Subastas virtuales

La casa de subasta “Compra y Vende Kenlly” quiere invertir en una página web de subastas en línea para su negocio, que les permita aumentar sus ventas y que los clientes ya no tengan la necesidad de viajar a su local para lograr vender o adquirir un producto.

Los trabajadores del local tienen fichas impresas donde registran a sus usuarios tanto como del comprador como del vendedor, entre los datos que se le pida son (nombres, apellidos, CI, teléfono, ciudad donde reside, dirección, correo, edad), dentro del formulario de quienes van a vender productos tiene que ingresar los siguientes parámetros (valor del producto, fecha cuando se ingresó, tipo de objeto, nombre del objeto), todo estos requisitos se registran en las fichas, al momento que un cliente obtiene un producto dentro de las subastes, se llenara otra ficha donde se ingresan los siguientes parámetros como (cuanto fue el monto vendido, fecha de salida del producto , valor inicial registrado previamente).

Si los productos son de gran escala, la casa de subasta tiene autos donde se les hace llegar los productos a los clientes, teniendo en cuenta que esto se solicita llenando una ficha donde se ingresan (costo de viaje, ciudad donde se dirige, fecha requerida de llegada, hora de requerida de llegada), los parámetros ingresados en las fichas ayudan a la administración del negocio.

Modelo

Diagrama

Descripción generada automáticamente

Modelo lógico

Diagrama, Esquemático

Descripción generada automáticamente

Modelo físico

Diagrama, Esquemático

Descripción generada automáticamente

Consultas

1 consulta

Mostrar los datos personales de las personas que ofertaron un producto tipo consola en las subastas en línea.

SELECT

login.res\_ci,

oferta.ofer\_valor,

oferta.ofer\_fecha,

oferta.ofer\_tipo,

registro.res\_nombre,

registro.res\_apellido

FROM

public.login

inner join public.perfil on perfil.per\_id = login.per\_id

inner join public.registro on registro.res\_ci = login.res\_ci

inner join public.oferta on perfil.per\_id = oferta.per\_id

WHERE

oferta.ofer\_tipo = 'consola';

2 consulta

¿Cuál fue el valor que se pudo vender los productos tipos ‘¿Consola’, mostrando su valor inicial y los datos personas de sus respectivos compradores?

SELECT

login.res\_ci,

registro.res\_nombre,

registro.res\_apellido,

oferta.ofer\_tipo,

compra.compr\_valor\_inical,

compra.compr\_valor\_comprado

FROM

public.login

inner join public.perfil on perfil.per\_id = login.per\_id

inner join public.oferta on perfil.per\_id = oferta.per\_id

inner join public.registro on registro.res\_ci = login.res\_ci

inner join public.compra on compra.compr\_id = oferta.compr\_id

WHERE

oferta.ofer\_tipo = 'portatil';

3 consulta

¿Cuánto fue el valor total que generaron las ventas de los productos de tipo portátil y cuadro?

SELECT

oferta.ofer\_tipo,

sum (compra.compr\_valor\_comprado) as suma\_total

FROM

public.oferta

inner join public.compra on compra.compr\_id = oferta.compr\_id

GROUP BY (oferta.ofer\_tipo)

4 consulta

¿Cuántos fueron los productos vendidos exitosamente en las subastas, muestre los datos personales de su vendedor y el producto vendido?

SELECT

registro.res\_ci,

registro.res\_nombre,

registro.res\_apellido,

registro.res\_numtelefonico,

oferta.ofer\_tipo,

oferta.ofer\_nombre,

oferta\_estado.ofer\_est\_descripcion

FROM

public.oferta

inner join public.perfil on perfil.per\_id = oferta.per\_id

inner join public.login on perfil.per\_id = login.per\_id

inner join public.registro on registro.res\_ci = login.res\_ci

inner join public.oferta\_estado on oferta\_estado.ofer\_est\_id = oferta.ofer\_est\_id

WHERE

oferta\_estado.ofer\_est\_descripcion = 'vendido'

Trigger

Crear un trigger que impida exceder el máximo de $900.00 en una subasta hacia un producto ofertado

create or replace function valormaximo()

returns trigger as $valormaximo$

declare

maximo decimal;

begin

SELECT compr\_valor\_comprado INTO maximo FROM COMPRA;

if (maximo < 900.00) then /\*\*/

raise exception 'NO se permite vender mas de $900.00 en una venta' ;

end if;

return new;/\*\*/

end;

$valormaximo$

Language plpgsql;

/\*trigger\*/

CREATE trigger valormaximo after insert

on compra for each row

execute procedure valormaximo();

insert into COMPRA values (20,1,'2022-12-20','2022-01-01',500.00,950.00);

Cursor

Curos que muestre los datos personales de la persona que se le tiene que enviar un producto, al igual que la ciudad , la cantidad y las fechas de salida y llegada.

do

$$

DECLARE

recorrido RECORD;

datos cursor for SELECT

registro.res\_nombre,

registro.res\_apellido,

registro.res\_ci,

oferta.ofer\_tipo,

envios.env\_ciudad,

envios.env\_costo\_envio,

envios.env\_fecha\_salida,

envios.env\_fecha\_llegada

FROM

public.oferta,

public.perfil,

public.login,

public.registro,

public.compra,

public.envios

WHERE

perfil.per\_id = oferta.per\_id AND

perfil.per\_id = login.per\_id AND

login.res\_ci = registro.res\_ci AND

compra.compr\_id = envios.compr\_id AND

compra.compr\_id = oferta.compr\_id;

BEGIN

OPEN datos;

FETCh datos INTO recorrido;

while (found) loop

RAISE NOTICE

'

Nombre de la persona: % Apellido de la persona: % Cedula: %

Tipo de producto: %

Ciudad donde se enviara: % Costo de envio: %

Fecha de salida: % Fecha de llegada: %

',

recorrido.res\_nombre, recorrido.res\_apellido, recorrido.res\_ci,recorrido.ofer\_tipo,

recorrido.env\_ciudad,recorrido.env\_costo\_envio,recorrido.env\_fecha\_salida,recorrido.env\_fecha\_llegada;

FETCh datos INTO recorrido;

end loop;

end $$

Language 'plpgsql';

Procedimiento almacenado

Procedimiento almacenado que devuelva la cantidad de personas que vendieron productor de acuerdo a su tipo.

create or replace function ventas\_total(varchar)

returns text

as $ventas\_total$

declare

datos alias for $1;

totalpersona int;

begin

SELECT

count(registro.res\_ci) INTO totalpersona

FROM

public.oferta

inner join public.perfil on perfil.per\_id = oferta.per\_id

inner join public.login on perfil.per\_id = login.per\_id

inner join public.registro on login.res\_ci = registro.res\_ci

WHERE

$1 = oferta.ofer\_tipo

GROUP BY

oferta.ofer\_tipo;

return totalpersona;

end;

$ventas\_total$

Language plpgsql;

SELECT ventas\_total('portatil')

Ireport

Reporte que muestre la cantidad que al final fueron vendidos los productos de acorde a su estado de compra

SELECT

oferta.ofer\_tipo,

compra\_estado.comp\_est\_descipcion,

compra.compr\_valor\_inical,

compra.compr\_valor\_comprado

FROM

public.oferta,

public.compra\_estado,

public.compra

WHERE

compra\_estado.comp\_est\_id = oferta.comp\_est\_id AND

compra.compr\_id = oferta.compr\_id;