Returns

[CasaDeCambio](.)

Class Cliente

Namespace: [Exchanger](#).[Domain](#)

Assembly: Exchanger.Domain.dll

```
public class Cliente
```

Inheritance

[object](#) ← Cliente

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

Cliente(string, string, string?)

```
public Cliente(string nombreUsuario, string contraseña, string? token = null)
```

Parameters

nombreUsuario [string](#)

contraseña [string](#)

token [string](#)

Fields

historialTransacciones

```
public List<HistorialTransaccion> historialTransacciones
```

Field Value

List <HistorialTransaccion>

saldoPorMoneda

```
public Dictionary<string, double> saldoPorMoneda
```

Field Value

Dictionary <string, double >

Properties

MedioDePago

```
public MedioDePago? MedioDePago { get; set; }
```

Property Value

MedioDePago

NombreUsuario

```
public string? NombreUsuario { get; }
```

Property Value

string

Methods

Depositar(double, string)

```
public bool Depositar(double cantidad, string tipoMoneda)
```

Parameters

cantidad [double](#)

tipoMoneda [string](#)

Returns

[bool](#)

LimpiarHistorialTransacciones()

```
public void LimpiarHistorialTransacciones()
```

RealizarTransaccion(double, string, string, double)

```
public bool RealizarTransaccion(double cantidad, string monedaOrigen, string monedaDestino,
double cantidadConvertida)
```

Parameters

cantidad [double](#)

monedaOrigen [string](#)

monedaDestino [string](#)

cantidadConvertida [double](#)

Returns

[bool](#)

Retirar(double, string)

```
public bool Retirar(double cantidad, string tipoMoneda)
```

Parameters

`cantidad` [double](#)

`tipoMoneda` [string](#)

Returns

[bool](#)

VerHistorial()

```
public string VerHistorial()
```

Returns

[string](#)

VerSaldoActual()

```
public void VerSaldoActual()
```

VerificarCredenciales(string, string, string?)

```
public bool VerificarCredenciales(string nombreUsuario, string contraseña, string? token
= null)
```

Parameters

`nombreUsuario` [string](#)

`contraseña` [string](#)

`token` [string](#)

Returns

[bool](link)

Class HistorialTransaccion

Namespace: [Exchanger](#).[Domain](#)

Assembly: Exchanger.Domain.dll

```
public class HistorialTransaccion
```

Inheritance

[object](#) ← HistorialTransaccion

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

HistorialTransaccion(DateTime, double, string, string, double)

```
public HistorialTransaccion(DateTime fecha, double cantidad, string monedaOrigen, string
monedaDestino, double cantidadConvertida)
```

Parameters

`fecha` [DateTime](#)

`cantidad` [double](#)

`monedaOrigen` [string](#)

`monedaDestino` [string](#)

`cantidadConvertida` [double](#)

Properties

Cantidad

```
public double Cantidad { get; }
```

Property Value

[double](#)

CantidadConvertida

```
public double CantidadConvertida { get; }
```

Property Value

[double](#)

Fecha

```
public DateTime Fecha { get; }
```

Property Value

[DateTime](#)

MonedaDestino

```
public string MonedaDestino { get; }
```

Property Value

[string](#)

MonedaOrigen

```csharp
public string MonedaOrigen { get; }
```

Property Value

[string](#)

Class MedioDePago

Namespace: [Exchanger](#).[Domain](#)

Assembly: Exchanger.Domain.dll

```
public class MedioDePago
```

Inheritance

[object](#) ← MedioDePago

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

MedioDePago(string)

```
public MedioDePago(string cci)
```

Parameters

`cci` [string](#)

Properties

CCI

```
public string CCI { get; set; }
```

Property Value

[string](#)

Class Moneda

Namespace: [Exchanger](.).[Domain](.)

Assembly: Exchanger.Domain.dll

```
public class Moneda
```

Inheritance

[object](#) ← Moneda

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

Moneda(string)

```
public Moneda(string nombre)
```

Parameters

`nombre` [string](#)

Fields

EUR

```
public static Moneda EUR
```

Field Value

[Moneda](#)

GBP

```
public static Moneda GBP
```

Field Value

[Moneda](#)

PEN

```
public static Moneda PEN
```

Field Value

[Moneda](#)

USD

```
public static Moneda USD
```

Field Value

[Moneda](#)

Properties

Nombre

```
public string Nombre { get; }
```

Property Value

[string](#)⧉

Class TasaDeCambio

Namespace: [Exchanger](#).[Domain](#)

Assembly: Exchanger.Domain.dll

```csharp
public class TasaDeCambio
```

Inheritance

[object](#) ← TasaDeCambio

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Methods

GetInstance()

```csharp
public static TasaDeCambio GetInstance()
```

Returns

[TasaDeCambio](#)

ObtenerTasasDeCambio()

```csharp
public Dictionary<string, double> ObtenerTasasDeCambio()
```

Returns

[Dictionary](#)<[string](#), [double](#)>

Class Token

Namespace: [Exchanger.Domain](#)

Assembly: Exchanger.Domain.dll

```
public class Token
```

Inheritance

[object](#) ← Token

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

Token(string)

```
public Token(string value)
```

Parameters

`value` [string](#)

Properties

Value

```
public string Value { get; }
```

Property Value

[string](#)

Class Token.TokenBuilder

Namespace: [Exchanger](.)[.Domain](.)

Assembly: Exchanger.Domain.dll

```
public class Token.TokenBuilder
```

Inheritance

[object](.) ← Token.TokenBuilder

Inherited Members

[object.Equals(object)](.) , [object.Equals(object, object)](.) , [object.GetHashCode()](.) , [object.GetType()](.) ,
[object.MemberwiseClone()](.) , [object.ReferenceEquals(object, object)](.) , [object.ToString()](.)

Constructors

TokenBuilder()

```
public TokenBuilder()
```

Methods

AddRandomChars(int)

```
public Token.TokenBuilder AddRandomChars(int length)
```

Parameters

`length` [int](.)

Returns

[Token](.)[.TokenBuilder](.)

Build()

```
public Token Build()
```

Returns

[Token](#)

Class Transaccion

Namespace: [Exchanger](#).[Domain](#)

Assembly: Exchanger.Domain.dll

```
public abstract class Transaccion
```

Inheritance

[object](#) ← Transaccion

Derived

[TransaccionMoneda](#)

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Properties

Factory

```
public static TransaccionFactory Factory { get; }
```

Property Value

[TransaccionFactory](#)

Methods

RealizarConversion(double, string, string)

```
public abstract double RealizarConversion(double cantidad, string monedaOrigen,
string monedaDestino)
```

Parameters

cantidad `double`

monedaOrigen `string`

monedaDestino `string`

Returns

`double`

Class TransaccionFactory

Namespace: Exchanger.Domain

Assembly: Exchanger.Domain.dll

```
public class TransaccionFactory
```

Inheritance

object ← TransaccionFactory

Inherited Members

object.Equals(object) , [object.Equals(object, object)](object.Equals(object,%20object)) , object.GetHashCode() , object.GetType() ,
object.MemberwiseClone() , [object.ReferenceEquals(object, object)](object.ReferenceEquals(object,%20object)) , object.ToString()

Methods

CrearTransaccion(Dictionary<string, double>)

```
public Transaccion CrearTransaccion(Dictionary<string, double> tasasDeCambio)
```

Parameters

`tasasDeCambio` Dictionary<string, double>

Returns

Transaccion

Class TransaccionMoneda

Namespace: [Exchanger](#).[Domain](#)

Assembly: Exchanger.Domain.dll

```
public class TransaccionMoneda : Transaccion
```

Inheritance

[object](#) ← [Transaccion](#) ← TransaccionMoneda

Inherited Members

[Transaccion.Factory](#) , [object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

TransaccionMoneda(Dictionary<string, double>)

```
public TransaccionMoneda(Dictionary<string, double> tasasDeCambio)
```

Parameters

`tasasDeCambio` [Dictionary](#)<[string](#), [double](#)>

Methods

RealizarConversion(double, string, string)

```
public override double RealizarConversion(double cantidad, string monedaOrigen,
string monedaDestino)
```

Parameters

`cantidad` [double](#)

monedaOrigen `string`

monedaDestino `string`

Returns

double

Namespace Exchanger.Domain.Tests

Classes

CasaDeCambioTests

Class CasaDeCambioTests

Namespace: [Exchanger](#).[Domain](#).[Tests](#)

Assembly: Exchanger.Domain.Tests.dll

```csharp
[TestFixture]
public class CasaDeCambioTests
```

Inheritance

[object](#) ← CasaDeCambioTests

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Methods

Depositar_AddsToBalance()

```csharp
[Test]
public void Depositar_AddsToBalance()
```

Depositar_ReturnsFalse_WhenInvalidCurrency()

```csharp
[Test]
public void Depositar_ReturnsFalse_WhenInvalidCurrency()
```

RealizarTransaccion_AddsToHistorial_WhenSuccessful()

```csharp
[Test]
public void RealizarTransaccion_AddsToHistorial_WhenSuccessful()
```

RealizarTransaccion_ReturnsFalse_WhenInsufficientBalance()

```
[Test]
public void RealizarTransaccion_ReturnsFalse_WhenInsufficientBalance()
```

RealizarTransaccion_ReturnsFalse_WhenInvalidMonedaDestino()

```
[Test]
public void RealizarTransaccion_ReturnsFalse_WhenInvalidMonedaDestino()
```

RealizarTransaccion_ReturnsFalse_WhenInvalidMonedaOrigen()

```
[Test]
public void RealizarTransaccion_ReturnsFalse_WhenInvalidMonedaOrigen()
```

RealizarTransaccion_ReturnsTrue_WhenSufficientBalance()

```
[Test]
public void RealizarTransaccion_ReturnsTrue_WhenSufficientBalance()
```

VerHistorial_ReturnsCorrectFormat_WhenTransactionsExist()

```
[Test]
public void VerHistorial_ReturnsCorrectFormat_WhenTransactionsExist()
```

VerificarCredenciales_InvalidCredentials_ReturnsFalse()

```
[Test]
public void VerificarCredenciales_InvalidCredentials_ReturnsFalse()
```

VerificarCredenciales_NullTokenWithProvidedToken_Returns False()

```
[Test]
public void VerificarCredenciales_NullTokenWithProvidedToken_ReturnsFalse()
```

VerificarCredenciales_NullToken_ReturnsTrue()

```
[Test]
public void VerificarCredenciales_NullToken_ReturnsTrue()
```

VerificarCredenciales_ValidCredentials_ReturnsTrue()

```
[Test]
public void VerificarCredenciales_ValidCredentials_ReturnsTrue()
```