-

inventory@localhost\SQLEXPRESS

Data Dictionary

8/19/2021

Table of contents

Legend

 Primary key

 Primary key disabled

 User-defined primary key

 Unique key

 Unique key disabled

 User-defined unique key

 Active trigger

 Disabled trigger

 Many to one relation

 User-defined many to one relation

 One to many relation

 User-defined one to many relation

 Many to many relation

 User-defined many to many relation

 One to one relation

 User-defined one to one relation

 Input

 Output

 Input/Output

 Uses dependency

 User-defined uses dependency

 Used by dependency

 User-defined used by dependency

inventory@localhost\SQLEXPRESS

Data base for invebtory operations from selling ,acquisition and orders

1. Tables

1.1. Table: cash_order (cash_order)

Information related to the payment of a product

Columns

		Name	Data type	Description / Attributes
		cash_order_id	int	unique number to differntiate every order,automatic increase by 1 **Identity / Auto increment**
		invoice_number	int	number of the payment method
		brand_name	nvarchar(50)	brand logo or name **Nullable**
		model	nvarchar(50)	model of product **Nullable**
		serial_number	int	unique number of the product to differntiate
		price	int	
		unit_type	nvarchar(55)	**Nullable**

Unique keys

Columns		Name / Description
	cash_order_id	PK__cash_ord__6A55A155FEB7EF02 unique numer for differentiating of orders

1.2. Table: charge_order (payment)

Table that shows the price and payment methods of an order

Columns

		Name	Data type	Description / Attributes
		chargeoder_id	int	unique number to show the number of operation **Identity / Auto increment**
		invoice_number	int	numer of the payment method **Nullable**
		brand_name	nvarchar(50)	manufacuring logo or name **Nullable**
		model	nvarchar(50)	model of a product **Nullable**
		unit_Type	nvarchar(50)	**Nullable**
		serial_Number	int	unique number to differentiate between products
		price	int	price of the order
		down_Payment	int	the first payment for the order
		month_Term	int	the method for monthly payments **Nullable**
		monthly_Payment	int	the monthly payment for the order

Unique keys

Columns	Name / Description
chargeoder_id	PK__charge_o__5015DEC19D31300A unique number to show the number of operation

1.3. Table: customer (customer info)

Table that show some information about customers

Columns

		Name	Data type	Description / Attributes
		customer_id	int	primary key to identify customers **Identity / Auto increment**
		first_name	nvarchar(40)	first name of the customer
		last_name	nvarchar(40)	last name of the custome
		address	nvarchar(255)	address of the customer **Nullable**
		contact_number	nvarchar(40)	customer's way of contact

Linked from

	Table	Join	Title / Name / Description
	orders (orders)	**customer**customer_id = orderscustomer_id	FK__orders__customer__3C69FB99 linking the orders table by customer id
	product (product info)	**customer**customer_id = productcustomer_id	FK__product__custome__4222D4EF linking the product table by customer id

Unique keys

	Columns	Name / Description
	customer_id	PK__customer__CD65CB8588FF6573 unique number to identify customers

1.4. Table: Employee (Employee)

Table that shows some information about our employees

Columns

		Name	Data type	Description / Attributes
		employee_id	int	unique number for every employee **Identity / Auto increment**
		first_name	nvarchar(40)	first name of the employee
		last_name	nvarchar(40)	last name of the employee
		address	nvarchar(255)	address of the employee **Nullable**
		contact_number	nvarchar(40)	how to contact the employee
		position	nvarchar(40)	the employee's job

Linked from

	Table	Join	Title / Name / Description
	orders (orders)	**Employee**employee_id = ordersemployee_id	FK__orders__employee__3D5E1FD2 linking orders to employee using employee id

Unique keys

	Columns	Name / Description
	employee_id	PK__Employee__C52E0BA8EE30BED0 unique number to identify employees

1.5. Table: orders (orders)

Table that shows information about orders

Columns

		Name	Data type	Description / Attributes
		order_id	int	unique number to differentiate orders **Identity / Auto increment**
		customer_id	int	unique number to identify customers , foreign key for custome table **References**: customer (customer info)
		employee_id	int	unique number to identify employees, foreign key for employee table **References**: Employee (Employee)
		invoice_number	int	unique number for the payment method
		total_price	int	total price of the order'
		date_order	date	when the order was placed
		quantity	int	quantity of the order
		brand_id	int	unique number to identify products, foreign key fot product table **References**: product (product info)

Links to

	Table	Join	Title / Name / Description
	customer (customer info)	**orders**customer_id = customercustomer_id	FK__orders__customer__3C69FB99 linking the orders table by customer id
	Employee (Employee)	**orders**employee_id = Employeeemployee_id	FK__orders__employee__3D5E1FD2 linking orders to employee using employee id
	product (product info)	**orders**brand_id = productproduct_id	order_product linkking the product table using product id=brand_id

Unique keys

Columns		Name / Description
	order_id	PK__orders__465962293DB302C4 shows the number of the order

1.6. Table: product (product info)

shows information about the product

Columns

		Name	Data type	Description / Attributes
		product_id	int	unique number of the product **Identity / Auto increment**
		date_received	date	date the product was delivered **Nullable**
		brand_id	int	
		brand_name	nvarchar(55)	name or logo of brand **Nullable**
		model	nvarchar(55)	model of the product **Nullable**
		serial_number	int	unique number to differnciate products
		availability	nvarchar(55)	is there enough in stock
		date_sold	date	when was it sold **Nullable**
		customer_id	int	unique number about the customer **References**: customer (customer info)
		unit_type	nvarchar(55)	

Links to

	Table	Join	Title / Name / Description
	customer (customer info)	**product**customer_id = customercustomer_id	FK__product__custome__4222D4EF linking the product table by customer id

Linked from

	Table	Join	Title / Name / Description
	orders (orders)	**product**product_id = ordersbrand_id	order_product linkking the product table using product id=brand_id
	supplier (supplier info)	**product**product_id = supplierbrand_id	brand_product linking supplier using brand_id=product_id

Unique keys

	Columns	Name / Description
	product_id	PK__product__47027DF5086E9E1E

1.7. Table: supplier (supplier info)

show information about the supplier

Columns

		Name	Data type	Description / Attributes
		supplier_id	int	unique number to differntiate suppliers **Identity / Auto increment**
		name	nvarchar(40)	name of supplier
		address	nvarchar(255)	address of supplier **Nullable**
		contact_number	nvarchar(40)	contact of supllier
		brand_id	int	the product the supplier is responsiblie for **References**: product (product info)

Links to

	Table	Join	Title / Name / Description
	product (product info)	**supplier**brand_id = productproduct_id	brand_product linking supplier using brand_id=product_id

Unique keys

	Columns	Name / Description
	supplier_id	PK__supplier__6EE594E81F4DC0CC unique number to differentiate suppliers

2. Views

2.1. View: trial

Columns

		Name	Data type	Description / Attributes
		brand_id	int	
		brand_name	nvarchar(55)	Nullable
		invoice_number	int	
		quantity	int	

Script

```
create view trial as
select b.brand_id,b.brand_name ,o.invoice_number,o.quantity from product b
inner join orders o on o.customer_id=b.customer_id
```

3. Procedures

3.1. Procedure: cashorder_insert (insert infro to cash order table)

Input/Output

	Name	Data type	Description
→@	invoice_number	int	
→@	brand_name	nvarchar(55)	
→@	model	nvarchar(255)	
→@	serial_number	int	
→@	price	int	
→@	unit_type	nvarchar(55)	

Uses

Name
cashorder_insert (insert infro to cash order table)
cash_order (cash_order)

Script

```
create procedure cashorder_insert @invoice_number int,
@brand_name nvarchar(55),@model nvarchar(255),@serial_number int,@price int,@unit_type nvarchar(55)
AS
insert into  [dbo].[cash_order] (invoice_number,brand_name, model, serial_number, price, unit_type)
             values(@invoice_number,@brand_name, @model, @serial_number, @price, @unit_type );
```

3.2. Procedure: chargeorder_insert (insert info to charge order table)

Input/Output

	Name	Data type	Description
→@	invoice_number	int	
→@	brand_name	nvarchar(55)	
→@	model	nvarchar(255)	
→@	unit_Type	nvarchar(55)	
→@	serial_number	int	
→@	price	int	
→@	down_Payment	int	
→@	month_Term	int	
→@	monthly_Payment	int	

Uses

Name
chargeorder_insert (insert info to charge order table)
charge_order (payment)

Script

```
create procedure chargeorder_insert @invoice_number int,
@brand_name nvarchar(55),@model nvarchar(255),@unit_Type nvarchar(55),@serial_number int
,@price int,@down_Payment int,@month_Term int,@monthly_Payment int
AS
insert into  [dbo].[charge_order] (invoice_number, brand_name, model, unit_Type
, serial_number, price, down_Payment, month_Term, monthly_Payment)
              values(@invoice_number, @brand_name, @model, @unit_Type
, @serial_number, @price, @down_Payment, @month_Term, @monthly_Payment );
```

3.3. Procedure: customer_insert (insert info to customer table)

Input/Output

	Name	Data type	Description
→@	first_name	nvarchar(55)	
→@	last_name	nvarchar(55)	
→@	address	nvarchar(255)	
→@	contact_number	nvarchar(55)	

Uses

Name
customer_insert (insert info to customer table)
customer (customer info)

Script

```
create procedure customer_insert @first_name nvarchar(55),
@last_name nvarchar(55),@address nvarchar(255),@contact_number nvarchar(55)
AS
insert into  [dbo].[customer] (first_name, last_name, [address], contact_number)
             values(@first_name ,@last_name ,@address ,@contact_number );
```

3.4. Procedure: employee_insert (insert info to employee table)

Input/Output

	Name	Data type	Description
→@	first_name	nvarchar(55)	
→@	last_name	nvarchar(55)	
→@	address	nvarchar(255)	
→@	contact_number	nvarchar(55)	
→@	position	nvarchar(55)	

Uses

Name
employee_insert (insert info to employee table)
Employee (Employee)

Script

```
create procedure employee_insert @first_name nvarchar(55),
@last_name nvarchar(55),@address nvarchar(255),@contact_number nvarchar(55),@position nvarchar(55)
AS
insert into  [dbo].Employee (first_name, last_name, [address], contact_number, position)
             values(@first_name ,@last_name ,@address ,@contact_number,@position );
```

3.5. Procedure: order_insert (insert info to orders table)

Input/Output

	Name	Data type	Description
→@	customer_id	int	
→@	employee_id	int	
→@	invoice_number	int	
→@	total_price	int	
→@	date_order	date	
→@	quantity	int	
→@	brand_id	int	

Uses

Name
order_insert (insert info to orders table)
orders (orders)

Script

```
create procedure order_insert @customer_id int,
@employee_id int,@invoice_number int,@total_price int,@date_order date,
@quantity int ,@brand_id int
AS
insert into  [dbo].[orders] (customer_id, employee_id,
invoice_number, total_price, date_order, quantity, brand_id)
              values(@customer_id ,@employee_id ,@invoice_number ,@total_price,@date_order,
              @quantity,@brand_id);
```

3.6. Procedure: product_insert (insert info to product table)

Input/Output

	Name	Data type	Description
@	date_received	date	
@	brand_id	int	
@	brand_name	nvarchar(255)	
@	model	nvarchar(55)	
@	serial_number	int	
@	availability	nvarchar(55)	
@	date_sold	date	
@	customer_id	int	
@	unit_type	nvarchar(55)	

Uses

Name
product_insert (insert info to product table)
product (product info)

Script

```
create procedure  product_insert   @date_received date,
@brand_id int,@brand_name nvarchar(255),@model nvarchar(55),  @serial_number int,@availability nvarchar(55),
@date_sold date ,@customer_id int ,@unit_type  nvarchar(55)
AS
insert into  [dbo].[product] (date_received, brand_id, brand_name,
model, serial_number, [availability], date_sold,customer_id, unit_type)
              values(@date_received ,@brand_id ,@brand_name,@model,@serial_number
              ,@availability,@date_sold,@customer_id,@unit_type );
```

3.7. Procedure: supplier_insert (insert info to supplier table)

Input/Output

	Name	Data type	Description
→@	name	nvarchar(55)	
→@	address	nvarchar(55)	
→@	contact_number	nvarchar(255)	
→@	brand_id	int	

Uses

Name
supplier_insert (insert info to supplier table)
supplier (supplier info)

Script

```
create procedure supplier_insert @name nvarchar(55),
@address nvarchar(55),@contact_number nvarchar(255),@brand_id int
AS
insert into  [dbo].[supplier] ([name],[address],contact_number,brand_id)
             values(@name ,@address ,@contact_number ,@brand_id );
```