Advanced Databases CA

Maximiliano Herrera, 20103212

Contents

[Scope of the document 2](#_Toc70173646)

[Technical Design to include 2](#_Toc70173647)

[ER Diagram 2](#_Toc70173648)

[Physical Model in Third Normal Form (3NF) 3](#_Toc70173649)

[Data Dictionary 3](#_Toc70173650)

[Accounts (Click on table name to view the DDL related) 3](#_Toc70173651)

[Associations 3](#_Toc70173652)

[Customers 3](#_Toc70173653)

[ReferenceData 3](#_Toc70173654)

[Securities 4](#_Toc70173655)

[Technology used 4](#_Toc70173656)

[Embedded SQL File containing all DDL and DML statements. 4](#_Toc70173657)

[GitHub references to stored procedures and views 4](#_Toc70173658)

[Stored procedures 4](#_Toc70173659)

[Views 4](#_Toc70173660)

[Testing (test cases file on GitHub) 4](#_Toc70173661)

[Test case: Customer cannot be associated to another customer more than once under the same Association Type. 4](#_Toc70173662)

[Steps 4](#_Toc70173663)

[Expected result 5](#_Toc70173664)

[Test case: Visualize association details 5](#_Toc70173665)

[Steps 5](#_Toc70173666)

[Expected result 5](#_Toc70173667)

[Expected result 5](#_Toc70173668)

[Test case: Create customer and association 6](#_Toc70173669)

[Pre-conditions 6](#_Toc70173670)

[Steps 6](#_Toc70173671)

[Expected result 6](#_Toc70173672)

[Test case: Delete a customer and foreign key records 6](#_Toc70173673)

[Pre-conditions 6](#_Toc70173674)

[Steps 7](#_Toc70173675)

[Expected result 7](#_Toc70173676)

[Test case: Validate soft delete on ReferenceData. 7](#_Toc70173677)

[Steps 7](#_Toc70173678)

[Expected result 8](#_Toc70173679)

[Test case: No modifications can be done on underlying schema of view vwCustomersAssociations 8](#_Toc70173680)

[Steps 8](#_Toc70173681)

[Expected result 8](#_Toc70173682)

[Data Governance & Security 8](#_Toc70173683)

[Data integrity 8](#_Toc70173684)

[Entity Integrity 8](#_Toc70173685)

[Referential Integrity 8](#_Toc70173686)

[Domain Integrity 8](#_Toc70173687)

[Consistency 8](#_Toc70173688)

[Security 8](#_Toc70173689)

[Reflection on Learnings 9](#_Toc70173690)

[References 9](#_Toc70173691)

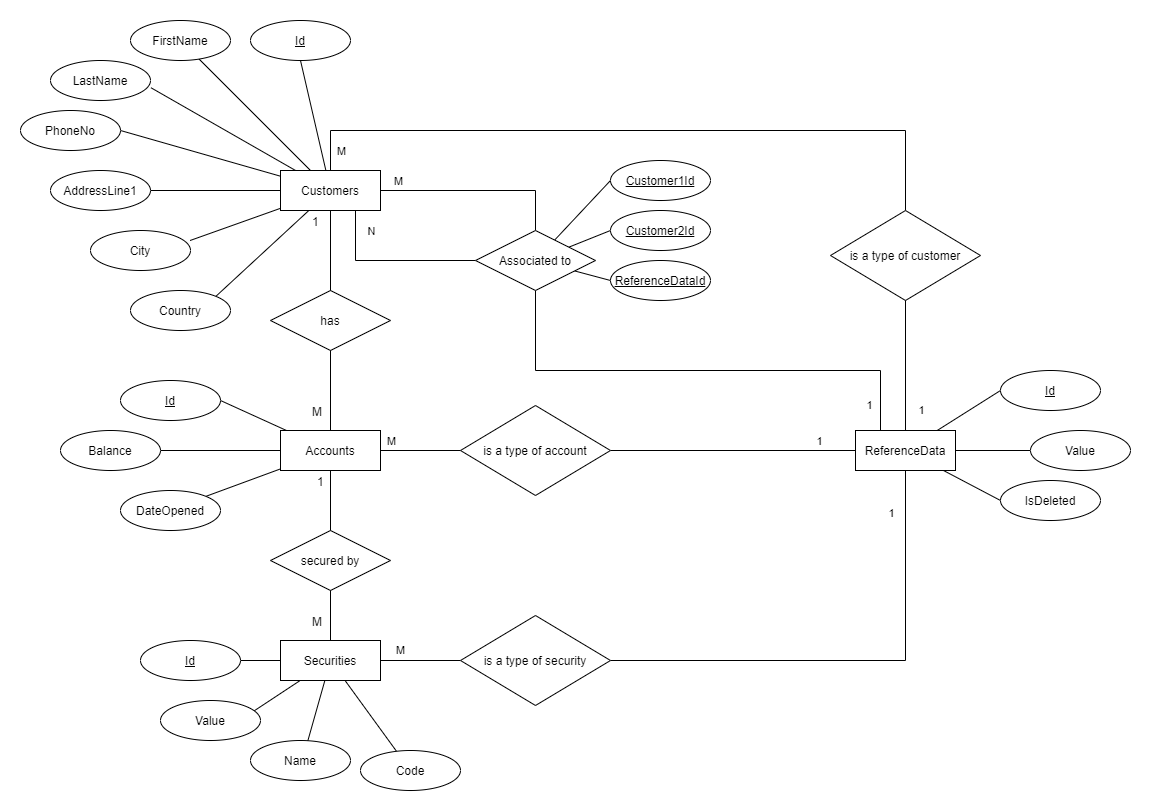
# 

# Scope of the document

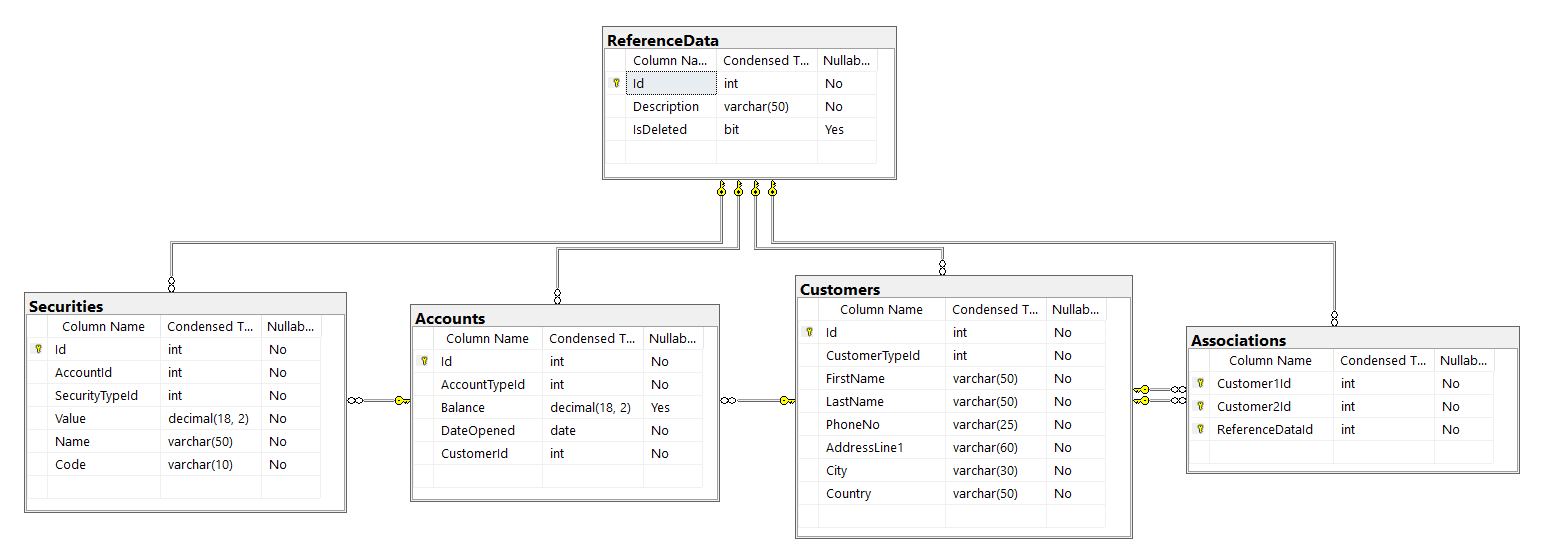
The scope of this document is the design and development of a single database to consolidate data, which is stored in multiple systems.

# Technical Design to include

## ER Diagram



## Physical Model in Third Normal Form (3NF)



## Data Dictionary

### [Accounts](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L78) (Click on table name to view the DDL related)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column name | Data type | Primary Key | Foreign Key | Not null | Reference to | Constraint |
| AccountTypeId | int |  | x | x | ReferenceData(Id) |  |
| Balance | decimal(18,2) |  |  |  |  |  |
| CustomerId | int |  | x | x | Customers(Id) |  |
| DateOpened | date |  |  | x |  |  |
| Id | int | x |  | x |  |  |

### [Associations](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L95)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column name | Data type | Primary Key | Foreign Key | Not null | Reference to | Constraint |
| Customer1Id | int | x | x | x | Customers(id) |  |
| Customer2Id | int | x | x | x | Customers(id) |  |
| ReferenceDataId | int | x | x | x | ReferenceData(Id) |  |

### [Customers](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L112)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column name | Data type | Primary Key | Foreign Key | Not null | Reference to | Constraint |
| AddressLine1 | varchar(60) |  |  | x |  |  |
| City | varchar(30) |  |  | x |  |  |
| Country | varchar(50) |  |  | x |  |  |
| CustomerTypeId | int |  | x | x | ReferenceData(Id) |  |
| FirstName | varchar(50) |  |  | x |  |  |
| Id | int | x |  | x |  |  |
| LastName | varchar(50) |  |  | x |  |  |
| PhoneNo | varchar(25) |  |  | x |  |  |

### [ReferenceData](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L132)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column name | Data type | Primary Key | Foreign Key | Not null | Reference to | Constraint |
| Description | varchar(50) |  |  | x |  |  |
| Id | int | x |  | x |  |  |
| IsDeleted | bit |  |  |  |  |  |

### [Securities](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L147)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Column name | Data type | Primary Key | Foreign Key | Not null | Reference to | Constraint |
| AccountId | int |  | x | x | Accounts(id) |  |
| Id | int | x |  | x |  |  |
| Name | varchar(50) |  |  | x |  |  |
| SecurityTypeId | int |  | x | x | ReferenceData(Id) |  |
| Value | decimal(18,2) |  |  | x |  |  |
| Code | varchar(10) |  |  | x |  |  |

## Technology used

* Microsoft SQL Server: Used as relational database management system. Version: Microsoft SQL Server 2014 - 12.0.2269.0 (X64).
* SQL Server Management Studio (v18.8): Integrated environment for managing SQL database and generating physical entity relationship diagram.
* Draw.io: Used for drawing conceptual entity relationship diagram.

## Embedded SQL File containing all DDL and DML statements.

* 1. 

## GitHub references to stored procedures and views

### Stored procedures

* [CreateCustomerAndAssociation](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L482)
* [DeleteCustomer](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L510)

### Views

* [vwCustomersAssociations](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L165)
* [vwReferenceDataDeleted](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/SQL%20Script%20-%20DDL%20and%20DML.sql#L186)

# [Testing](https://github.com/herreramaxi/NCIRL-AD-CA/blob/6b20d3bfe711335a2b7b066ab13c3bef5e829d69/Test%20cases.sql#L1) (test cases file on GitHub)

NB: The test cases file is no needed for below test cases, but it is available just for make the tests execution faster and easier.

## Test case: Customer cannot be associated to another customer more than once under the same Association Type.

Validate that a customer cannot be associated to another customer more than once under the same Association Type.

### Steps

1. Execute the following inserts:

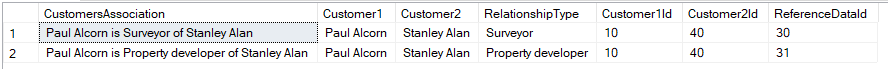
INSERT [dbo].[Associations] ([Customer1Id], [Customer2Id], [ReferenceDataId]) VALUES (10, 40, 30)

INSERT [dbo].[Associations] ([Customer1Id], [Customer2Id], [ReferenceDataId]) VALUES (10, 40, 31)

1. Execute below query:

select \* from dbo.vwCustomersAssociations where [Customer1Id] = 10

* 1. Expected result:

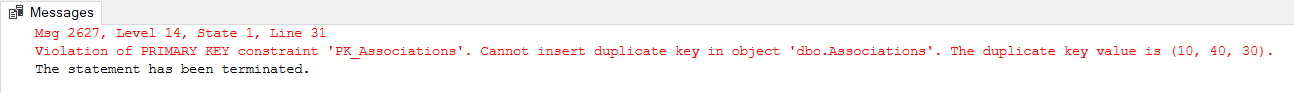


1. Execute below insert:

INSERT [dbo].[Associations] ([Customer1Id], [Customer2Id], [ReferenceDataId]) VALUES (10, 40, 30)

### Expected result

An error message should be shown when trying to execute the insert as there is a composite primary key (by Customer1Id, Customer2Id and ReferenceDataId) on table Associations which prevents the duplication of association type for a tuple of customers.



## Test case: Visualize association details

Validate that the view vwCustomersAssociations returns the association details between all clients showing both sides of the relationship.

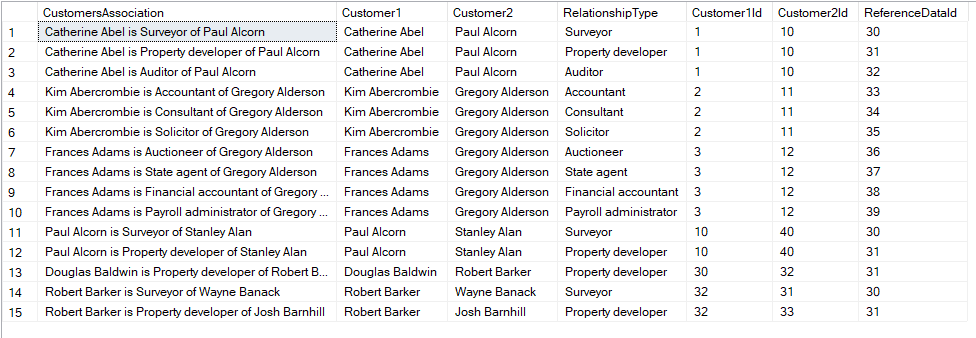
### Steps

1. Execute following query:

SELECT \* FROM dbo.vwCustomersAssociations

### Expected result

The above query should return the following records:



1. Execute following insert

INSERT [dbo].[Associations] ([Customer1Id], [Customer2Id], [ReferenceDataId]) VALUES (32, 34, 35)

1. Execute following query:

SELECT \* FROM dbo.vwCustomersAssociations

### Expected result

The new association should be shown as below:



## Test case: Create customer and association

Validate that stored procedure CreateCustomerAndAssociation creates a new customer and association to customerId2 of AssociationTypeId.

### Pre-conditions

The following query should return no records:

SELECT \* FROM dbo.vwCustomersAssociations where Customer2Id = 20

### Steps

1. Execute the following stored procedure:

exec CreateCustomerAndAssociation

@FirstName = 'Wanda',

@LastName = 'Vernon',

@AddressLine1 = 'Belgrove Park',

@City = 'Dublin 5',

@Country = 'Ireland',

@PhoneNo = '0878940684',

@CustomerTypeId = 20,

@CustomerId2 = 20,

@AssociationTypeId = 30

### Expected result

The following query should return one record:

SELECT \* FROM dbo.vwCustomersAssociations where Customer2Id = 20



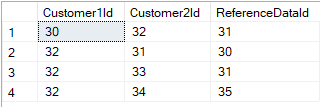
## Test case: Delete a customer and foreign key records

Validate that the stored procedure DeleteCustomer deletes a customer and all related entities, which are: Associations, Accounts, Securities and Customers.

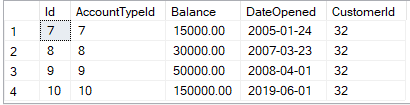
### Pre-conditions

The following queries should return following records:

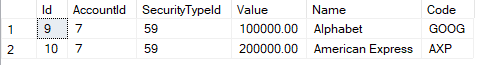
SELECT \* FROM Associations WHERE Customer1Id = 32 or Customer2Id = 32



SELECT \* FROM Accounts WHERE CustomerId = 32



SELECT \* FROM Securities WHERE AccountId IN (SELECT id FROM Accounts WHERE CustomerId = 32)



SELECT \* FROM Customers WHERE id = 32



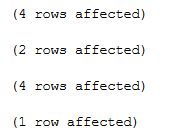
### Steps

1. Execute the below stored procedure:

exec DeleteCustomer @CustomerId= 32

### Expected result

* The stored procedure is executed successfully, and the rows affected are as follow:



* If the queries from Pre-conditions are executed , no records should be returned by the queries.

SELECT \* FROM Associations WHERE Customer1Id = 32 or Customer2Id = 32

SELECT \* FROM Accounts WHERE CustomerId = 32

SELECT \* FROM Securities WHERE AccountId IN (SELECT id FROM Accounts WHERE CustomerId = 32)

SELECT \* FROM Customers WHERE id = 32

## Test case: Validate soft delete on ReferenceData.

Validate that view vwReferenceDataDeleted returns only ReferenceData records where isDeleted is true.

### Steps

1. Execute following query:

select \* from dbo.vwReferenceDataDeleted

* 1. Expected result: No records are returned.

1. Execute following query:

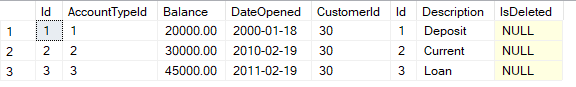
SELECT \*

FROM Accounts a

INNER JOIN ReferenceData r ON r.id = a.AccountTypeId

WHERE a.id IN (1,2,3)

* 1. Expected result: The following records are returned and the flag isDeleted is NULL.



1. Execute following update:

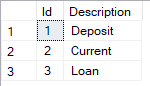
UPDATE dbo.ReferenceData SET IsDeleted = 1 WHERE id IN (1,2,3)

1. Execute following query:

select \* from dbo.vwReferenceDataDeleted

### Expected result

The following records are returned:



## Test case: No modifications can be done on underlying schema of view vwCustomersAssociations

The view vwCustomersAssociations is secured by attribute SCHEMABINDING to prevent modification on underlying schema.

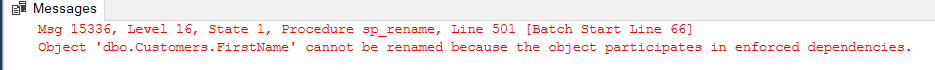
### Steps

1. Execute the query:

EXEC sp\_rename 'dbo.Customers.FirstName', 'FirstNameNew', 'COLUMN';

### Expected result

An error message as below is shown and the rename is not executed.



# Data Governance & Security

## Data integrity

The data model provides data integrity by using constraints. There are three levels of integrity that the data model is providing:

### Entity Integrity

* Primary key
* Not Null

### Referential Integrity

* Foreign Key

### Domain Integrity

* Default

## Consistency

Consistency is achieved by using the stored procedures “CreateCustomerAndAssociation” and “DeleteCustomer” when managing creation and deletion of customers. Internally those store procedures manage transactions to prevent any inconsistency when inserting or deleting to/from multiple tables.

## Security

* The view vwCustomersAssociations is secured by using the attribute SCHEMABINDING, so that modifications of underlying schema, which are related to the view, are not allowed.
* The stored procedures “CreateCustomerAndAssociation” and “DeleteCustomer” provide a level of security by centralising the operations of adding and removing customers. This is analogue to encapsulation on Object oriented programming in where encapsulation refers to the bundling of data with the methods that operate on that data, or the restricting of direct access to some of an object's components.

# Reflection on Learnings

Working on the CA I was able to reinforce my knowledge on database concepts such as stored procedures, views, transactions, data manipulation language as well as data definition language. It was useful to read below reference links specially those related to GDPR and hard vs soft delete.

# References

Wikipedia Contributors (2019). *Microsoft SQL Server*. [online] Wikipedia. Available at: <https://en.wikipedia.org/wiki/Microsoft_SQL_Server> [Accessed 21 Apr. 2021].

markingmyname (n.d.). *SQL Server Management Studio (SSMS) - SQL Server Management Studio (SSMS)*. [online] docs.microsoft.com. Available at: <https://docs.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver15> [Accessed 21 Apr. 2021].

Wikipedia. (2020). *Third normal form*. [online] Available at: <https://en.wikipedia.org/wiki/Third_normal_form> [Accessed 21 Apr. 2021].

WilliamDAssafMSFT (n.d.). *CREATE VIEW (Transact-SQL) - SQL Server*. [online] docs.microsoft.com. Available at: <https://docs.microsoft.com/en-us/sql/t-sql/statements/create-view-transact-sql?view=sql-server-ver15> [Accessed 21 Apr. 2021].

‌cybersecurity.att.com. (n.d.). *Data security governance explained*. [online] Available at: <https://cybersecurity.att.com/blogs/security-essentials/data-governance.at-the-heart-of-security-privacy-and-risk#:~:text=Data%20governance%20is%20the%20capability>

[Accessed 21 Apr. 2021].

‌

Kenton, W. (2019). *Security*. [online] Investopedia. Available at: <https://www.investopedia.com/terms/s/security.asp> [Accessed 21 Apr. 2021].

‌LIOUDIS, N. (2020). *Common Examples of Marketable Securities*. [online] www.investopedia.com. Available at: <https://www.investopedia.com/ask/answers/033015/what-are-some-common-examples-marketable-securities.asp> [Accessed 21 Apr. 2021].

Stack Overflow. (n.d.). *sql - Best way to get identity of inserted row?* [online] Available at: <https://stackoverflow.com/questions/42648/best-way-to-get-identity-of-inserted-row>

[Accessed 21 Apr. 2021].

‌Treacy, M. (2020). *Hard vs Soft delete user data — forget-me or forget-me-not???* [online] Medium. Available at: <https://medium.com/@mtreacy002/hard-vs-soft-delete-user-data-forget-me-or-forget-me-not-e5b564363607> [Accessed 21 Apr. 2021].

‌Stack Overflow. (n.d.). *sql - Physical vs. logical (hard vs. soft) delete of database record?* [online] Available at: <https://stackoverflow.com/questions/378331/physical-vs-logical-hard-vs-soft-delete-of-database-record> [Accessed 21 Apr. 2021].

‌Law Stack Exchange. (n.d.). *data protection - Is it allowed to Psuedonym or Soft delete the User Records in a highly complex User centric Software under GDPR?* [online] Available at: <https://law.stackexchange.com/questions/29078/is-it-allowed-to-psuedonym-or-soft-delete-the-user-records-in-a-highly-complex-u> [Accessed 21 Apr. 2021].

‌

General Data Protection Regulation (GDPR). (n.d.). *Recital 26 - Not Applicable to Anonymous Data*. [online] Available at: <https://gdpr-info.eu/recitals/no-26/> [Accessed 21 Apr. 2021].

Wikipedia Contributors (2019). *Encapsulation (computer programming)*. [online] Wikipedia. Available at: <https://en.wikipedia.org/wiki/Encapsulation_(computer_programming)>

www.plus500.ie. (n.d.). *Online CFD Trading | Trading the Markets | Plus500 Ireland*. [online] Available at: <https://www.plus500.ie/> [Accessed 24 Apr. 2021].

‌

‌