

Alba Márquez Rodríguez

2nd Report

Online Videogames Store Database

Database System Engineering – Computer Science

The project will consists on an Online Videogames Store Database

Professor: Krystian Wojtkiewicz



Contents

[DDL **Error! Bookmark not defined.**](#_Toc91332508)

[Create **Error! Bookmark not defined.**](#_Toc91332509)

[Alter **Error! Bookmark not defined.**](#_Toc91332510)

[Drop 2](#_Toc91332511)

[Data Dictionary 3](#_Toc91332512)

[Game 3](#_Toc91332513)

[Sales 9](#_Toc91332514)

[Person 16](#_Toc91332515)

[Database Populating 18](#_Toc91332516)

[CRUD Operations 39](#_Toc91332517)

[Additional Indexes 52](#_Toc91332518)

[List of Applications views 53](#_Toc91332519)

[Interface State Diagram 54](#_Toc91332520)

[Mock-up for each interface view 55](#_Toc91332521)

[Comments on the interface design principles used while preparing mock-ups. **Error! Bookmark not defined.**](#_Toc91332522)

[Bibliography 59](#_Toc91332523)

## Drop

To eliminate all the tables:

DROP TABLE IF EXISTS `onlinevideogamestore`.`address`, `onlinevideogamestore`.`addresstype`, `onlinevideogamestore`.`cardtype`, `onlinevideogamestore`.`contact`, `onlinevideogamestore`.`contactcreditcard`, `onlinevideogamestore`.`countryregion`, `onlinevideogamestore`.`creditcard`, `onlinevideogamestore`.`currency`, `onlinevideogamestore`.`currencyrate`, `onlinevideogamestore`.`customer`, `onlinevideogamestore`.`game`, `onlinevideogamestore`.`game\_shoppingcartitem`, `onlinevideogamestore`.`gameaudiovisuals`, `onlinevideogamestore`.`gamelistpricehistory`, `onlinevideogamestore`.`gamephoto`, `onlinevideogamestore`.`gamereview`, `onlinevideogamestore`.`gamevideo`, `onlinevideogamestore`.`genre`, `onlinevideogamestore`.`saga`, `onlinevideogamestore`.`salesorderdetail`, `onlinevideogamestore`.`salesorderheader`, `onlinevideogamestore`.`salesterritory`, `onlinevideogamestore`.`shoppingcartitem`, `onlinevideogamestore`.`specialoffer`, `onlinevideogamestore`.`specialoffergame`, `onlinevideogamestore`.`stateprovince`, `onlinevideogamestore`.`territory`, `onlinevideogamestore`.`title`,`onlinevideogamestore`.`unitmeasure`,`onlinevideogamestore`.`countryregioncurrency`;

# Data Dictionary

## Game

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Game | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GameID | int | U | Primary key for game records.  Identity / Auto increment column |
|  | Name | varchar(50) | U | Name of the game |
|  | GameNumber | varchar(25) | U | Unique game identification number. |
|  | StandardCost | Decimal(13,4) |  | Standard cost of the game.  Default: 0.00 |
|  | ListPrice | Decimal(13,4) |  | Selling price. |
|  | Storage | varchar(5) | N | Product storage. |
|  | StorageUnitMeasureCode | char(3) | N | Unit of measure for Storage column. |
| F | GenreID | int | N | Game is a member of this game genre.  Foreign key to ProductSubCategory.ProductSubCategoryID. |
| F | PlatformID | Int |  | Game is a member of this game Platform. |
|  | Description | Varchar(255) |  | Description of the game |
| F | SagaID | Int |  | Game is a member of this game saga. |
| F | GameReviewID | int |  |  |
|  | SellStartDate | datetime |  | Date the game was available for sale. |
|  | SellEndDate | datetime | N | Date the game was no longer available for sale. |
|  | LaunchDate | datetime | N | Date the game was launched. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated.  Default: getdate() //current\_timestamp |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Genre | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GenreID | int | U | Primary key for genre game records.  Identity/Auto increment column. |
|  | Name | varchar(50) | U | Name of the game. |
|  | Description | varchar(255) | N | Game description. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GameAudioVisuals | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GameID | int | U | Primary key for gameaudiovisuals records.  Identity/Auto increment column. |
| F | GamePhotoID | int | N/U | Foreign key to GamePhoto.GamePhotoID. |
| F | GameVideoID | int | N/U | Foreign key to GameVideo.GameVideoID. |
|  | ModifiedDate | Datetime |  | Date and time the record was last updated.  Default: getdate()/current\_timestamp |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GamePhoto | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GamePhotoID | int | U | Primary key for GamePhoto records.  Identity/Auto increment column. |
|  | Icon | varchar(50) | N | Icon file. |
|  | Wallpaper | varchar(50) | N | Wallpaper file. |
|  | Header | varchar(50) | N | Header file. |
|  | Preview | varchar(50) | N | Preview file. |
|  | ModifiedDate | Datetime |  | Date and time the record was last updated.  Default: getdate()/current\_timestamp |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GameVideo | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GameVideoID | int | U | Primary key for GamePhoto records.  Identity/Auto increment column. |
|  | Icon | varchar(50) | N | Icon file. |
|  | Wallpaper | varchar(50) | N | Wallpaper file. |
|  | Header | varchar(50) | N | Header file. |
|  | Preview | varchar(50) | N | Preview file. |
|  | ModifiedDate | Datetime |  | Date and time the record was last updated.  Default:getdate()/current\_timestamp |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GameVideo | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GameVideoID | int | U | Primary key for GameVideo records.  Identity/Auto increment column. |
|  | ReviewVideo | varchar(50) | N | Review video file. |
|  | PreviewVideo | varchar(50) | N | Preview Video file. |
|  | ModifiedDate | Datetime |  | Date and time the record was last updated.  Default:getdate()/current\_timestamp |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GameReview | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GameReviewID | int | U | Primary key for GameReview records.  Identity / Auto increment column |
| F | GameID | int |  | Game identification number. Foreign key to Game.GameID. |
|  | ReviewerName | varchar(50) |  | Name of the reviewer. |
|  | ReviewDate | datetime |  | Date review was submitted. Default: getdate() |
|  | EmailAddress | nvarchar(50) |  | Reviewer's e-mail address. |
|  | Rating | int |  | Game rating given by the reviewer. Scale is 1 to 5 with 5 as the highest rating. |
|  | Comments | nvarchar(3850) | N | Reviewer's comments |
|  | ModifiedDate | datetime |  | Date and time the record was last updated.  Default:getdate/current\_timestamp |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UnitMeasure | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | UnitMeasureCode | nchar(3) | U | Primary key. |
|  | Name | nvarchar(50) | U | Unit of measure description/name. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated.  Default:getdate/current\_timestamp |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GameListPriceHistory | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GameID | int | U | Game identification number. Foreign key to Game.GameID |
| P | StartDate | datetime | U | List price start date. |
|  | EndDate | datetime | N | List price end date |
|  | ListPrice | Decimal(13,4) |  | Product list price. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated.  Default: getdate()/current\_timestamp |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Game\_ShoppingCartItem | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | GameID | int |  | Game identification number. Foreign key to Game.GameID |
| P | ShoppingCartItemID | int |  | ShoppingCartItem identification number. Foreign key to ShoppingCartItem. ShoppingCartItemID |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Saga | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | SagaID | int | U | Saga identification number. |
|  | Name | varchar(50) |  | Saga name. |
|  | Quantity | int |  | Quantity of games of the saga. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated.  Default: getdate()/current\_timestamp |

## Sales

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SalesOrderHeader | | | | | |
| P/F | **Field Name** | **Data Type** | **Field Size** | **Null/Unique** | **Description/Attributes** |
| P | SalesOrderID | int |  | U | Primary key  Identity / Auto increment column |
|  | RevisionNumber | tinyint |  |  | Incremental number to track changes to the sales order over time.  Default: 0 |
|  | OrderDate | datetime |  |  | Dates the sales order was created.  Default: getdate() |
|  | DueDate | datetime |  |  | Date the order is due to the customer |
|  | Status | tinyint |  |  | Order current status. 1 = In process;  2 = Approved;  3 = Backordered;  4 =Rejected;  5 = Cancelled  Default: 1 |
|  | SalesOrderNumber | varchar(25) |  | N/U | Unique sales order identification number.Computed: isnull(N'SO'+CONVERT([nvarchar](23),[SalesOrderID]),N'\*\*\*ERROR \*\*\*') |
|  | AccountNumber | varchar(15) |  | N | Financial accounting number reference |
| F | CustomerID | Int |  |  | Customer identification number.  Foreign key to Customer.BusinessEntityID |
| F | ContactID | Int |  |  | Contact identification number.  Foreign key to Contact.ContactID |
| F | TerritoryID | Int |  | N | Territory in which the sale was made.  Foreign key to Territory.TerritoryID |
| F | BillToAddressID | Int |  |  | Customer billing address.  Foreign key to Address.AddressID. |
|  | CreditCardApprovalCode | Varchar | 15 | N | Approval code provided by the credit card company |
|  | SubTotal | decimal(13, 4) |  |  | Sales subtotal. Computed as SUM(SalesOrderDetail.LineTotal)for the appropriate SalesOrderID.  Default: 0.00 |
|  | TaxAmt | decimal(13, 4) |  |  | Tax amount.  Default: 0.00 |
|  | Comment | varchar(128) | 128 | N | Sales representative comments. |
|  | ModifiedDate | Datetime |  |  | Date and time the record was last updated.  Default: getdate() |
|  | CreditCardID | Int |  | N | Credit card identification number. Foreign key to CreditCard.CreditCardID |
| F | CurrencyRateID | Int |  | N | Currency exchange rate used. Foreign key to CurrencyRate.CurrencyRateID. |
|  | ContactCreditCardCreditCardID | int |  |  |  |

Triggers: uSalesOrderHeader -> After Update -> AFTER UPDATE trigger that updates the RevisionNumber and ModifiedDate columns in the SalesOrderHeader table.Updates the SalesYTD column in the SalesPerson and SalesTerritory tables.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CountryRegionCurrency | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | CountryRegionCode | varchar(3) |  | ISO code for countries and regions.  Foreign key to CountryRegion.CountryRegionCode. |
| P | CurrencyCode | char(3) |  | ISO standard currency code.  Foreign key to Currency.CurrencyCode. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CreditCard | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | CreditCardID | int | U | Primary key for CreditCard records. Identity / Auto increment column |
|  | CardType | varchar(50) |  | Credit card name. |
|  | CardNumber | varchar(25) |  | Credit card number. |
|  | ExpMonth | tinyint |  | Credit card expiration month. |
|  | ExpYear | smallint |  | Credit card expiration year. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CardType | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | CreditCardID | int | U | Primary key for CardType records. Identity / Auto increment column |
|  | Name | Varchar(25) | U | Credit card name. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated.  Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Currency | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | CurrencyCode | char(3) | U | The ISO code for the Currency. |
|  | Name | varchar(50) | U | Currency name. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated.  Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CurrencyRate | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | CurrencyRateID | int | U | Primary key for CurrencyRate records. Identity / Auto increment column |
|  | CurrencyRateDate | datetime |  | Date and time the exchange rate was obtained. |
|  | FromCurrencyCode | char(3) |  | Exchange rate was converted from this currency code. |
|  | ToCurrencyCode | char(3) |  | Exchange rate was converted to this currency code. |
|  | AverageRate | Decimal(13,4) |  | Average exchange rate for the day. |
|  | EndOfDayRate | Decimal(13,4) |  | Final exchange rate for the day. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated.  Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ContactCreditCard | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | ContactID | int | U | Primary key.  ID of the Contact  Identity / Auto increment column |
| P | CreditCardID | int | U | PrimaryKey.  ID of the CreditCard of the Contact. |
|  | ModifiedDate | date |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Customer | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | CustomerID | int | U | Primary key.  Identity / Auto increment column |
|  | TerritoryID | int | N | ID of the territory in which the customer is located. Foreign key to SalesTerritory.SalesTerritoryID. |
|  | AccountNumber | varchar(10) | U | Unique number identifying the customer assigned by the accounting system.  Computed: isnull('AW'+[ufnLeadingZeros]([CustomerID]),'') |
|  | ContactID | int |  | ID of the contact.  Foreign key to SalesTerritory.SalesTerritoryID. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SalesOrderDetail | | | | |
| P/F | **Field Name** | **Data Type** | **Null/Unique** | **Description/Attributes** |
| P | SalesOrderID | int |  | Primary key. Foreign key to SalesOrderHeader.SalesOrderID. |
| P | SalesOrderDetailID | int |  | Primary key. One incremental unique number per product sold. Identity / Auto increment column |
|  | CarrierTrackingNumber | varchar(25) | N | Shipment tracking number supplied by the shipper. |
|  | OrderQty | smallint |  | Quantity ordered per product. |
|  | ProductID | int |  | Product sold to customer. Foreign key to Product.ProductID. |
|  | SpecialOfferID | int |  | Promotional code. Foreign key to SpecialOffer.SpecialOfferID. |
|  | UnitPrice | Decimal(13,4) |  | Selling price of a single product. |
|  | UnitPriceDiscount | Decimal(13,4) |  | Discount amount. Default: 0.0 |
|  | SalesOrderID | int |  | Primary key. Foreign key to SalesOrderHeader.SalesOrderID. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SalesOrderHeader | | | | | |
| P/F | Field Name | Data Type | Field Size | Null/Unique | Description/Attributes |
| P | TerritoryID | int |  | U | Primary key for SalesTerritory records. Identity / Auto increment column |
|  | Name | nvarchar | 50 |  | Sales territory name |
|  | CountryRegionCode | nvarchar | 3 |  | ISO standard country or region code. Foreign key toCountryRegion.CountryRegionCode |
|  | Group | nvarchar | 50 |  | Geographic area to which the sales territory belong |
|  | Status | Int (tinyint) |  |  | Order current status. 1 = In process;  2 = Approved;  3 = Backordered;  4 =Rejected;  5 = Cancelled  Default: 1 |
|  | SalesOrderNumber | nvarchar | 25 | N/U | Unique sales order identification number.Computed: isnull(N'SO'+CONVERT([nvarchar](23),[SalesOrderID]),N'\*\*\*ERROR \*\*\*') |
|  | AccountNumber | Nvarchar | 15 | N | Financial accounting number reference |
| F | CustomerID | Int |  |  | Customer identification number. Foreign key to Customer.BusinessEntityID |
| F | ContactID | Int |  |  |  |
| F | TerritoryID | Int |  | N | Territory in which the sale was made.  Foreign key to Territory. TerritoryID |
| F | BillToAddressID | Int |  |  | Customer billing address. Foreign key to Address.AddressID. |
|  | CreditCardApprovalCode | Varchar | 15 | N | Approval code provided by the credit card company |
|  | SubTotal | Money |  |  | Sales subtotal. Computed as SUM(SalesOrderDetail.LineTotal)for the appropriate SalesOrderID.  Default: 0.00 |
|  | TaxAmt | Money |  |  | Tax amount.  Default: 0.00 |
|  | Comment | Nvarchar | 128 | N | Sales representative comments. |
|  | ModifiedDate | Datetime |  |  | Date and time the record was last updated.  Default: getdate() |
|  | CreditCardID | Int |  | N | Credit card identification number. Foreign key to CreditCard.CreditCardID |
| F | CurrencyRateID | Int |  | N | Currency exchange rate used. Foreign key to CurrencyRate.CurrencyRateID. |
|  | ContactCreditCardCreditCardID | int |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SalesTerritory | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | TerritoryID | int |  | Primary key for SalesTerritory records. Identity / Auto increment column |
|  | Name | varchar(50) |  | Sales territory description |
|  | CountryRegionCode | varchar(3) |  | ISO standard country or region code.  Foreign key to CountryRegion.CountryRegionCode. |
|  | Group | varchar(50) |  | Geographic area to which the sales territory belong. |
|  | SalesYTD | DECIMAL(13, 4) |  | Sales in the territory year to date. Default: 0.00 |
|  | SalesLastYear | DECIMAL(13, 4) |  | Sales in the territory the previous year. Default: 0.00 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ShoppingCartItem | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | ShoppingCartItemID | int |  | Primary key for ShoppingCartItem records. Identity / Auto increment column |
|  | ShoppingCartID | varchar(50) |  | Shopping cart identification number. |
|  | GameID | int |  | Game ordered.  Foreign key to Game.GameID. |
|  | DateCreated | date |  | Date the time the record was created.  Default: getdate() |
|  | ModifiedDate | date |  | Date and time the record was last updated.  Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SpecialOffer | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | SpecialOfferID | int |  | Primary key for SpecialOffer records. Identity / Auto increment column |
|  | Description | varchar(255) |  | Discount description. |
|  | Discount | DECIMAL(13, 4) |  | Discount precentage. Default: 0.00 |
|  | Type | varchar(50) |  | Discount type category. |
|  | StartDate | date |  | Discount start date. |
|  | EndDate | date |  | Discount end date. |
|  | ModifiedDate | date |  | Date and time the record was last updated.  Default: getdate() |
|  | SpecialOfferProductGameID | int |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SpecialOfferGame | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | SpecialOfferID | int | U | Primary key for SpecialOfferGame records. |
| P | GameID | varchar(255) |  | Game of the special offer.  Foreign key to Game.GameID. |
|  | ModifiedDate | date |  | Date and time the record was last updated.  Default: getdate() |

## Person

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SpecialOfferGame | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | AddressID | int | U | Primary key for Address records.  Identity / Auto increment column |
| F | AddressTypeID | Int |  | Foreign key for AddressType |
|  | AddressLine1 | varchar(60) | N | First street address line. |
|  | AddressLine2 | varchar(60) | N | Second street address line. |
|  | City | varchar(30) |  | Name of the city. |
|  | StateProvinceID | int |  | Unique identification number for the state or province. Foreign key to StateProvince table. |
|  | PostalCode | nvarchar(15) |  | Postal code for the street address. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AddressType | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | AddressTypeID | int | U | Primary key for AddressType records.  Identity / Auto increment column |
| F | Name | Varchar(155) | U | Name of the Address Type |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Contact | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | ContactID | int | U | Primary key for Contact records.  Identity / Auto increment column |
| F | TitleID | Int |  | Foreign key for Title |
|  | FirstName | varchar(255) |  | First Name of the contact |
|  | MiddleName | varchar(255) | N | Middle Name of the contact |
|  | LastName | varchar(255) |  | LastName of the contact |
|  | EmailAddress | Varchar(255) | U | Email address of the contact |
|  | Phone | int |  | Phone of the contact |
|  | Password | int |  | Password ID of the contact |
|  | AdditionalContactInfo | Varchar(255) | N |  |
|  | AddressID | Int |  | ForeignKey for Address |
|  | ModifiedDate | datetime |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CountryRegion | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | CountryRegionCode | varchar(3) | U | ISO standard code for countries and regions. |
|  | Name | varchar(50) | U | Country or region name. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| StateProvince | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | StateProvinceID | int |  | Primary key for StateProvince records. Identity / Auto increment column |
| F | StateProvinceCode | char(3) |  | ISO standard state or province code. |
|  | CountryRegionCode | varchar(3) |  | ISO standard country or region code. Foreign key to CountryRegion.CountryRegionCode. |
|  | IsOnlyStateProvinceFlag | binary |  | 0 = StateProvinceCode exists. 1 = StateProvinceCode unavailable |
|  | Name | varchar(50) |  | State or province description. |
|  | TerritoryID | int |  | ID of the territory in which the state or province is located. Foreign key to SalesTerritory.SalesTerritoryID. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Territory | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | TerritoryID | int |  | Primary key for Territory records. Identity / Auto increment column |
|  | Name | varchar(255) |  | Name of the Territory |
|  | CountryRegionCode | varchar(3) |  | ISO standard country or region code. Foreign key to CountryRegion.CountryRegionCode. |
|  | ModifiedDate | datetime |  | Date and time the record was last updated. Default: getdate() |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Title | | | | |
| P/F | Field Name | Data Type | Null/Unique | Description/Attributes |
| P | TitleID | int | U | Primary key for Title records. Identity / Auto increment column |
|  | Name | varchar(255) | U | Name of the Title |

# Database Populating

Database Populating:

SELECT \* FROM addresstype;

Show columns FROM addresstype;

INSERT INTO addresstype (Name)

VALUES(

'Billing'

);

INSERT INTO addresstype (Name)

VALUES(

'Home'

);

INSERT INTO addresstype (Name)

VALUES(

'Main Office'

);

INSERT INTO addresstype (Name)

VALUES(

'Primary'

);

INSERT INTO addresstype (Name)

VALUES(

'Archive'

);

SELECT \* FROM cardtype;

INSERT INTO cardtype (Name)

VALUES(

'Mastercard'

);

INSERT INTO cardtype (Name)

VALUES(

'VISA'

);

SELECT \* FROM title;

INSERT INTO title (Name)

VALUES(

'Sr.'

);

INSERT INTO title (Name)

VALUES(

'Mrs.'

);

INSERT INTO title (Name)

VALUES(

'Null'

);

INSERT INTO title (Name)

VALUES(

'Sra.'

);

INSERT INTO title (Name)

VALUES(

'Ms.'

);

INSERT INTO title (Name)

VALUES(

'Mr.'

);

SELECT \* FROM countryregion;

Show columns FROM countryregion;

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'AD',

'Andorra'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'AR',

'Argentina'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'Be',

'Belgium'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'CA',

'Canada'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'CN',

'China'

);

/\*INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'AD',

'bla'

);\*/

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'CZ',

'Czech Republic'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'DE',

'Germany'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'ES',

'Spain'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'GB',

'United Kingdom'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'PL',

'Poland'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'PT',

'Portugal'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'US',

'United States'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'FR',

'France'

);

INSERT INTO countryregion (CountryRegionCode, Name)

VALUES(

'AU',

'Australia'

);

SELECT \* FROM territory;

Show columns FROM territory;

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'NorthWest',

'US',

'North America'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'NorthEast',

'US',

'North America'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'Central',

'US',

'North America'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'SouthWest',

'US',

'North America'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'SouthEast',

'US',

'North America'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'Canada',

'CA',

'North America'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'France',

'FR',

'Europe'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'Germany',

'DE',

'Europe'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'Australia',

'AU',

'Pacific'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'United Kingdom',

'GB', /\*GR\*/

'Europe'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'Spain',

'ES',

'Europe'

);

INSERT INTO territory (Name, CountryRegionCode, Groupname)

VALUES(

'Poland',

'PL', /\*GR\*/

'Europe'

);

SELECT \* FROM TERRITORY;

SELECT \* FROM stateprovince;

Show columns FROM stateprovince;

INSERT INTO stateprovince (StateProvinceCode,CountryRegionCode,IsOnlyStateProvinceFlag, Name,TerritoryID)

VALUES(

'ENG',

'GB',

1,

'England',

7 /\*update to 13\*/

);

INSERT INTO stateprovince (StateProvinceCode,CountryRegionCode,IsOnlyStateProvinceFlag, Name,TerritoryID)

VALUES(

'AB',

'CA',

0,

'Alberta',

6

);

SELECT \* FROM TERRITORY;

INSERT INTO stateprovince (StateProvinceCode,CountryRegionCode,IsOnlyStateProvinceFlag, Name,TerritoryID)

VALUES(

'HU',

'ES',

0,

'Huelva',

15

);

INSERT INTO stateprovince (StateProvinceCode,CountryRegionCode,IsOnlyStateProvinceFlag, Name,TerritoryID)

VALUES(

'MAD',

'ES',

0,

'Madrid',

15

);

INSERT INTO stateprovince (StateProvinceCode,CountryRegionCode,IsOnlyStateProvinceFlag, Name,TerritoryID)

VALUES(

'BA',

'ES',

0,

'Barcelona',

15

);

INSERT INTO stateprovince (StateProvinceCode,CountryRegionCode,IsOnlyStateProvinceFlag, Name,TerritoryID)

VALUES(

'WRO',

'PL',

0,

'Wroclaw',

16

);

INSERT INTO stateprovince (StateProvinceCode,CountryRegionCode,IsOnlyStateProvinceFlag, Name,TerritoryID)

VALUES(

'WAR',

'PL',

0,

'Warsaw',

16

);

INSERT INTO stateprovince (StateProvinceCode,CountryRegionCode,IsOnlyStateProvinceFlag, Name,TerritoryID)

VALUES(

'KRA',

'PL',

0,

'Krakov',

16

);

SELECT \* FROM stateprovince;

Show columns FROM address;

INSERT INTO address (AddressTypeID, AddressLine1, AddressLine2,City,StateProvinceID,PostalCode)

VALUES(

3,

'Avenida Julio Caro Baroja',

'nº12 4ºA',

'Huelva',

2,

21002

);

INSERT INTO address (AddressTypeID, AddressLine1, AddressLine2,City,StateProvinceID,PostalCode)

VALUES(

3,

'Boleslawa Prusa',

'nº9 410B',

'Wroclaw',

5,

50319

);

SELECT \* FROM contact;

Show columns FROM contact;

INSERT INTO contact (TitleID, FirstName, LastName,EmailAdress,Phone,Password,AddressID)

VALUES(

4,

'Alba',

'Marquez Rodriguez',

'albamrqz751@gmail.com',

601195832,

'123456789',

1

);

SELECT \* FROM creditcard;

Show columns FROM creditcard;

INSERT INTO creditcard (CardTypeID, CardNumber, ExpMonth,ExpYear)

VALUES(

1,

1234567898765432,

12,

2025

);

INSERT INTO creditcard (CardTypeID, CardNumber, ExpMonth,ExpYear)

VALUES(

1,

0000000012348765,/\*1234567898765432,\*/

4,

2022

);

INSERT INTO creditcard (CardTypeID, CardNumber, ExpMonth,ExpYear)

VALUES(

2,

0120000012348765,/\*1234567898765432,\*/

7,

2028

);

SELECT \* FROM contactcreditcard;

Show columns FROM contactcreditcard;

INSERT INTO contactcreditcard (ContactID, CreditCardID)

VALUES(

1,

1

);

/\*INSERT INTO contactcreditcard (ContactID, CreditCardID)

VALUES(

1,

2

);\*/

SELECT \* FROM currency;

Show columns FROM currency;

INSERT INTO currency (CurrencyCode, Name)

VALUES(

'EUR',

'EURO'

);

INSERT INTO currency (CurrencyCode, Name)

VALUES(

'PLN',

'Zloty'

);

INSERT INTO currency (CurrencyCode, Name)

VALUES(

'GBP',

'United Kingdom Pound'

);

INSERT INTO currency (CurrencyCode, Name)

VALUES(

'USD',

'US Dollar'

);

INSERT INTO currency (CurrencyCode, Name)

VALUES(

'AUD',

'Australian Dollar'

);

SELECT \* FROM countryregioncurrency;

Show columns FROM countryregioncurrency;

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'AU',

'AUD'

);

select \* from countryregion;

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'AD',

'EUR'

);

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'ES',

'EUR'

);

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'BE',

'EUR'

);

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'DE',

'EUR'

);

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'FR',

'EUR'

);

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'PL',

'PLN'

);

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'PT',

'EUR'

);

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'US',

'USD'

);

INSERT INTO countryregioncurrency (CountryRegionCode,CurrencyCode)

VALUES(

'GB',

'GBP'

);

SELECT \* FROM territory;

Show columns FROM customer;

INSERT INTO customer (TerritoryID,AccountNumber,ContactID)

VALUES(

15,

1,

1

);

SELECT \* FROM unitmeasure;

Show columns FROM unitmeasure;

INSERT INTO unitmeasure (UnitMeasureCode,Name)

VALUES(

'GB',

'GigaByte'

);

INSERT INTO unitmeasure (UnitMeasureCode,Name)

VALUES(

'MB',

'MegaByte'

);

SELECT \* FROM genre;

Show columns FROM genre;

INSERT INTO genre (Name)

VALUES(

'Action'

);

INSERT INTO genre (Name)

VALUES(

'Fantasy'

);

INSERT INTO genre (Name)

VALUES(

'Thriller'

);

INSERT INTO genre (Name)

VALUES(

'Casual'

);

SELECT \* FROM saga;

Show columns FROM saga;

INSERT INTO saga (Name,Quantity)

VALUES(

'Just Cause',

4

);

SELECT \* FROM game;

Show columns FROM game;

INSERT INTO game (Name,GameNumber,StandardCost,Storage,StorageUnitMeasureCode,GenreID,PlatformID,Description,SagaID,SellStartDate)

VALUES(

'Just Cause 4 Reloaded',

1,

32.99,

47,

'GB',

1,

'PC',

'Rebel agent Rico Rodríguez arrives in Solís, a vast world located in South America mired in conflict, oppression and extreme weather conditions. Buckle up your wingsuit, equip the customizable hook, and prepare to unleash the tempest! ',

1,

current\_timestamp()

);

INSERT INTO game (Name,GameNumber,StandardCost,Storage,StorageUnitMeasureCode,GenreID,PlatformID,Description,SellStartDate)

VALUES(

'Minecraft',

2,

23.95,

1,

'GB',

1,

'PC',

'Prepare for an adventure of limitless possibilities as you build, mine, battle creatures and explore the ever-changing landscape of Minecraft ',

current\_timestamp()

);

INSERT INTO game (Name,GameNumber,StandardCost,GenreID,PlatformID,Description,SellStartDate)

VALUES(

'Among Us',

3,

3.99,

3,

'PC',

'A game for 4 to 15 players online or in local mode via Wi-Fi in which you have to prepare your spaceship for take off. But beware: one or more randomly chosen players from the crew are imposters ready to kill the rest! ',

current\_timestamp()

);

SELECT \* FROM game;

Show columns FROM gamereview;

INSERT INTO gamereview (GameID,ReviewerName,EmailAddress,Rating)

VALUES(

4,

'Alba',

'albamrqz751@gmail.com',

8

);

INSERT INTO gamereview (GameID,ReviewerName,EmailAddress,Rating,Comments)

VALUES(

4,

'Mary',

'mary@gmail.com',

6,

'Good Game'

);

/\*SELECT \* FROM gamelistpricehistory;

Show columns FROM gamelistpricehistory;

INSERT INTO gamelistpricehistory (GameID,StartDate,ListPrice)

VALUES(

4,

current\_timestamp(),

32.99

);\*/

SELECT \* FROM shoppingcartitem;

Show columns FROM shoppingcartitem;

INSERT INTO shoppingcartitem (ShoppingCartID,GameID)

VALUES(

1,

2

);

INSERT INTO shoppingcartitem (ShoppingCartID,GameID)

VALUES(

1,

4

);

INSERT INTO shoppingcartitem (ShoppingCartID,GameID)

VALUES(

2,

2

);

INSERT INTO shoppingcartitem (ShoppingCartID,GameID)

VALUES(

3,

3

);

SELECT \* FROM game\_shoppingcartitem;

Show columns FROM game\_shoppingcartitem;

INSERT INTO game\_shoppingcartitem (GameID,shoppingcartitemID)

VALUES(

2,

1

);

INSERT INTO game\_shoppingcartitem (GameID,shoppingcartitemID)

VALUES(

3,

2

);

SELECT \* FROM gamephoto;

Show columns FROM gamephoto;

INSERT INTO gamephoto (Icon,Header)

VALUES(

'url1',

'url2'

);

INSERT INTO gamephoto (Icon,Header)

VALUES(

'url1',

'url2'

);

INSERT INTO gamephoto (Header,Preview)

VALUES(

'url1',

'url2'

);

INSERT INTO gamephoto (Icon,Header,Preview)

VALUES(

'url1',

'url2',

'url3'

);

SELECT \* FROM gamevideo;

Show columns FROM gamevideo;

INSERT INTO gamevideo (ReviewVideo,PreviewVideo)

VALUES(

'url1',

'url2'

);

INSERT INTO gamevideo (PreviewVideo)

VALUES(

'url1'

);

INSERT INTO gamevideo (PreviewVideo)

VALUES(

'url1'

);

INSERT INTO gamevideo (reviewVideo)

VALUES(

'url1'

);

SELECT \* FROM gameaudiovisuals;

Show columns FROM gameaudiovisuals;

INSERT INTO gameaudiovisuals (GameID,GamePhotoID)

VALUES(

2,

1

);

INSERT INTO gameaudiovisuals (GameID,GamePhotoID,GameVideoID)

VALUES(

1,

2,

1

);

SELECT \* FROM gamelistpricehistory;

Show columns FROM gamelistpricehistory;

INSERT INTO gamelistpricehistory (GameID,StartDate,ListPrice)

VALUES(

2,

current\_timestamp(),

32.99

);

SELECT \* FROM specialoffergame;

Show columns FROM specialoffergame;

INSERT INTO specialoffergame (SpecialOfferID,GameID)

VALUES(

1,

2

);

INSERT INTO specialoffergame (SpecialOfferID,GameID)

VALUES(

2,

3

);

INSERT INTO specialoffergame (SpecialOfferID,GameID)

VALUES(

3,

4

);

SELECT \* FROM specialoffer;

Show columns FROM specialoffer;

INSERT INTO specialoffer (SpecialOfferID,Description,Discount,Type,StartDate,EndDate,SpecialOfferProductGameID)

VALUES(

1,

'No Discount',

0,

'No discount',

current\_timestamp(),

'2030-11-23 00:00:00',

1

);

INSERT INTO specialoffer (SpecialOfferID,Description,Discount,Type,StartDate,EndDate,SpecialOfferProductGameID)

VALUES(

2,

'No Discount',

0,

'No discount',

current\_timestamp(),

'2030-11-23 00:00:00',

2

);

INSERT INTO specialoffer (SpecialOfferID,Description,Discount,Type,StartDate,EndDate,SpecialOfferProductGameID)

VALUES(

3,

'No Discount',

0,

'No discount',

current\_timestamp(),

'2030-11-23 00:00:00',

3

);

SELECT \* FROM salesorderdetail;

Show columns FROM salesorderdetail;

INSERT INTO salesorderdetail (salesorderdetailID,OrderQnty,GameID,UnitPrice,SpecialOfferID,LineTotal)

VALUES(

1,

1,

2,

32.99,

0,

32.99

);

SELECT \* FROM salesorderheader;

Show columns FROM salesorderheader;

INSERT INTO salesorderheader (SalesOrderID,RevisionNumber,DueDate,Status,CustomerID,ContactID,BillToAddressID,SubTotal,TaxAmt,ContactCreditCardCreditCardID)

VALUES(

3,

1,

'2021-12-25 00:00:00',

4,

1,

1,

1,

32.99,

0,

1

);

SELECT \* FROM salesterritory;

Show columns FROM salesterritory;

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'NorthWest',

'US',

'North America',

0,

0,

0,

0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'NorthEast',

'US',

'North America',

0,0,0,0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'Central',

'US',

'North America',

0,0,0,0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)VALUES(

'SouthWest',

'US',

'North America',

0,0,0,0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'SouthEast',

'US',

'North America',

0,0,0,0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'Canada',

'CA',

'North America',

0,0,0,0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'France',

'FR',

'Europe',

0,0,0,0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'Germany',

'DE',

'Europe',

0,0,0,0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'Australia',

'AU',

'Pacific',

0,0,0,0

);

INSERT INTO salesterritory (Name, CountryRegionCode, `Group`,SalesYTD,SalesLastYear,CostYTD,CostLastYear)

VALUES(

'United Kingdom',

'GB',

'Europe',

0,0,0,0

);

# CRUD Operations

Most of the CRUD operations are on the files attached with the submission pdf. In this document there will be just some examples.

## Create

CREATE TABLE Game (

GameID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

Name varchar(50) NOT NULL UNIQUE,

GameNumber varchar(25) NOT NULL UNIQUE,

StandardCost DECIMAL(13, 4) NOT NULL DEFAULT 0.00,

ListPrice Decimal(13,4),

Storage varchar(5),

StorageUnitMeasureCode char(3),

GenreID int,

PlatformID char(5),

Description varchar(255),

SagaID int,

GameReviewID int,

SellStartDate datetime NOT NULL,

SellEndDate datetime,

LaunchDate datetime,

ModifiedDate datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE UnitMeasure (

UnitMeasureCode char(3) NOT NULL PRIMARY KEY UNIQUE ,

Name varchar(50) NOT NULL UNIQUE,

ModifiedDate datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE Genre (

GenreID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

Name varchar(50) NOT NULL UNIQUE,

Description varchar(255),

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE GameAudiovisuals (

GameID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

GamePhotoID int UNIQUE,

GameVideoID int UNIQUE,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE GamePhoto (

GamePhotoID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

Icon varchar(50),

Wallpaper varchar(50),

Header varchar(50),

Preview varchar(50),

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE GameVideo (

GameVideoID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

ReviewVideo varchar(50),

PreviewVideo varchar(50),

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE GameReview (

GameReviewID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

GameID int NOT NULL,

ReviewerName varchar(50) NOT NULL,

ReviewDate datetime NOT NULL default current\_timestamp,

EmailAddress varchar(50) NOT NULL,

Rating int NOT NULL,

Comments varchar(3000),

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE UnitMeasure (

UnitMeasureCode char(3) NOT NULL UNIQUE PRIMARY KEY,

Name varchar(50) NOT NULL UNIQUE,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE GameListPriceHistory (

GameID int NOT NULL,

StartDate datetime NOT NULL,

EndDate datetime,

ListPrice decimal(13,4) NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp,

Primary key(GameID,StartDate)

);

CREATE TABLE Game\_ShoppingCartItem (

GameID int NOT NULL,

ShoppingCartItemID int NOT NULL,

PRIMARY KEY (GameID, ShoppingCartItemID));

CREATE TABLE Saga (

SagaID int NOT NULL UNIQUE PRIMARY KEY AUTO\_INCREMENT,

Name varchar(50) NOT NULL,

Quantity int NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

/\*Sales\*/

CREATE TABLE CountryRegionCurrency(

CountryRegionCode varchar(3) not null,

CurrencyCode char(3) not null,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp,

PRIMARY KEY (CountryRegionCode, CurrencyCode)

);

CREATE TABLE CreditCard (

CreditCardID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

CardTypeID int NOT NULL,

CardNumber varchar(25) NOT NULL UNIQUE,

ExpMonth TINYINT NOT NULL,

ExpYear smallint NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE CardType (

CardTypeID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

Name varchar(25) NOT NULL UNIQUE,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE Currency (

CurrencyCode char(3) PRIMARY KEY NOT NULL,

Name varchar(25) NOT NULL UNIQUE,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE CurrencyRate (

CurrencyRateID int primary KEY NOT NULL UNIQUE AUTO\_INCREMENT,

CurrencyRateDate datetime NOT NULL,

FromCurrencyCode char(3) NOT NULL,

ToCurrencyCode char(3) NOT NULL,

AverageRate Decimal(13,4) NOT NULL,

EndOfDayRate Decimal(13,4) NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp,

constraint UC\_CurrencyRate Unique (CurrencyRateDate,FromCurrencyCode,ToCurrencyCode)

);

CREATE TABLE Customer (

CustomerID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

TerritoryID int,

AccountNumber varchar(10) NOT NULL UNIQUE,

ContactID int NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE ContactCreditCard (

ContactID int NOT NULL,

CreditCardID int NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp,

PRIMARY KEY (ContactID, CreditCardID),

constraint UC\_ContactCreditCard Unique (ContactID,CreditCardID)

);

CREATE TABLE SalesOrderDetail (

SalesOrderID int NOT NULL auto\_increment,

SalesOrderDetailID int NOT NULL,

TrackingNumber int,

OrderQnty int,

GameID int NOT NULL,

SpecialOfferID int NOT NULL,

UnitPrice int,

UnitPriceDiscount int,

LineTotal int,

ModifiedDate DATETIME,

PRIMARY KEY (SalesOrderID, SalesOrderDetailID)

);

CREATE TABLE SalesOrderHeader (

SalesOrderID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

RevisionNumber tinyint NOT NULL DEFAULT 0,

OrderDate datetime NOT NULL DEFAULT CURRENT\_TIMESTAMP,

DueDate datetime NOT NULL,

Status tinyint NOT NULL DEFAULT 1,

SalesOrderNumber varchar(25) UNIQUE,

AccountNumber varchar(15),

CustomerID int NOT NULL,

ContactID int NOT NULL,

TerritoryID int,

BillToAddressID int NOT NULL,

CreditCartdApprovalCode varchar(15),

SubTotal DECIMAL(13, 4) NOT NULL DEFAULT 0.00, /\*DECIMAL(13, 4) = money\*/

TaxAmt DECIMAL(13, 4) NOT NULL DEFAULT 0.00,

Comment varchar(128),

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp,

CreditCardID int,

CurrencyRateID int,

ContactCreditCardCreditCardID int NOT NULL);

/\*PRIMARY KEY (SalesOrderID), UNIQUE INDEX (SalesOrderID)\*/

CREATE TABLE SalesTerritory (

TerritoryID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

Name varchar(50) NOT NULL UNIQUE,

CountryRegionCode varchar(3) NOT NULL,

`Group` varchar(50) NOT NULL,

SalesYTD DECIMAL(13, 4) NOT NULL default 0,

SalesLastYear DECIMAL(13, 4) NOT NULL default 0,

CostYTD DECIMAL(13, 4) NOT NULL default 0,

CostLastYear DECIMAL(13, 4) NOT NULL default 0,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE ShoppingCartItem (

ShoppingCartItemID int PRIMARY KEY NOT NULL UNIQUE AUTO\_INCREMENT,

ShoppingCartID varchar(50) NOT NULL,

GameID int NOT NULL,

DateCreated datetime NOT NULL DEFAULT current\_timestamp,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE SpecialOffer (

SpecialOfferID int PRIMARY KEY NOT NULL UNIQUE,

Description varchar(255) NOT NULL,

Discount DECIMAL(13, 4) NOT NULL default 0,

Type varchar(50) NOT NULL,

StartDate date NOT NULL,

EndDate date NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp,

SpecialOfferProductGameID int NOT NULL

);

/\*Person\*/

CREATE TABLE SpecialOfferGame (

SpecialOfferID int NOT NULL UNIQUE,

GameID int NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp,

PRIMARY KEY (SpecialOfferID, GameID)

);

CREATE TABLE Address (

AddressID int NOT NULL unique PRIMARY KEY AUTO\_INCREMENT,

AddressTypeID int NOT NULL,

AddressLine1 varchar(60),

AddressLine2 varchar(60),

City varchar(30) NOT NULL,

StateProvinceID int NOT NULL,

PostalCode int NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE AddressType (

AddressTypeID int NOT NULL unique PRIMARY KEY AUTO\_INCREMENT,

Name varchar(255) NOT NULL UNIQUE

);

CREATE TABLE Contact (

ContactID int NOT NULL unique PRIMARY KEY AUTO\_INCREMENT,

TitleID int NOT NULL,

FirstName varchar(255) NOT NULL,

MiddleName varchar(255),

LastName varchar(255) NOT NULL,

EmailAdress varchar(255) NOT NULL UNIQUE,

Phone int NOT NULL,

Password int NOT NULL,

AdditionalContactInfo varchar(255),

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp,

AddressID int NOT NULL

);

CREATE TABLE CountryRegion (

CountryRegionCode varchar(3) PRIMARY KEY UNIQUE NOT NULL,

Name varchar(255) NOT NULL Unique,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE StateProvince (

StateProvinceID inT NOT NULL unique PRIMARY KEY AUTO\_INCREMENT,

CountryRegionCode varchar(3) NOT NULL,

IsOnlyStateProvinceFlag binary(1) NOT NULL DEFAULT 0,

Name varchar(255) NOT NULL,

TerritoryID int NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE Territory (

TerritoryID int NOT NULL unique PRIMARY KEY AUTO\_INCREMENT,

Name varchar(255) NOT NULL UNIQUE,

CountryRegionCode varchar(3) NOT NULL,

ModifiedDate datetime NOT NULL DEFAULT current\_timestamp

);

CREATE TABLE Title (

TitleID int NOT NULL unique PRIMARY KEY AUTO\_INCREMENT,

Name varchar(255) NOT NULL UNIQUE

);

## Read

SELECT \* FROM address;

SELECT \* FROM addresstype;

SELECT \* FROM cardtype;

SELECT \* FROM contact;

SELECT \* FROM contactcreditcard;

SELECT \* FROM countryregion;

SELECT \* FROM countryregioncurrency;

SELECT \* FROM creditcard;

SELECT \* FROM currency;

SELECT \* FROM customer;

SELECT \* FROM game;

SELECT \* FROM game\_shoppingcartitem;

SELECT \* FROM gameaudiovisuals;

SELECT \* FROM gamelistpricehistory;

SELECT \* FROM gamephoto;

SELECT \* FROM gamereview;

SELECT \* FROM gamevideo;

SELECT \* FROM genre;

SELECT \* FROM saga;

SELECT \* FROM salesorderdetail;

SELECT \* FROM salesorderheader;

SELECT \* FROM salesterritory;

SELECT \* FROM shoppingcartitem;

SELECT \* FROM specialoff;

SELECT \* FROM specialoffergame;

SELECT \* FROM stateprovince;

SELECT \* FROM territory;

SELECT \* FROM title;

select \* from address

where City='Wroclaw'

select \* from address

where City='Wroclaw';

select \* from territory

where CountryRegionCode='US';

select city from address as ad

join stateprovince as st on ad.StateProvinceID=st.StateProvinceID

where st.CountryRegionCode='PL';

## Update

To set relations between databases:

ALTER TABLE Game\_ShoppingCartItem ADD CONSTRAINT FKGame\_ShoppingCartItem\_Game FOREIGN KEY (GameID) REFERENCES Game (GameID);

ALTER TABLE Game ADD CONSTRAINT FKGame\_Genre FOREIGN KEY (GenreID) REFERENCES Genre (GenreID);

ALTER TABLE Game ADD CONSTRAINT FKGame\_Saga FOREIGN KEY (SagaID) REFERENCES Saga (SagaID);

ALTER TABLE Game\_ShoppingCartItem ADD CONSTRAINT FKGame\_ShoppingCartItem FOREIGN KEY (ShoppingCartItemID) REFERENCES ShoppingCartItem (ShoppingCartItemID);

ALTER TABLE CreditCard ADD CONSTRAINT FKCreditCard\_CardType FOREIGN KEY (CardTypeID) REFERENCES CardType (CardTypeID);

ALTER TABLE CurrencyRate ADD CONSTRAINT FKCurrencyRate\_CurrencyFROM FOREIGN KEY (FromCurrencyCode) REFERENCES Currency (CurrencyCode);

ALTER TABLE CurrencyRate ADD CONSTRAINT FKCurrencyRate\_CurrencyTO FOREIGN KEY (ToCurrencyCode) REFERENCES Currency (CurrencyCode);

ALTER TABLE ContactCreditCard ADD CONSTRAINT FKContactCreditCard\_CreditCard FOREIGN KEY (CreditCardID) REFERENCES CreditCard (CreditCardID);

ALTER TABLE Customer ADD CONSTRAINT FKCustomer\_Territory FOREIGN KEY (TerritoryID) REFERENCES Territory (TerritoryID);

ALTER TABLE Territory ADD CONSTRAINT FKTerritory\_CountryRegion FOREIGN KEY (CountryRegionCode) REFERENCES CountryRegion (CountryRegionCode);

ALTER TABLE Customer ADD CONSTRAINT FKCustomer\_Contact FOREIGN KEY (ContactID) REFERENCES Contact (ContactID);

ALTER TABLE Contact ADD CONSTRAINT FKContact\_Title FOREIGN KEY (TitleID) REFERENCES Title (TitleID);

ALTER TABLE Contact ADD CONSTRAINT FKContact\_Address FOREIGN KEY (AddressID) REFERENCES Address (AddressID);

ALTER TABLE Address ADD CONSTRAINT FKAddress\_AddressType FOREIGN KEY (AddressTypeID) REFERENCES AddressType (AddressTypeID);

ALTER TABLE Address ADD CONSTRAINT FKAddress\_StateProvince FOREIGN KEY (StateProvinceID) REFERENCES StateProvince (StateProvinceID);

ALTER TABLE StateProvince ADD CONSTRAINT FKStateProvince\_CountryRegion FOREIGN KEY (CountryRegionCode) REFERENCES CountryRegion (CountryRegionCode);

ALTER TABLE SalesOrderHeader ADD CONSTRAINT FKSalesOrderHeader\_SalesOrderDetail FOREIGN KEY (SalesOrderID) REFERENCES SalesOrderDetail (SalesOrderID);

ALTER TABLE SalesOrderHeader ADD CONSTRAINT FKSalesOrderHeader\_Contact FOREIGN KEY (ContactID) REFERENCES Contact (ContactID);

ALTER TABLE SalesOrderHeader ADD CONSTRAINT FKSalesOrderHeader\_ContactCreditCard2 FOREIGN KEY (ContactID, CreditCardID) REFERENCES ContactCreditCard (ContactID, CreditCardID);

ALTER TABLE SalesOrderHeader ADD CONSTRAINT FKSalesOrderHeader\_Territory FOREIGN KEY (TerritoryID) REFERENCES Territory (TerritoryID);

ALTER TABLE SalesOrderHeader ADD CONSTRAINT FKSalesOrderHeader\_SalesTerritory FOREIGN KEY (TerritoryID) REFERENCES SalesTerritory (TerritoryID);

ALTER TABLE SalesOrderHeader ADD CONSTRAINT FKSalesOrderHeader\_CurrencyRate FOREIGN KEY (CurrencyRateID) REFERENCES CurrencyRate (CurrencyRateID);

ALTER TABLE GameListPriceHistory ADD CONSTRAINT FKGameListPriceHistory\_Game FOREIGN KEY (GameID) REFERENCES Game (GameID);

ALTER TABLE GameReview ADD CONSTRAINT FKGameReview\_Game FOREIGN KEY (GameID) REFERENCES Game (GameID);

/\*ALTER TABLE Game ADD CONSTRAINT FKGame\_GameAudiovisuals FOREIGN KEY (GameID) REFERENCES GameAudiovisuals (GameID);\*/

/\*ALTER TABLE Game DROP FOREIGN KEY FKGame\_GameAudiovisuals;\*/

ALTER TABLE GameAudiovisuals ADD CONSTRAINT FKGameAudiovisuals\_GamePhoto FOREIGN KEY (GamePhotoID) REFERENCES GamePhoto (GamePhotoID);

ALTER TABLE GameAudiovisuals ADD CONSTRAINT FKGameAudiovisuals\_GameVideo FOREIGN KEY (GameVideoID) REFERENCES GameVideo (GameVideoID);

ALTER TABLE SalesOrderHeader ADD CONSTRAINT FKSalesOrderHeader\_Customer FOREIGN KEY (CustomerID) REFERENCES Customer (CustomerID);

ALTER TABLE SpecialOfferGame ADD CONSTRAINT FKSpecialOfferGame\_Game FOREIGN KEY (GameID) REFERENCES Game (GameID);

/\*ALTER TABLE SpecialOfferGame ADD CONSTRAINT FKSpecialOfferGame\_SpecialOffer FOREIGN KEY (SpecialOfferID) REFERENCES SpecialOffer (SpecialOfferID);\*/

/\*ALTER TABLE SpecialOfferGame DROP FOREIGN KEY FKSpecialOfferGame\_SpecialOffer;\*/

/\*ALTER TABLE SpecialOffer ADD CONSTRAINT FKSpecialOffer FOREIGN KEY (SpecialOfferID, SpecialOfferProductGameID) REFERENCES SpecialOfferGame (SpecialOfferID, GameID);

ALTER TABLE SpecialOffer DROP FOREIGN KEY FKSpecialOffer;\*/

/\*ALTER TABLE SalesOrderDetail ADD CONSTRAINT FKSalesOrderDetail\_SpecialOfferGame FOREIGN KEY (SpecialOfferID, GameID) REFERENCES SpecialOfferGame (SpecialOfferID, GameID);

ALTER TABLE SalesOrderDetail DROP FOREIGN KEY FKSalesOrderDetail\_SpecialOfferGame;\*/

ALTER TABLE ContactCreditCard ADD CONSTRAINT FKContactCreditCard FOREIGN KEY (ContactID) REFERENCES Contact (ContactID);

ALTER TABLE countryregioncurrency ADD CONSTRAINT FKcountryregioncurrency FOREIGN KEY (CountryRegionCode) REFERENCES CountryRegion (CountryRegionCode);

ALTER TABLE gamelistpricehistory ADD CONSTRAINT FKGameListPriceHistory FOREIGN KEY (GameID) REFERENCES Game (GameID);

/\*ALTER TABLE unitmeasure ADD CONSTRAINT FKunitmeasure FOREIGN KEY (UnitMeasureCode) REFERENCES Game (StorageUnitMeasureCode);\*/

Other updates on the tables structure:

ALTER TABLE territory

ADD COLUMN GroupName VARCHAR(50) AFTER CountryRegionCode;

ALTER TABLE STATEPROVINCE

ADD COLUMN StateProvinceCode char(3) AFTER StateProvinceID;

Updates referring to tables data:

UPDATE stateprovince

SET TerritoryID=13

WHERE Name='England';

## DELETE

INSERT INTO address (AddressTypeID, AddressLine1,City,StateProvinceID,PostalCode)

VALUES(

3,

'test street',

'Madrid',

2,

21000

);

SELECT \* FROM ADDRESS;

DELETE FROM address

WhERE City='Madrid';

SELECT \* FROM ADDRESS

# Additional Indexes

# List of Applications views

## Main Login Screen

Data:

* User login information (email and password from Contact table)
* User registration information (email, password, first name, last name… -> Contact table)

## Register

Data:

* User personal information -> Contact table, Address Table

## Main Screen

* Game information -> Game
* Game offers information -> Game, GameSpecialOffer, GameOffer
* User personal information -> Contact table

## Show Personal User information

Data:

* User personal information -> Contact table

## Filtered Offers

Data:

* Game information -> Game
* Game offers information -> Game, GameSpecialOffer, GameOffer

## Show Game Information

Data:

* Game information -> Game
* Game AudioVisuals -> GameAudivisuals, GamePhoto, GameVideo
* Other Game’s Saga -> Saga
* Game Review’s -> GameReview

## Add Review

Data:

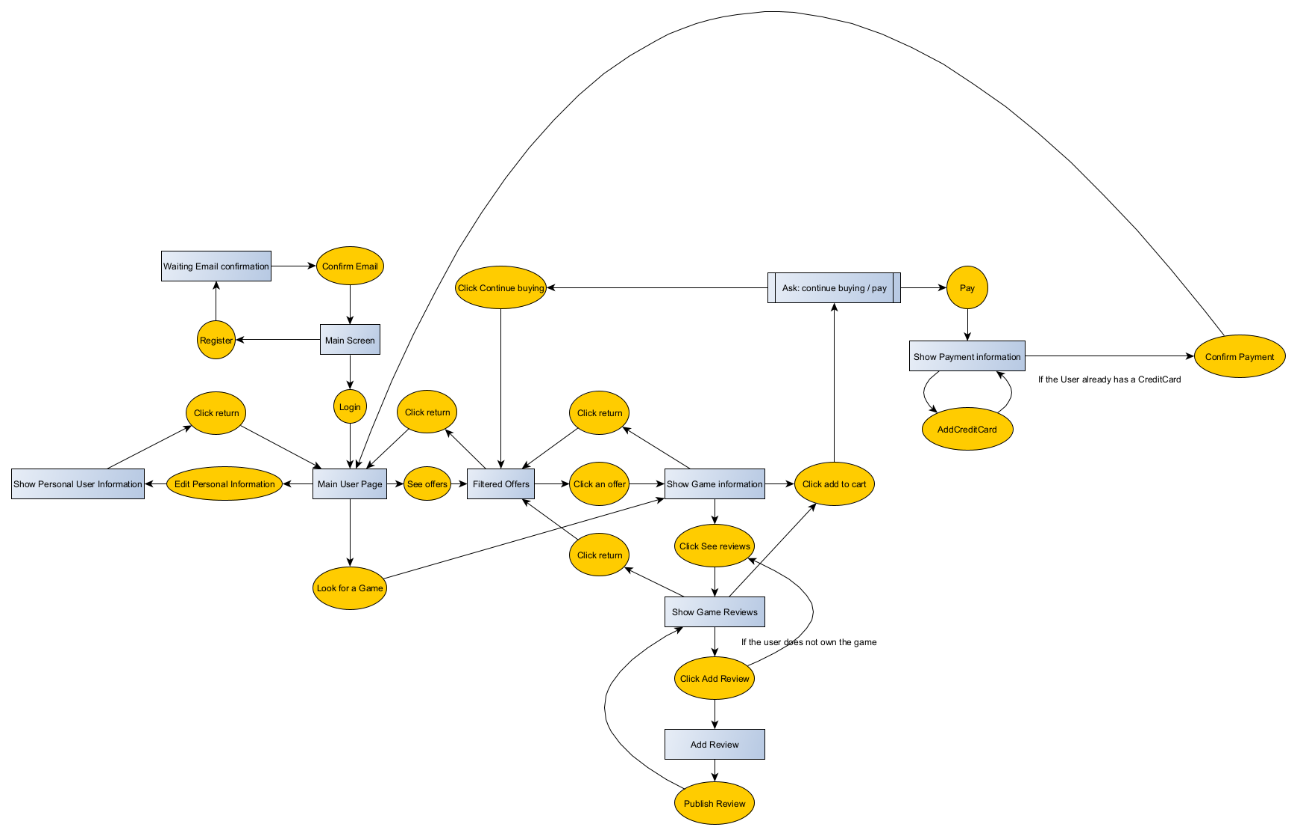
* Game Review’s -> GameReview

## Show payment information

Data:

* Personal User Information -> Contact
* Order -> SalesOrderHeader, SalesOrderDetail
* CreditCard -> ContactCreditCard, CreditCard, CardType

# Interface State Diagram



# Mock-up for each interface view

## Main Login Screen

Graphical user interface, application

Description automatically generated

## Register

Graphical user interface

Description automatically generated

## Main Screen

Graphical user interface, text, application

Description automatically generated

## Personal Information

Graphical user interface

Description automatically generated

## Filtered Offers

Graphical user interface, text, application

Description automatically generated

## Show game information

## Graphical user interface, text, application Description automatically generated

## Reviews

## Graphical user interface, text, application Description automatically generated

## Add Review

## Graphical user interface, text, application Description automatically generated

## Continue Buying/ Pay

## Graphical user interface, application Description automatically generated

## Show Payment Information

## Graphical user interface, text Description automatically generated

## Add Payment Method

## Graphical user interface, application Description automatically generated

# Bibliography