

A6: Indexes, triggers, user functions and population

This artefact contains the physical schema of the database, the identification and characterisation of the indexes, the support of data integrity rules with triggers and the definition of the database user-defined functions. This artefact also contains the database's workload as well as the complete database creation script, including all SQL necessary to define all integrity constraints, indexes and triggers.

1. Database Workload

A study of the predicted system load (database load), organized in subsections.

1.1. Tuple Estimation

Estimate of tuples at each relation.

Relation reference	Relation Name	Order of magnitude	Estimated growth
R01	Auction	tens of thousands	hundreds per week
R02	Authenticated_User	thousands	dozens per day
R03	Add_Credits	tens of thousands	hundreds per day
R04	Bid	hundreds of thousands	thousands per day
R05	Category	units	no growth

R06	City	thousands	units per month
R07	Comment	hundreds	dozens per day
R08	Country	hundreds	no growth
R09	Notification	hundreds of thousands	thousands per day
R10	Report	hundreds	dozens per month
R11	CategoryOfAuction	units	units per day
R12	Blocks	dozens	units per month
R13	Edit_Moderator	dozens	units per month
R14	Edit_Categories	units	no growth

1.2. Frequent Queries

Most important queries (SELECT) and their frequency.

Query reference	SELECT01
Query description	User's login
Query frequency	hundreds per day

```
SELECT password
FROM "authenticated_user"
```

```
WHERE "authenticated_user".username = $username;
```

Query reference	SELECT02
-----------------	----------

Query description	User's profile information
-------------------	----------------------------

Query frequency	hundreds per day
-----------------	------------------

```
SELECT username, pathToPhoto, email, "/rating",  
balance  
FROM "authenticated_user"  
WHERE "authenticated_user".ID = $userid;
```

Query reference	SELECT03
-----------------	----------

Query description	See a notification
-------------------	--------------------

Query frequency	dozens per day
-----------------	----------------

```
SELECT date, description , type  
FROM "notification"  
WHERE "notification".ID = $notificationid;
```

Query reference	SELECT04
-----------------	----------

Query description	See an auction
-------------------	----------------

Query frequency

hundreds per day

```
SELECT title, description, sellingReason,  
pathToPhoto, startingPrice, minimumSellingPrice,  
buyNow, startDate, limitDate, auctionCreator  
FROM "auction"  
WHERE "auction".ID = $auctionid;
```

```
SELECT MAX(value) as CurrentBid FROM "bid" WHERE  
"bid".auctionBidded=$auctionid;
```

Query reference

SELECT05

Query description

Get user's auctions

Query frequency

hundreds per day

```
SELECT "auction".ID, "auction".pathToPhoto,  
title, auctionCreator, "bid".value, "bid".date  
FROM "auction", "bid", "authenticated_user"  
WHERE "authenticated_user".ID = $userid AND  
"auction".ID = "bid".auctionbidded AND  
"authenticated_user".ID = "bid".bidder;
```

```
SELECT MAX(value) as CurrentBid FROM "bid" WHERE  
"bid".auctionBidded=$auctionid;
```

Query reference	SELECT06
Query description	Get auctions associated to category.
Query frequency	hundreds per day

```

SELECT title, pathToPhoto, startDate, limitDate,
numberOfBids
FROM "auction", "categoryOfAuction"
WHERE "auction".ID = "categoryOfAuction".auction
AND "categoryOfAuction".category = $categoryid AND
"auction".state='Active';

```

Query reference	SELECT07
Query description	Get auction's highest bidder.
Query frequency	hundreds per week

```

SELECT bidder
FROM "bid"
WHERE "bid".auctionBidded=$auctionid
ORDER BY value DESC;

```

Query reference	SELECT08
Query description	Get user's auctions he won
Query frequency	hundreds per month

```
SELECT pathToPhoto, title, finalPrice, finalDate
FROM "auction"
WHERE "auction".auctionWinner = $userId;
```

Query reference	SELECT09
Query description	Get user's "sold" auctions
Query frequency	hundreds per month

```
SELECT pathToPhoto, title, finalPrice, finalDate
FROM "auction"
WHERE "auction".auctionCreator = $userId AND
"auction".state='Over';
```

Query reference	SELECT10
Query description	Get moderator's active auctions
Query frequency	hundreds day

```
SELECT ID, pathToPhoto, title, auctionCreator,
minimumSellingPrice, buyNow, limitDate
FROM "auction"
WHERE responsibleModerator = $ID AND
state='Active';
```

SELECT MAX(value) as CurrentBid FROM "bid" WHERE "bid".auctionBidded = \$auctionId;

Query reference	SELECT11
Query description	Get pending auctions
Query frequency	hundreds per day

```
SELECT ID, pathToPhoto, title, auctionCreator,  
minimumSellingPrice, buyNow, limitDate  
FROM "auction"  
WHERE state='Pending';
```

Query reference	SELECT12
Query description	Get user's notifications
Query frequency	dozens per day

```
SELECT type  
FROM "notification"  
WHERE "notification".authenticatedUserID = $ID;
```

Query reference	SELECT13
Query description	Search an auction/user based on some words

Query frequency	hundreds per day
-----------------	------------------

```
SELECT ID, title
From "auction"
WHERE title ILIKE '%$search%' and state =
'Active'
```

```
SELECT ID, username From "authenticated_user" WHERE username ILIKE
'%$search%' or completename ILIKE '%$search%'
```

1.3. Frequent Updates

Most important updates (INSERT, UPDATE, DELETE) and their frequency.

Query reference	UPDATE01
Query description	Make Auction active
Query frequency	hundreds per week

```
UPDATE "auction"
SET state = 'Active', responsibleModerator =
$responsibleModerator
WHERE ID = $ID;
```

Query reference	UPDATE02
Query description	Finalize Auction

Query frequency

hundreds per week

```
UPDATE "auction"  
SET state='Over', auctionWinner = $winner,  
finalPrice = $finalPrice, finalDate= $finalDate  
WHERE ID = $ID;
```

Query reference UPDATE03

Query description Rate Auction

Query frequency hundreds per week

```
UPDATE "auction"  
SET rate = $rate  
WHERE ID = $ID;
```

Query reference UPDATE04

Query description Update Credit

Query frequency dozens per month

```
UPDATE "authenticated_user"  
SET balance = $credit  
WHERE ID = $ID;
```

Query reference	UPDATE05
------------------------	----------

Query description	Unblock/Block a previous blocked user
--------------------------	---------------------------------------

Query frequency	units per month
------------------------	-----------------

```
UPDATE "blocks"  
SET state = $state  
WHERE blocked = $ID;
```

Query reference	UPDATE06
------------------------	----------

Query description	Update user's profile photo
--------------------------	-----------------------------

Query frequency	units per month
------------------------	-----------------

```
UPDATE "authenticated_user"  
SET pathToPhoto = $pathToPhoto  
WHERE ID = $ID;
```

Query reference	UPDATE07
------------------------	----------

Query description	Update user's location
--------------------------	------------------------

Query frequency	units per month
------------------------	-----------------

```
UPDATE "authenticated_user"  
SET address = $address, postalCode = $postalCode,  
city = $city
```

```
WHERE ID = $ID;
```

Query reference	UPDATE08
Query description	Update user's password
Query frequency	units per month

```
UPDATE "authenticated_user"  
SET password = $password  
WHERE ID = $ID;
```

Query reference	INSERT01
Query description	New user registered
Query frequency	dozens per day

```
INSERT INTO "authenticated_user" (typeOfUser,  
username, password, pathToPhoto, completeName,  
email, birthDate, "/rating", address, postalCode,  
balance, city)  
VALUES ('Normal', $username, $password,  
$pathToPhoto, $completeName, $email, $birthDate,  
NULL, $address, $postalCode, 0, $city);
```

Query reference	INSERT02
Query description	New moderator registered

Query frequency

units per month

```
INSERT INTO "authenticated_user" (typeOfUser,
username, password, pathToPhoto, completeName,
email, birthDate, "/rating", address, postalCode,
balance, city)
VALUES ('Moderator', $username, $password, NULL,
NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL);
```

Query reference

INSERT03

Query description

Create an Auction

Query frequency

hundreds per week

```
INSERT INTO "auction" (state, title, description,
sellingReason, pathToPhoto, startingPrice,
minimumSellingPrice, buyNow, startDate, limitDate,
refusalDate, "/numberOfBids", reasonOfRefusal,
auctionWinner, auctionCreator,
responsibleModerator, rate, finalPrice, finalDate)
VALUES ('Pending', $title, $description,
$sellingReason, $pathToPhoto, $startingPrice,
$minimumSellingPrice, $buyNow, $startDate,
$limitDate, NULL, 0, NULL, NULL, $auctionCreator,
NULL, NULL, NULL, NULL);
```

Query reference

INSERT04

Query description

Bid

Query frequency

hundreds per day

```
INSERT INTO "bid" (auctionBidded, bidder, date,
value, isBuyNow)
VALUES ($auctionBidded, $bidder, $date, $value,
$isBuyNow);
```

Query reference

INSERT05

Query description

Add Credit

Query frequency

dozens per month

```
INSERT INTO "add_Credits" (value, date, user,
paypalID)
VALUES ($value, $date, $user, $paypalID);
```

Query reference

INSERT06

Query description

Add Category

Query frequency

units per year

```
INSERT INTO "category" (name, parent)
VALUES ($name, $parent);
```

Query reference

INSERT07

Query description

Make a report

Query frequency

dozens per month

```
INSERT INTO "report" (auctionID, normalUserID,  
date, reason)  
VALUES ($auctionID, $normalUserID, $date,  
$reason);
```

Query reference

INSERT08

Query description

Comment an auction

Query frequency

dozens per day

```
INSERT INTO "comment" (date, description,  
auctionCommented, userCommenter)  
VALUES ($date, $description, $auctionCommented,  
$userCommenter);
```

Query reference

INSERT08

Query description

Block an user

Query frequency

units per month

```
INSERT INTO "blocks" (blocked, blocker, state,  
description, date)  
VALUES ($blocked, $blocker, 'Blocked',
```

```
$description, $date);
```

Query reference	DELETE01
Query description	Delete Moderator
Query frequency	units per year

```
DELETE FROM "authenticated_user"  
WHERE ID = $ID;
```

Query reference	DELETE02
Query description	Delete a comment
Query frequency	units per month

```
DELETE FROM "comment"  
WHERE ID = $ID;
```

Query reference	DELETE03
Query description	Delete a notification
Query frequency	dozens per day

```
DELETE FROM "notification"  
WHERE ID = $ID;
```

2. Proposed Indexes

2.1. Performance Indexes

Indexes proposed to improve performance of the identified queries.

Index reference	IDX01
Related queries	SELECT02
Index relation	authenticated_user
Index attribute	id
Index type	Hash
Cardinality	high
Clustering	No
Justification	Used to search for the user's information. This query has to be fast because it's executed several times and the table has many entries. Cardinality is high due to the fact that id is unique and so it doesn't need ranged query support. This being said, clustering won't be used.


```
CREATE INDEX user_ind ON authenticated_user USING  
hash (id);
```

Index reference	IDX02
Related queries	SELECT07
Index relation	bid
Index attribute	auctionbidded
Index type	B-tree
Cardinality	medium
Clustering	Yes
Justification	Used to search the bids of an auction and has to be fast because it's executed several times. Cardinality is medium so it's a good candidate for clustering.

```
CREATE INDEX bid_ind ON bid USING btree  
(auctionbidded);
```

Index reference	IDX03
Related queries	SELECT11

Index relation	auction
Index attribute	state
Index type	B-tree
Cardinality	low
Clustering	Yes
Justification	Used to search the pending auctions. Cardinality is low and it needs a ranged query so we use clustering.

```
CREATE INDEX pending_ind ON auction USING btree
(state);
```

2.2. Full-text Search Indices

The system being developed must provide full-text search features supported by PostgreSQL. Thus, it is necessary to specify the fields where full-text search will be available and the associated setup, namely all necessary configurations, indexes definitions and other relevant details.

Index reference	IDX04
Related queries	SELECT13
Index relation	auction

Index attribute	title
Index type	GiST
Clustering	No
Justification	To improve the performance of full text searches while searching for the titles of the auctions; GiST because it's better for dynamic data.

```
CREATE INDEX search_title_idx ON auction USING
GiST (to_tsvector('english', title))
```

Index reference	IDX05
Related queries	SELECT13
Index relation	authenticated_user
Index attribute	username
Index type	GiST
Clustering	No
Justification	To improve the performance of full text searches while searching for the usernames of users; GiST because it's better for dynamic data.

```
CREATE INDEX search_username_idx ON
```

```
authenticated_user USING GIST  
(to_tsvector('english', username))
```

Index reference	IDX06
Related queries	SELECT13
Index relation	authenticated_user
Index attribute	completename
Index type	GiST
Clustering	No
Justification	To improve the performance of full text searches while searching for the names of users; GiST because it's better for dynamic data.

```
CREATE INDEX search_name_idx ON authenticated_user  
USING GIST (to_tsvector('english', completename))
```

3. Triggers

User-defined functions and trigger procedures that add control structures to the SQL language or perform complex computations, are identified and described to be trusted by the database server. Every kind of function (SQL functions, Stored procedures, Trigger procedures) can take base types, composite types, or

combinations of these as arguments (parameters). In addition, every kind of function can return a base type or a composite type. Functions can also be defined to return sets of base or composite values.

3.1 User-defined functions

UDF reference	UDF01
UDF description	Check if auction creator and auction winner (in case of existence) are Normal users and responsiblemoderator is Administrator or Moderator (in case of existence).

```
CREATE FUNCTION check_auction_users(auctioncreator
integer, auctionwinner integer,
responsiblemoderator integer) RETURNS boolean
LANGUAGE plpgsql
AS $$DECLARE passed BOOLEAN;
DECLARE creatorUser TypeOfUser;
DECLARE winnerUser TypeOfUser;
DECLARE responsibleUser TypeOfUser;
BEGIN
    SELECT typeOfUser INTO creatorUser FROM
Authenticated_User where id = auctioncreator ;
    IF(auctionwinner IS NULL)
        THEN winnerUser := NULL ;
        ELSE SELECT typeOfUser INTO winnerUser
FROM Authenticated_User where id = auctionwinner ;
    END IF;
    IF(responsiblemoderator IS NULL)
        THEN responsibleUser := NULL ;
        ELSE SELECT typeOfUser INTO
responsibleUser FROM Authenticated_User where id =
responsiblemoderator ;
    END IF;
```

```

        IF creatorUser = 'Normal'::TypeOfUser AND
(winnerUser = 'Normal'::TypeOfUser OR winnerUser IS
NULL) AND (
responsibleUser = 'Moderator'::TypeOfUser OR
responsibleUser = 'Administrator'::TypeOfUser OR
responsibleUser IS NULL
)
        THEN passed := true;
        ELSE passed := false;
        END IF;
        RETURN passed;
END;$$;

```

**UDF
reference**

UDF02

**UDF
description**

Check in block association if user blocked is a Normal user and blocker is an Administrator or Moderator.

```

CREATE FUNCTION check_block_users(blocked integer,
blocker integer) RETURNS boolean
LANGUAGE plpgsql
AS $$DECLARE passed BOOLEAN;
DECLARE blockedUser TypeOfUser;
DECLARE blokerUser TypeOfUser;
BEGIN
        SELECT typeOfUser INTO blockedUser
FROM Authenticated_User where id = blocked ;
        SELECT typeOfUser INTO blokerUser
FROM Authenticated_User where id = blocker ;
        IF blockedUser =
'Normal'::TypeOfUser AND (blokerUser =
'Administrator'::TypeOfUser OR blokerUser =
'Moderator'::TypeOfUser)
        THEN passed := true;
        ELSE passed := false;

```

```

                                END IF;
RETURN passed;
END;$$;

```

UDF reference UDF03

UDF description Check in edit moderators association if edited Moderator is a Moderator user and editor is an Administrator.

```

CREATE FUNCTION check_edit_moderators(removedmod
integer, removeradmin integer) RETURNS boolean
LANGUAGE plpgsql
AS $$DECLARE passed boolean;
DECLARE removerModUser TypeOfUser;
DECLARE removerAdminUser TypeOfUser;
BEGIN
    SELECT typeOfUser INTO removerModUser FROM
Authenticated_User where id = removedMod ;
    SELECT typeOfUser INTO removerAdminUser
FROM Authenticated_User where id = removerAdmin ;
    IF removerModUser = 'Moderator'::TypeOfUser
AND removerAdminUser = 'Administrator'::TypeOfUser
    THEN passed := true;
    ELSE passed := false;
    END IF;
RETURN passed;
END;$$;

```

UDF reference UDF04

UDF description	Check if rejected auction has date of rejection and reason.
------------------------	---

```

CREATE FUNCTION check_rejected_auction(state
auctionstate, dateofrefusal timestamp with time
zone, reasonofrefusal character varying) RETURNS
boolean
    LANGUAGE plpgsql
    AS $$DECLARE pass boolean;
BEGIN
    IF state = 'Rejected'::auctionstate
        THEN IF dateOfRefusal IS NOT NULL AND
reasonOfRefusal IS NOT NULL
            THEN pass := true;
            ELSE pass := false;
            END IF;
        ELSE pass := true;
        END IF;
    RETURN pass;
END;$$;

```

UDF reference	UDF05
----------------------	-------

UDF description	Check in edit category association if editor is an Administrator.
------------------------	---

```

CREATE FUNCTION check_admin_modify_category(admin
integer) RETURNS boolean
    LANGUAGE plpgsql
    AS $$DECLARE passed BOOLEAN;
DECLARE adminUser TypeOfUser;
BEGIN
    SELECT typeOfUser INTO adminUser FROM

```



```

Authenticated_User where id = admin ;
IF
adminUser = 'Administrator'::TypeOfUser THEN passed
:= true; ELSE passed := false;
END IF;
RETURN passed;
END;
$$;

```

UDF

reference

UDF06

UDF

description

Check if an auction that is over has all obligatory camps not null.

```

CREATE FUNCTION check_auction_win(state
auctionstate, finaldate timestamp with time zone,
finalprice integer, auctionwinner integer) RETURNS
boolean
LANGUAGE plpgsql
AS $$DECLARE pass boolean;
BEGIN
IF state = 'Over'::auctionstate
THEN IF finaldate IS NOT NULL AND finalprice
IS NOT NULL AND auctionwinner IS NOT NULL
THEN pass := true;
ELSE pass := false;
END IF;
ELSE pass := true;
END IF;
RETURN pass;
END;$$;

```

3.2 Triggers

Trigger reference	TRIGGER01
Trigger description	An auction cannot be bade on by its own creator.

```
CREATE FUNCTION auction_creator() RETURNS trigger
LANGUAGE plpgsql
AS $$BEGIN
  IF EXISTS (SELECT * FROM auction
    INNER JOIN Authenticated_User ON
Authenticated_User.id = auction.auctioncreator
    WHERE NEW.auctionbided = auction.id AND
NEW.bidder = Authenticated_User.id) THEN
    RAISE EXCEPTION 'An auction cannot be bade on
by its creator.';
  END IF;
  RETURN NEW;
END;
$$;
```

Trigger reference	TRIGGER02
Trigger description	A bid on this auction has to have a greater value than a previous one on this auction. When this happens it updates the user that made the last bid with highest value so that its balance is restored and generates a notification for that user.

```
CREATE FUNCTION bid_greater_than_last() RETURNS
trigger
```

```

LANGUAGE plpgsql
AS $$BEGIN
    IF EXISTS (SELECT * FROM bid WHERE
NEW.auctionbided = auctionbided AND NEW.value

```

**Trigger
reference** TRIGGER03

**Trigger
description** Checks if bidder has enough balance to make a bid of
this value and if it has discounts it on the balance

```

CREATE FUNCTION bidder_has_money() RETURNS trigger
LANGUAGE plpgsql
AS $$BEGIN
    IF EXISTS (SELECT * FROM Authenticated_User
WHERE NEW.bidder = id AND NEW.value > balance)
THEN
        RAISE EXCEPTION 'To make a bid on
this auction the bidder must have a balance greater
than bid value';
ELSE IF EXISTS (SELECT * FROM Authenticated_User
WHERE NEW.bidder = id AND NEW.value <= balance)="
then="" update="" authenticated_user="" set=""
balance="(SELECT" from="" where="" new.bidder="id)"
-="" new.value="" id="NEW.bidder;" end="" if;=""
return="" new;="" end$$;="" <="" pre="">

```

**Trigger
reference** TRIGGER04

**Trigger
description** When the rating of an auction is updated, the rating of
its respective creator is also updated with the media of
all rated auctions created by this user.

```

CREATE FUNCTION update_ratings() RETURNS trigger
LANGUAGE plpgsql
AS $$DECLARE sumOfExistRates integer;
DECLARE countOfExistRates integer;
BEGIN
    IF NEW.rate IS NOT NULL
    THEN
        sumOfExistRates := (SELECT SUM(rate) FROM
        auction where auctioncreator = NEW.auctioncreator
        AND rate IS NOT NULL) ;
        countOfExistRates := (SELECT COUNT(rate)
        FROM auction where auctioncreator =
        NEW.auctioncreator AND rate IS NOT NULL) ;
        IF countOfExistRates IS NOT NULL
        THEN UPDATE Authenticated_User SET
        "/rating" = ((sumOfExistRates + NEW.rate) /
        (countOfExistRates + 1)) WHERE NEW.auctioncreator =
        id;
        ELSE UPDATE Authenticated_User SET
        "/rating" = NEW.rate;
        END IF;
    END IF;
    RETURN NEW;
END;$$;

```

Trigger reference	TRIGGER05
--------------------------	-----------

Trigger description	If bid is buy now it terminates auction.
----------------------------	--

```

CREATE FUNCTION buy_now() RETURNS trigger
LANGUAGE plpgsql
AS $$BEGIN

```

```

IF EXISTS (SELECT * FROM auction WHERE
NEW.auctionbidded = id AND NEW.value = buyNow AND
NEW."isBuyNow" = true)
THEN
UPDATE auction SET state = 'Over'::auctionstate,
finaldate = NEW.date, finalprice = NEW.value,
auctionwinner = NEW.bidder WHERE id =
NEW.auctionbidded;
END IF;
RETURN NEW;
END;$$;

```

**Trigger
reference**

TRIGGER06

**Trigger
description**

A Bid has to have a greater value than starting
price of auction.

```

CREATE FUNCTION check_bid_value() RETURNS trigger
LANGUAGE plpgsql
AS $$BEGIN
IF EXISTS (SELECT * FROM auction WHERE
NEW.auctionbidded = id AND NEW.value <
startingprice)
THEN
RAISE EXCEPTION 'A Bid has to have a greater value
than starting price of auction.';
END IF;
RETURN NEW;
END;$$;

```

**Trigger
reference**

TRIGGER07

Trigger description	If current date is greater than limit date it updates auction to state over and the highest bidder wins the auction.
----------------------------	--

```

CREATE FUNCTION win_auction() RETURNS trigger
LANGUAGE plpgsql
AS $$BEGIN
IF (transaction_timestamp() >= NEW.limitdate AND
NEW.state = 'Active'::auctionstate)
THEN
UPDATE auction SET state = 'Over'::auctionstate,
finaldate = ( SELECT date
FROM "bid"
WHERE "bid".auctionBidded= NEW.id
ORDER BY value DESC ), finalprice = ( SELECT
value
FROM "bid"
WHERE "bid".auctionBidded= NEW.id
ORDER BY value DESC ), auctionwinner = ( SELECT
bidder
FROM "bid"
WHERE "bid".auctionBidded=NEW.id
ORDER BY value DESC ) WHERE id = NEW.id;
END IF;
RETURN NEW;
END;$$;

```

4. Complete SQL Code

The database script must also include the SQL to populate a database with test data with an amount of tuples suitable for testing and with plausible values for the fields of the database. This code should also be included in the group's github repository as an SQL script, and a link include here.

```
--  
-- PostgreSQL database  
--  
  
SET statement_timeout = 0;  
SET lock_timeout = 0;  
SET client_encoding = 'UTF8';  
SET standard_conforming_strings = on;  
SET check_function_bodies = false;  
SET client_min_messages = warning;  
  
SET search_path = public, pg_catalog;  
  
ALTER TABLE ONLY public.report DROP CONSTRAINT  
report_normaluserid_fkey;  
ALTER TABLE ONLY public.report DROP CONSTRAINT  
report_auctionid_fkey;  
ALTER TABLE ONLY public.notification DROP CONSTRAINT  
notification_authenticated_userid_fkey;  
ALTER TABLE ONLY public.notification DROP CONSTRAINT  
notification_auctionassociated_fkey;  
ALTER TABLE ONLY public.edit_moderator DROP  
CONSTRAINT edit_moderator_removaladmin_fkey;  
ALTER TABLE ONLY public.edit_moderator DROP  
CONSTRAINT edit_moderator_removedmod_fkey;  
ALTER TABLE ONLY public.edit_categories DROP  
CONSTRAINT edit_categories_category_fkey;  
ALTER TABLE ONLY public.edit_categories DROP  
CONSTRAINT edit_categories_admin_fkey;  
ALTER TABLE ONLY public.comment DROP CONSTRAINT  
comment_usercommenter_fkey;  
ALTER TABLE ONLY public.comment DROP CONSTRAINT  
comment_auctioncommented_fkey;  
ALTER TABLE ONLY public.city DROP CONSTRAINT  
city_country_fkey;  
ALTER TABLE ONLY public.categoryofauction DROP
```

```

CONSTRAINT categoryofauction_category_fkey;
ALTER TABLE ONLY public.categoryofauction DROP
CONSTRAINT categoryofauction_auction_fkey;
ALTER TABLE ONLY public.category DROP CONSTRAINT
category_parent_fkey;
ALTER TABLE ONLY public.blocks DROP CONSTRAINT
blocks_blocker_fkey;
ALTER TABLE ONLY public.blocks DROP CONSTRAINT
blocks_blocked_fkey;
ALTER TABLE ONLY public.bid DROP CONSTRAINT
bid_bidder_fkey;
ALTER TABLE ONLY public.bid DROP CONSTRAINT
bid_auctionbidded_fkey;
ALTER TABLE ONLY public.authenticated_user DROP
CONSTRAINT authenticated_user_city_fkey;
ALTER TABLE ONLY public.auction DROP CONSTRAINT
auction_responsiblemoderator_fkey;
ALTER TABLE ONLY public.auction DROP CONSTRAINT
auction_auctionwinner_fkey;
ALTER TABLE ONLY public.auction DROP CONSTRAINT
auction_auctioncreator_fkey;
ALTER TABLE ONLY public.add_credits DROP CONSTRAINT
add_credits_user_fkey;
DROP TRIGGER win_auction ON public.bid;
DROP TRIGGER update_rating ON public.auction;
DROP TRIGGER check_bid_value ON public.bid;
DROP TRIGGER buy_now ON public.bid;
DROP TRIGGER bidder_has_money ON public.bid;
DROP TRIGGER bid_greater_than_last ON public.bid;
DROP TRIGGER auction_creator ON public.bid;
ALTER TABLE ONLY public.report DROP CONSTRAINT
report_pkey;
ALTER TABLE ONLY public.notification DROP CONSTRAINT
notification_pkey;
ALTER TABLE ONLY public.edit_moderator DROP
CONSTRAINT edit_moderator_pkey;
ALTER TABLE ONLY public.edit_categories DROP
CONSTRAINT edit_categories_pkey;
ALTER TABLE ONLY public.country DROP CONSTRAINT

```



```
country_pkey;
ALTER TABLE ONLY public.country DROP CONSTRAINT
country_name_key;
ALTER TABLE ONLY public.comment DROP CONSTRAINT
comment_pkey;
ALTER TABLE ONLY public.city DROP CONSTRAINT
city_pkey;
ALTER TABLE ONLY public.city DROP CONSTRAINT
city_name_key;
ALTER TABLE ONLY public.categoryofauction DROP
CONSTRAINT categoryofauction_pkey;
ALTER TABLE ONLY public.category DROP CONSTRAINT
category_pkey;
ALTER TABLE ONLY public.category DROP CONSTRAINT
category_name_key;
ALTER TABLE ONLY public.blocks DROP CONSTRAINT
blocks_pkey;
ALTER TABLE ONLY public.bid DROP CONSTRAINT bid_pkey;
ALTER TABLE ONLY public.authenticated_user DROP
CONSTRAINT authenticated_user_username_key;
ALTER TABLE ONLY public.authenticated_user DROP
CONSTRAINT authenticated_user_pkey;
ALTER TABLE ONLY public.authenticated_user DROP
CONSTRAINT authenticated_user_email_key;
ALTER TABLE ONLY public.auction DROP CONSTRAINT
auction_pkey;
ALTER TABLE ONLY public.add_credits DROP CONSTRAINT
add_credits_pkey;
DROP TABLE public.report;
DROP TABLE public.notification;
DROP TABLE public.lastbidder;
DROP TABLE public.edit_moderator;
DROP TABLE public.edit_categories;
DROP TABLE public.country;
DROP TABLE public.comment;
DROP TABLE public.city;
DROP TABLE public.categoryofauction;
DROP TABLE public.category;
DROP TABLE public.blocks;
```

```

DROP TABLE public.bid;
DROP SEQUENCE public.auto_increment_notification;
DROP SEQUENCE public.auto_increment_comment;
DROP SEQUENCE public.auto_increment_city;
DROP SEQUENCE public.auto_increment_category;
DROP TABLE public.authenticated_user;
DROP SEQUENCE public.auto_increment_user;
DROP TABLE public.auction;
DROP SEQUENCE public.auto_increment_auction;
DROP TABLE public.add_credits;
DROP SEQUENCE public.auto_increment_credits;
DROP FUNCTION public.win_auction();
DROP FUNCTION public.update_ratings();
DROP FUNCTION public.check_rejected_auction(state
auctionstate, dateofrefusal timestamp with time zone,
reasonofrefusal character varying);
DROP FUNCTION public.check_edit_moderators(removedmod
integer, removeradmin integer);
DROP FUNCTION public.check_block_users(blocked
integer, blocker integer);
DROP FUNCTION public.check_bid_value();
DROP FUNCTION public.check_auction_win(state
auctionstate, finaldate timestamp with time zone,
finalprice integer, auctionwinner integer);
DROP FUNCTION
public.check_auction_users(auctioncreator integer,
auctionwinner integer, responsiblemoderator integer);
DROP FUNCTION
public.check_admin_modify_category(admin integer);
DROP FUNCTION public.buy_now();
DROP FUNCTION public.bidder_has_money();
DROP FUNCTION public.bid_greater_than_last();
DROP FUNCTION public.auction_creator();
DROP TYPE public.typeofuser;
DROP TYPE public.blockingstate;
DROP TYPE public.auctionstate;
DROP SCHEMA public;
--
-- Name: public; Type: SCHEMA; Schema: -; Owner:

```

```

lbaw1716
--

CREATE SCHEMA public;

ALTER SCHEMA public OWNER TO lbaw1716;

--
-- Name: SCHEMA public; Type: COMMENT; Schema: -;
Owner: lbaw1716
--

COMMENT ON SCHEMA public IS 'standard public schema';

SET search_path = public, pg_catalog;

--
-- Name: auctionstate; Type: TYPE; Schema: public;
Owner: lbaw1716
--

CREATE TYPE auctionstate AS ENUM (
    'Active',
    'Rejected',
    'Pending',
    'Over'
);

ALTER TYPE auctionstate OWNER TO lbaw1716;

--
-- Name: blockingstate; Type: TYPE; Schema: public;
Owner: lbaw1716
--

CREATE TYPE blockingstate AS ENUM (

```

```
        'Blocked',
        'Allowed'
    );
```

```
ALTER TYPE blockingstate OWNER TO lbaw1716;
```

```
--
```

```
-- Name: typeofuser; Type: TYPE; Schema: public;
Owner: lbaw1716
```

```
--
```

```
CREATE TYPE typeofuser AS ENUM (
    'Moderator',
    'Administrator',
    'Normal'
);
```

```
ALTER TYPE typeofuser OWNER TO lbaw1716;
```

```
--
```

```
-- USER FUNCTIONS AND TRIGGERS
```

```
--
```

```
--
```

```
-- Name: auction_creator(); Type: FUNCTION; Schema:
public; Owner: lbaw1716
```

```
--
```

```
CREATE FUNCTION auction_creator() RETURNS trigger
    LANGUAGE plpgsql
    AS $$BEGIN
    IF EXISTS (SELECT * FROM auction
        INNER JOIN Authenticated_User ON
Authenticated_User.id = auction.auctioncreator
        WHERE NEW.auctionbided = auction.id AND
NEW.bidder = Authenticated_User.id) THEN
        RAISE EXCEPTION 'An auction cannot be bade on by
its creator.';
```

```

    END IF;
    RETURN NEW;
END;
$$;

ALTER FUNCTION public.auction_creator() OWNER TO
lbaw1716;

--
-- Name: bid_greater_than_last(); Type: FUNCTION;
Schema: public; Owner: lbaw1716
--

CREATE FUNCTION bid_greater_than_last() RETURNS
trigger
    LANGUAGE plpgsql
    AS $$BEGIN
        IF EXISTS (SELECT * FROM bid WHERE
NEW.auctionbidded = auctionbidded AND NEW.value <=
value) THEN
            RAISE EXCEPTION 'A bid on this
auction has to have a greater value than a previous
one on this auction.';
        ELSE
            UPDATE Authenticated_User SET balance =
((SELECT value FROM "bid" WHERE
"bid".auctionBidded=NEW.auctionbidded ORDER BY value
DESC LIMIT 1) + (SELECT balance FROM
Authenticated_User WHERE id = (SELECT bidder FROM
"bid" WHERE auctionbidded = NEW.auctionbidded ORDER
BY value DESC LIMIT 1))) WHERE id = (SELECT bidder
FROM "bid" WHERE auctionbidded = NEW.auctionbidded
ORDER BY value DESC LIMIT 1);
            INSERT INTO Notification (id, date,
description, type, auctionassociated,
authenticated_userid) VALUES (DEFAULT,
transaction_timestamp(), 'Your bid on this auction
was surpassed. Try again!', 'Bid Exceeded',

```

```

NEW.auctionbidded, (SELECT bidder FROM "bid" WHERE
auctionbidded = NEW.auctionbidded ORDER BY value DESC
LIMIT 1));
        END IF;
        RETURN NEW;
END$$;

```

```

ALTER FUNCTION public.bid_greater_than_last() OWNER
TO lbaw1716;

```

```

--
-- Name: FUNCTION bid_greater_than_last(); Type:
COMMENT; Schema: public; Owner: lbaw1716
--

```

```

COMMENT ON FUNCTION bid_greater_than_last() IS '
';

```

```

--
-- Name: bidder_has_money(); Type: FUNCTION; Schema:
public; Owner: lbaw1716
--

```

```

CREATE FUNCTION bidder_has_money() RETURNS trigger
LANGUAGE plpgsql
AS $$BEGIN
        IF EXISTS (SELECT * FROM Authenticated_User
WHERE NEW.bidder = id AND NEW.value > balance)
THEN
                RAISE EXCEPTION 'To make a bid on
this auction the bidder must have a balance greater
than bid value';
ELSE IF EXISTS (SELECT * FROM Authenticated_User
WHERE NEW.bidder = id AND NEW.value <= balance)
THEN
        UPDATE Authenticated_User SET balance = (SELECT
balance FROM Authenticated_User WHERE NEW.bidder =

```

```

id) - NEW.value WHERE id = NEW.bidder;
END IF;
        END IF;
        RETURN NEW;
END$$;

```

```

ALTER FUNCTION public.bidder_has_money() OWNER TO
lbaw1716;

```

```

--
-- Name: buy_now(); Type: FUNCTION; Schema: public;
Owner: lbaw1716
--

```

```

CREATE FUNCTION buy_now() RETURNS trigger
    LANGUAGE plpgsql
    AS $$BEGIN
IF EXISTS (SELECT * FROM auction WHERE
NEW.auctionbided = id AND NEW.value = buyNow AND
NEW."isBuyNow" = true)
THEN
UPDATE auction SET state = 'Over'::auctionstate,
finaldate = NEW.date, finalprice = NEW.value,
auctionwinner = NEW.bidder WHERE id =
NEW.auctionbided;
END IF;
RETURN NEW;
END;$$;

```

```

ALTER FUNCTION public.buy_now() OWNER TO lbaw1716;

```

```

--
-- Name: check_admin_modify_category(integer); Type:
FUNCTION; Schema: public; Owner: lbaw1716
--

```

```

CREATE FUNCTION check_admin_modify_category(admin

```

```

integer) RETURNS boolean
    LANGUAGE plpgsql
    AS $$DECLARE passed BOOLEAN;
DECLARE adminUser TypeOfUser;
BEGIN
SELECT typeOfUser INTO adminUser FROM
Authenticated_User where id = admin ;
IF
adminUser = 'Administrator'::TypeOfUser THEN passed
:= true; ELSE passed := false;
END IF;
RETURN passed;
END;
$$;

```

```

ALTER FUNCTION
public.check_admin_modify_category(admin integer)
OWNER TO lbaw1716;

```

```

--
-- Name: check_auction_users(integer, integer,
integer); Type: FUNCTION; Schema: public; Owner:
lbaw1716
--

```

```

CREATE FUNCTION check_auction_users(auctioncreator
integer, auctionwinner integer, responsiblemoderator
integer) RETURNS boolean
    LANGUAGE plpgsql
    AS $$DECLARE passed BOOLEAN;
DECLARE creatorUser TypeOfUser;
DECLARE winnerUser TypeOfUser;
DECLARE responsibleUser TypeOfUser;
BEGIN
    SELECT typeOfUser INTO creatorUser FROM
Authenticated_User where id = auctioncreator ;
    IF(auctionwinner IS NULL)
        THEN winnerUser := NULL ;

```



```

        ELSE SELECT typeOfUser INTO winnerUser
FROM Authenticated_User where id = auctionwinner ;
    END IF;
    IF(responsiblemoderator IS NULL)
        THEN responsibleUser := NULL ;
        ELSE SELECT typeOfUser INTO
responsibleUser FROM Authenticated_User where id =
responsiblemoderator ;
    END IF;
    IF creatorUser = 'Normal'::TypeOfUser AND
(winnerUser = 'Normal'::TypeOfUser OR winnerUser IS
NULL) AND (
    responsibleUser = 'Moderator'::TypeOfUser OR
    responsibleUser = 'Administrator'::TypeOfUser OR
    responsibleUser IS NULL
)
        THEN passed := true;
        ELSE passed := false;
        END IF;
    RETURN passed;
END;$$;

```

```

ALTER FUNCTION
public.check_auction_users(auctioncreator integer,
auctionwinner integer, responsiblemoderator integer)
OWNER TO lbaw1716;

```

```

--
-- Name: check_auction_win(auctionstate, timestamp
with time zone, integer, integer); Type: FUNCTION;
Schema: public; Owner: lbaw1716
--

```

```

CREATE FUNCTION check_auction_win(state auctionstate,
finaldate timestamp with time zone, finalprice
integer, auctionwinner integer) RETURNS boolean
    LANGUAGE plpgsql
    AS $$DECLARE pass boolean;

```

```

BEGIN
  IF state = 'Over'::auctionstate
    THEN IF finaldate IS NOT NULL AND finalprice IS
NOT NULL AND auctionwinner IS NOT NULL
      THEN pass := true;
      ELSE pass := false;
      END IF;
    ELSE pass := true;
    END IF;
  RETURN pass;
END;$$;

```

```

ALTER FUNCTION public.check_auction_win(state
auctionstate, finaldate timestamp with time zone,
finalprice integer, auctionwinner integer) OWNER TO
lbaw1716;

```

```

--
-- Name: check_bid_value(); Type: FUNCTION; Schema:
public; Owner: lbaw1716
--

```

```

CREATE FUNCTION check_bid_value() RETURNS trigger
  LANGUAGE plpgsql
  AS $$BEGIN
IF EXISTS (SELECT * FROM auction WHERE
NEW.auctionbided = id AND NEW.value < startingprice)
THEN
RAISE EXCEPTION 'A Bid has to have a greater value
than starting price of auction.';
END IF;
RETURN NEW;
END;$$;

```

```

ALTER FUNCTION public.check_bid_value() OWNER TO
lbaw1716;

```

```

--
-- Name: check_block_users(integer, integer); Type:
FUNCTION; Schema: public; Owner: lbaw1716
--

CREATE FUNCTION check_block_users(blocked integer,
blocker integer) RETURNS boolean
    LANGUAGE plpgsql
    AS $$DECLARE passed BOOLEAN;
DECLARE blokedUser TypeOfUser;
DECLARE blokerUser TypeOfUser;
BEGIN
        SELECT typeOfUser INTO blokedUser
FROM Authenticated_User where id = blocked ;
        SELECT typeOfUser INTO blokerUser
FROM Authenticated_User where id = blocker ;
        IF blokedUser = 'Normal'::TypeOfUser
AND (blokerUser = 'Administrator'::TypeOfUser OR
blokerUser = 'Moderator'::TypeOfUser)
        THEN passed := true;
        ELSE passed := false;
        END IF;
RETURN passed;
END;$$;

ALTER FUNCTION public.check_block_users(blocked
integer, blocker integer) OWNER TO lbaw1716;

--
-- Name: check_edit_moderators(integer, integer);
Type: FUNCTION; Schema: public; Owner: lbaw1716
--

CREATE FUNCTION check_edit_moderators(removedmod
integer, removeradmin integer) RETURNS boolean
    LANGUAGE plpgsql
    AS $$DECLARE passed boolean;
DECLARE removerModUser TypeOfUser;

```

```

DECLARE removerAdminUser TypeOfUser;
BEGIN
    SELECT typeOfUser INTO removerModUser FROM
Authenticated_User where id = removedMod ;
    SELECT typeOfUser INTO removerAdminUser FROM
Authenticated_User where id = removerAdmin ;
    IF removerModUser = 'Moderator'::TypeOfUser
AND removerAdminUser = 'Administrator'::TypeOfUser
    THEN passed := true;
    ELSE passed := false;
    END IF;
RETURN passed;
END;$$;

ALTER FUNCTION
public.check_edit_moderators(removedmod integer,
removeradmin integer) OWNER TO lbaw1716;

--
-- Name: check_rejected_auction(auctionstate,
timestamp with time zone, character varying); Type:
FUNCTION; Schema: public; Owner: lbaw1716
--

CREATE FUNCTION check_rejected_auction(state
auctionstate, dateofrefusal timestamp with time zone,
reasonofrefusal character varying) RETURNS boolean
    LANGUAGE plpgsql
    AS $$DECLARE pass boolean;
BEGIN
    IF state = 'Rejected'::auctionstate
        THEN IF dateOfRefusal IS NOT NULL AND
reasonOfRefusal IS NOT NULL
            THEN pass := true;
            ELSE pass := false;
            END IF;
        ELSE pass := true;
        END IF;
END IF;

```

```
    RETURN pass;
END;$$;
```

```
ALTER FUNCTION public.check_rejected_auction(state
auctionstate, dateofrefusal timestamp with time zone,
reasonofrefusal character varying) OWNER TO lbaw1716;
```

```
--
-- Name: update_ratings(); Type: FUNCTION; Schema:
public; Owner: lbaw1716
--
```

```
CREATE FUNCTION update_ratings() RETURNS trigger
    LANGUAGE plpgsql
    AS $$DECLARE sumOfExistRates integer;
DECLARE countOfExistRates integer;
BEGIN
    IF NEW.rate IS NOT NULL
    THEN
        sumOfExistRates := (SELECT SUM(rate) FROM
auction where auctioncreator = NEW.auctioncreator AND
rate IS NOT NULL) ;
        countOfExistRates := (SELECT COUNT(rate) FROM
auction where auctioncreator = NEW.auctioncreator AND
rate IS NOT NULL) ;
        IF countOfExistRates IS NOT NULL
        THEN UPDATE Authenticated_User SET "/rating"
= ((sumOfExistRates + NEW.rate) / (countOfExistRates
+ 1)) WHERE NEW.auctioncreator = id;
        ELSE UPDATE Authenticated_User SET "/rating"
= NEW.rate;
        END IF;
    END IF;
    RETURN NEW;
END;$$;
```

```
ALTER FUNCTION public.update_ratings() OWNER TO
```

```

lbaw1716;

--
-- Name: win_auction(); Type: FUNCTION; Schema:
public; Owner: lbaw1716
--

CREATE FUNCTION win_auction() RETURNS trigger
    LANGUAGE plpgsql
    AS $$BEGIN
IF (transaction_timestamp() >= NEW.limitdate AND
NEW.state = 'Active'::auctionstate)
THEN
UPDATE auction SET state = 'Over'::auctionstate,
finaldate = ( SELECT date
FROM "bid"
WHERE "bid".auctionBidded= NEW.id
ORDER BY value DESC ), finalprice = ( SELECT
value
FROM "bid"
WHERE "bid".auctionBidded= NEW.id
ORDER BY value DESC ), auctionwinner = ( SELECT
bidder
FROM "bid"
WHERE "bid".auctionBidded=NEW.id
ORDER BY value DESC ) WHERE id = NEW.id;
END IF;
RETURN NEW;
END;$$;

ALTER FUNCTION public.win_auction() OWNER TO
lbaw1716;

--
-- Name: auto_increment_credits; Type: SEQUENCE;
Schema: public; Owner: lbaw1716
--

```

```

CREATE SEQUENCE auto_increment_credits
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE auto_increment_credits OWNER TO lbaw1716;

SET default_tablespace = '';

SET default_with_oids = false;

--
-- Name: add_credits; Type: TABLE; Schema: public;
-- Owner: lbaw1716; Tablespace:
--

CREATE TABLE add_credits (
    id integer DEFAULT
nextval('auto_increment_credits'::regclass) NOT NULL,
    value integer NOT NULL,
    date timestamp with time zone DEFAULT
transaction_timestamp() NOT NULL,
    paypalid character varying NOT NULL,
    "user" integer NOT NULL
);

ALTER TABLE add_credits OWNER TO lbaw1716;

--
-- Name: auto_increment_auction; Type: SEQUENCE;
-- Schema: public; Owner: lbaw1716
--

CREATE SEQUENCE auto_increment_auction
    START WITH 1

```

```
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE
CACHE 1;
```

```
ALTER TABLE auto_increment_auction OWNER TO lbaw1716;
```

```
--
-- Name: auction; Type: TABLE; Schema: public; Owner:
lbaw1716; Tablespace:
--
```

```
CREATE TABLE auction (
    id integer DEFAULT
nextval('auto_increment_auction'::regclass) NOT NULL,
    state auctionstate NOT NULL,
    title character varying NOT NULL,
    description character varying,
    sellingreason character varying,
    pathtophoto character varying,
    startingprice integer NOT NULL,
    minimumsellingprice integer,
    buynow integer,
    startdate timestamp with time zone DEFAULT
transaction_timestamp() NOT NULL,
    limitdate timestamp with time zone NOT NULL,
    refusaldate timestamp with time zone,
    "/numberOfBids" integer,
    reasonofrefusal character varying,
    finaldate timestamp with time zone,
    finalprice integer,
    rate integer,
    auctioncreator integer NOT NULL,
    auctionwinner integer,
    responsiblemoderator integer,
    CONSTRAINT auction_buynow_check CHECK ((buynow >
0)),
    CONSTRAINT auction_check CHECK
```



```

((((minimumsellingprice > startingprice) OR
(minimumsellingprice = NULL::integer)) AND ((buynow >
startingprice) OR (buynow = NULL::integer))))),
    CONSTRAINT auction_minimumsellingprice_check
CHECK ((minimumsellingprice > 0)),
    CONSTRAINT auction_rate_check CHECK (((rate >= 0)
AND (rate <= 5))),
    CONSTRAINT auction_startingprice_check CHECK
((startingprice > 0)),
    CONSTRAINT check_auction_win CHECK
(check_auction_win(state, finaldate, finalprice,
auctionwinner)),
    CONSTRAINT check_dates CHECK ((age(limitdate,
startdate) <= '7 days'::interval)),
    CONSTRAINT check_rejected CHECK
(check_rejected_auction(state, refusaldate,
reasonofrefusal)),
    CONSTRAINT check_users CHECK
(check_auction_users(auctioncreator, auctionwinner,
responsiblemoderator))
);

```

```

ALTER TABLE auction OWNER TO lbaw1716;

```

```

--
-- Name: auto_increment_user; Type: SEQUENCE; Schema:
public; Owner: lbaw1716
--

```

```

CREATE SEQUENCE auto_increment_user
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

```

```

ALTER TABLE auto_increment_user OWNER TO lbaw1716;

```

```

--
-- Name: authenticated_user; Type: TABLE; Schema:
public; Owner: lbaw1716; Tablespace:
--

CREATE TABLE authenticated_user (
    id integer DEFAULT
nextval('auto_increment_user'::regclass) NOT NULL,
    typeofuser typeofuser NOT NULL,
    username character varying(50) NOT NULL,
    password character varying NOT NULL,
    pathtophoto character varying,
    completename character varying,
    email character varying,
    birthdate date,
    "/rating" integer,
    address character varying,
    postalcode character varying(50),
    balance integer,
    city integer,
    CONSTRAINT "authenticated_user_/rating_check"
CHECK ((("/rating" >= 0) AND ("/rating" <= 5))),
    CONSTRAINT authenticated_user_balance_check CHECK
((balance >= 0))
);

ALTER TABLE authenticated_user OWNER TO lbaw1716;

--
-- Name: auto_increment_category; Type: SEQUENCE;
Schema: public; Owner: lbaw1716
--

CREATE SEQUENCE auto_increment_category
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE

```

```
NO MAXVALUE  
CACHE 1;
```

```
ALTER TABLE auto_increment_category OWNER TO  
lbaw1716;
```

```
--  
-- Name: auto_increment_city; Type: SEQUENCE; Schema:  
public; Owner: lbaw1716  
--
```

```
CREATE SEQUENCE auto_increment_city  
    START WITH 4  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE auto_increment_city OWNER TO lbaw1716;
```

```
--  
-- Name: auto_increment_comment; Type: SEQUENCE;  
Schema: public; Owner: lbaw1716  
--
```

```
CREATE SEQUENCE auto_increment_comment  
    START WITH 1  
    INCREMENT BY 1  
    NO MINVALUE  
    NO MAXVALUE  
    CACHE 1;
```

```
ALTER TABLE auto_increment_comment OWNER TO lbaw1716;
```

```
--  
-- Name: auto_increment_notification; Type: SEQUENCE;
```

```

Schema: public; Owner: lbaw1716
--

CREATE SEQUENCE auto_increment_notification
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

ALTER TABLE auto_increment_notification OWNER TO
lbaw1716;

--
-- Name: bid; Type: TABLE; Schema: public; Owner:
lbaw1716; Tablespace:
--

CREATE TABLE bid (
    date timestamp with time zone DEFAULT
transaction_timestamp() NOT NULL,
    value integer NOT NULL,
    auctionbided integer NOT NULL,
    bidder integer NOT NULL,
    "isBuyNow" boolean
);

ALTER TABLE bid OWNER TO lbaw1716;

--
-- Name: blocks; Type: TABLE; Schema: public; Owner:
lbaw1716; Tablespace:
--

CREATE TABLE blocks (
    state blockingstate NOT NULL,
    description character varying,

```

```

        date timestamp with time zone DEFAULT
transaction_timestamp() NOT NULL,
        blocked integer NOT NULL,
        blocker integer NOT NULL,
        CONSTRAINT check_users CHECK
(check_block_users(blocked, blocker))
);

ALTER TABLE blocks OWNER TO lbaw1716;

--
-- Name: category; Type: TABLE; Schema: public;
Owner: lbaw1716; Tablespace:
--

CREATE TABLE category (
    categoryid integer DEFAULT
nextval('auto_increment_category'::regclass) NOT
NULL,
    name character varying(50) NOT NULL,
    parent integer,
    CONSTRAINT check_parent_not_equal CHECK ((parent
<> categoryid))
);

ALTER TABLE category OWNER TO lbaw1716;

--
-- Name: categoryofauction; Type: TABLE; Schema:
public; Owner: lbaw1716; Tablespace:
--

CREATE TABLE categoryofauction (
    category integer NOT NULL,
    auction integer NOT NULL
);

```

```

ALTER TABLE categoryofauction OWNER TO lbaw1716;

--
-- Name: city; Type: TABLE; Schema: public; Owner:
lbaw1716; Tablespace:
--

CREATE TABLE city (
    id integer DEFAULT
nextval('auto_increment_city'::regclass) NOT NULL,
    name character varying(50) NOT NULL,
    country integer NOT NULL
);

ALTER TABLE city OWNER TO lbaw1716;

--
-- Name: comment; Type: TABLE; Schema: public; Owner:
lbaw1716; Tablespace:
--

CREATE TABLE comment (
    id integer DEFAULT
nextval('auto_increment_comment'::regclass) NOT NULL,
    date timestamp with time zone DEFAULT
transaction_timestamp() NOT NULL,
    description character varying NOT NULL,
    auctioncommented integer NOT NULL,
    usercommenter integer NOT NULL
);

ALTER TABLE comment OWNER TO lbaw1716;

--
-- Name: country; Type: TABLE; Schema: public; Owner:
lbaw1716; Tablespace:

```

```
--

CREATE TABLE country (
    id integer DEFAULT
nextval('auto_increment_auction'::regclass) NOT NULL,
    name character varying(50) NOT NULL
);

ALTER TABLE country OWNER TO lbaw1716;

--
-- Name: edit_categories; Type: TABLE; Schema:
public; Owner: lbaw1716; Tablespace:
--

CREATE TABLE edit_categories (
    category integer NOT NULL,
    admin integer NOT NULL,
    CONSTRAINT check_admin_modify_category CHECK
(check_admin_modify_category(admin))
);

ALTER TABLE edit_categories OWNER TO lbaw1716;

--
-- Name: edit_moderator; Type: TABLE; Schema: public;
Owner: lbaw1716; Tablespace:
--

CREATE TABLE edit_moderator (
    removedmod integer NOT NULL,
    removeradmin integer NOT NULL,
    CONSTRAINT check_users CHECK
(check_edit_moderators(removedmod, removeradmin))
);
```

```

ALTER TABLE edit_moderator OWNER TO lbaw1716;

--
-- Name: lastbidder; Type: TABLE; Schema: public;
Owner: lbaw1716; Tablespace:
--

CREATE TABLE lastbidder (
    bidder integer
);

ALTER TABLE lastbidder OWNER TO lbaw1716;

--
-- Name: notification; Type: TABLE; Schema: public;
Owner: lbaw1716; Tablespace:
--

CREATE TABLE notification (
    id integer DEFAULT
nextval('auto_increment_notification'::regclass) NOT
NULL,
    date timestamp with time zone DEFAULT
transaction_timestamp() NOT NULL,
    description character varying(50) NOT NULL,
    type character varying(50) NOT NULL,
    auctionassociated integer,
    authenticated_userid integer NOT NULL
);

ALTER TABLE notification OWNER TO lbaw1716;

--
-- Name: report; Type: TABLE; Schema: public; Owner:
lbaw1716; Tablespace:
--

```



```

CREATE TABLE report (
    date timestamp with time zone DEFAULT
transaction_timestamp() NOT NULL,
    reason character varying NOT NULL,
    auctionid integer NOT NULL,
    normaluserid integer NOT NULL
);

ALTER TABLE report OWNER TO lbaw1716;

--
-- Name: add_credits_pkey; Type: CONSTRAINT; Schema:
public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY add_credits
    ADD CONSTRAINT add_credits_pkey PRIMARY KEY (id);

--
-- Name: auction_pkey; Type: CONSTRAINT; Schema:
public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY auction
    ADD CONSTRAINT auction_pkey PRIMARY KEY (id);

--
-- Name: authenticated_user_email_key; Type:
CONSTRAINT; Schema: public; Owner: lbaw1716;
Tablespace:
--

ALTER TABLE ONLY authenticated_user
    ADD CONSTRAINT authenticated_user_email_key
UNIQUE (email);

```

```

--
-- Name: authenticated_user_pkey; Type: CONSTRAINT;
Schema: public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY authenticated_user
    ADD CONSTRAINT authenticated_user_pkey PRIMARY
KEY (id);

--
-- Name: authenticated_user_username_key; Type:
CONSTRAINT; Schema: public; Owner: lbaw1716;
Tablespace:
--

ALTER TABLE ONLY authenticated_user
    ADD CONSTRAINT authenticated_user_username_key
UNIQUE (username);

--
-- Name: bid_pkey; Type: CONSTRAINT; Schema: public;
Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY bid
    ADD CONSTRAINT bid_pkey PRIMARY KEY (date,
auctionbided, bidder);

--
-- Name: blocks_pkey; Type: CONSTRAINT; Schema:
public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY blocks
    ADD CONSTRAINT blocks_pkey PRIMARY KEY (blocked,

```

```

blocker);

--
-- Name: category_name_key; Type: CONSTRAINT; Schema:
public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY category
    ADD CONSTRAINT category_name_key UNIQUE (name);

--
-- Name: category_pkey; Type: CONSTRAINT; Schema:
public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY category
    ADD CONSTRAINT category_pkey PRIMARY KEY
(categoryid);

--
-- Name: categoryofauction_pkey; Type: CONSTRAINT;
Schema: public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY categoryofauction
    ADD CONSTRAINT categoryofauction_pkey PRIMARY KEY
(category, auction);

--
-- Name: city_name_key; Type: CONSTRAINT; Schema:
public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY city
    ADD CONSTRAINT city_name_key UNIQUE (name);

```

```
--  
-- Name: city_pkey; Type: CONSTRAINT; Schema: public;  
Owner: lbaw1716; Tablespace:  
--
```

```
ALTER TABLE ONLY city  
    ADD CONSTRAINT city_pkey PRIMARY KEY (id);
```

```
--  
-- Name: comment_pkey; Type: CONSTRAINT; Schema:  
public; Owner: lbaw1716; Tablespace:  
--
```

```
ALTER TABLE ONLY comment  
    ADD CONSTRAINT comment_pkey PRIMARY KEY (id);
```

```
--  
-- Name: country_name_key; Type: CONSTRAINT; Schema:  
public; Owner: lbaw1716; Tablespace:  
--
```

```
ALTER TABLE ONLY country  
    ADD CONSTRAINT country_name_key UNIQUE (name);
```

```
--  
-- Name: country_pkey; Type: CONSTRAINT; Schema:  
public; Owner: lbaw1716; Tablespace:  
--
```

```
ALTER TABLE ONLY country  
    ADD CONSTRAINT country_pkey PRIMARY KEY (id);
```

```
--
```

```

-- Name: edit_categories_pkey; Type: CONSTRAINT;
Schema: public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY edit_categories
    ADD CONSTRAINT edit_categories_pkey PRIMARY KEY
(category, admin);

--

-- Name: edit_moderator_pkey; Type: CONSTRAINT;
Schema: public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY edit_moderator
    ADD CONSTRAINT edit_moderator_pkey PRIMARY KEY
(removedmod, removeradmin);

--

-- Name: notification_pkey; Type: CONSTRAINT; Schema:
public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY notification
    ADD CONSTRAINT notification_pkey PRIMARY KEY
(id);

--

-- Name: report_pkey; Type: CONSTRAINT; Schema:
public; Owner: lbaw1716; Tablespace:
--

ALTER TABLE ONLY report
    ADD CONSTRAINT report_pkey PRIMARY KEY
(auctionid, normaluserid);

```

```

--
-- Name: auction_creator; Type: TRIGGER; Schema:
public; Owner: lbaw1716
--

CREATE TRIGGER auction_creator BEFORE INSERT ON bid
FOR EACH ROW EXECUTE PROCEDURE auction_creator();

--
-- Name: bid_greater_than_last; Type: TRIGGER;
Schema: public; Owner: lbaw1716
--

CREATE TRIGGER bid_greater_than_last BEFORE INSERT ON
bid FOR EACH ROW EXECUTE PROCEDURE
bid_greater_than_last();

--
-- Name: bidder_has_money; Type: TRIGGER; Schema:
public; Owner: lbaw1716
--

CREATE TRIGGER bidder_has_money BEFORE INSERT ON bid
FOR EACH ROW EXECUTE PROCEDURE bidder_has_money();

--
-- Name: buy_now; Type: TRIGGER; Schema: public;
Owner: lbaw1716
--

CREATE TRIGGER buy_now BEFORE INSERT ON bid FOR EACH
ROW EXECUTE PROCEDURE buy_now();

--
-- Name: check_bid_value; Type: TRIGGER; Schema:

```

```

public; Owner: lbaw1716
--

CREATE TRIGGER check_bid_value BEFORE INSERT ON bid
FOR EACH ROW EXECUTE PROCEDURE check_bid_value();

--

-- Name: update_rating; Type: TRIGGER; Schema:
public; Owner: lbaw1716
--

CREATE TRIGGER update_rating BEFORE INSERT OR UPDATE
ON auction FOR EACH ROW EXECUTE PROCEDURE
update_ratings();

--

-- Name: win_auction; Type: TRIGGER; Schema: public;
Owner: lbaw1716
--

CREATE TRIGGER win_auction BEFORE UPDATE ON bid FOR
EACH ROW EXECUTE PROCEDURE win_auction();

--

-- Name: add_credits_user_fkey; Type: FK CONSTRAINT;
Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY add_credits
    ADD CONSTRAINT add_credits_user_fkey FOREIGN KEY
("user") REFERENCES authenticated_user(id);

--

-- Name: auction_auctioncreator_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716

```

```

--

ALTER TABLE ONLY auction
    ADD CONSTRAINT auction_auctioncreator_fkey
FOREIGN KEY (auctioncreator) REFERENCES
authenticated_user(id);

--

-- Name: auction_auctionwinner_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY auction
    ADD CONSTRAINT auction_auctionwinner_fkey FOREIGN
KEY (auctionwinner) REFERENCES
authenticated_user(id);

--

-- Name: auction_responsiblemoderator_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY auction
    ADD CONSTRAINT auction_responsiblemoderator_fkey
FOREIGN KEY (responsiblemoderator) REFERENCES
authenticated_user(id);

--

-- Name: authenticated_user_city_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY authenticated_user
    ADD CONSTRAINT authenticated_user_city_fkey
FOREIGN KEY (city) REFERENCES city(id);

```



```
--  
-- Name: bid_auctionbidded_fkey; Type: FK CONSTRAINT;  
Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY bid  
    ADD CONSTRAINT bid_auctionbidded_fkey FOREIGN KEY  
(auctionbidded) REFERENCES auction(id);
```

```
--  
-- Name: bid_bidder_fkey; Type: FK CONSTRAINT;  
Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY bid  
    ADD CONSTRAINT bid_bidder_fkey FOREIGN KEY  
(bidder) REFERENCES authenticated_user(id);
```

```
--  
-- Name: blocks_blocked_fkey; Type: FK CONSTRAINT;  
Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY blocks  
    ADD CONSTRAINT blocks_blocked_fkey FOREIGN KEY  
(blocked) REFERENCES authenticated_user(id);
```

```
--  
-- Name: blocks_blocker_fkey; Type: FK CONSTRAINT;  
Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY blocks  
    ADD CONSTRAINT blocks_blocker_fkey FOREIGN KEY  
(blocker) REFERENCES authenticated_user(id);
```

```
--  
-- Name: category_parent_fkey; Type: FK CONSTRAINT;  
Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY category  
    ADD CONSTRAINT category_parent_fkey FOREIGN KEY  
(parent) REFERENCES category(categoryid);
```

```
--  
-- Name: categoryofauction_auction_fkey; Type: FK  
CONSTRAINT; Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY categoryofauction  
    ADD CONSTRAINT categoryofauction_auction_fkey  
FOREIGN KEY (auction) REFERENCES auction(id);
```

```
--  
-- Name: categoryofauction_category_fkey; Type: FK  
CONSTRAINT; Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY categoryofauction  
    ADD CONSTRAINT categoryofauction_category_fkey  
FOREIGN KEY (category) REFERENCES  
category(categoryid);
```

```
--  
-- Name: city_country_fkey; Type: FK CONSTRAINT;  
Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY city
```

```

        ADD CONSTRAINT city_country_fkey FOREIGN KEY
(country) REFERENCES country(id);

--
-- Name: comment_auctioncommented_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY comment
    ADD CONSTRAINT comment_auctioncommented_fkey
FOREIGN KEY (auctioncommented) REFERENCES
auction(id);

--
-- Name: comment_usercommenter_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY comment
    ADD CONSTRAINT comment_usercommenter_fkey FOREIGN
KEY (usercommenter) REFERENCES
authenticated_user(id);

--
-- Name: edit_categories_admin_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY edit_categories
    ADD CONSTRAINT edit_categories_admin_fkey FOREIGN
KEY (admin) REFERENCES authenticated_user(id);

--
-- Name: edit_categories_category_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716

```

```

--

ALTER TABLE ONLY edit_categories
    ADD CONSTRAINT edit_categories_category_fkey
FOREIGN KEY (category) REFERENCES
category(categoryid);

--

-- Name: edit_moderator_removedmod_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY edit_moderator
    ADD CONSTRAINT edit_moderator_removedmod_fkey
FOREIGN KEY (removedmod) REFERENCES
authenticated_user(id);

--

-- Name: edit_moderator_removaladmin_fkey; Type: FK
CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY edit_moderator
    ADD CONSTRAINT edit_moderator_removaladmin_fkey
FOREIGN KEY (removaladmin) REFERENCES
authenticated_user(id);

--

-- Name: notification_auctionassociated_fkey; Type:
FK CONSTRAINT; Schema: public; Owner: lbaw1716
--

ALTER TABLE ONLY notification
    ADD CONSTRAINT
notification_auctionassociated_fkey FOREIGN KEY
(auctionassociated) REFERENCES auction(id);

```

```
--  
-- Name: notification_authenticated_userid_fkey;  
Type: FK CONSTRAINT; Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY notification  
    ADD CONSTRAINT  
notification_authenticated_userid_fkey FOREIGN KEY  
(authenticated_userid) REFERENCES  
authenticated_user(id);
```

```
--  
-- Name: report_auctionid_fkey; Type: FK CONSTRAINT;  
Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY report  
    ADD CONSTRAINT report_auctionid_fkey FOREIGN KEY  
(auctionid) REFERENCES auction(id);
```

```
--  
-- Name: report_normaluserid_fkey; Type: FK  
CONSTRAINT; Schema: public; Owner: lbaw1716  
--
```

```
ALTER TABLE ONLY report  
    ADD CONSTRAINT report_normaluserid_fkey FOREIGN  
KEY (normaluserid) REFERENCES authenticated_user(id);
```

```
--  
-- Name: public; Type: ACL; Schema: -; Owner:  
lbaw1716  
--
```

```
REVOKE ALL ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON SCHEMA public FROM lbaw1716;
GRANT ALL ON SCHEMA public TO lbaw1716;
```

Data File

```
--
-- PostgreSQL data (inserts)
--

SET statement_timeout = 0;
SET lock_timeout = 0;
SET client_encoding = 'UTF8';
SET standard_conforming_strings = on;
SET check_function_bodies = false;
SET client_min_messages = warning;

SET search_path = public, pg_catalog;

--
-- Data for Name: country; Type: TABLE DATA; Schema:
public; Owner: lbaw1716
--

INSERT INTO country VALUES (1, 'Afghanistan');
INSERT INTO country VALUES (2, 'Albania');
INSERT INTO country VALUES (3, 'Algeria');
INSERT INTO country VALUES (4, 'American Samoa');
INSERT INTO country VALUES (5, 'Andorra');
INSERT INTO country VALUES (6, 'Angola');
INSERT INTO country VALUES (7, 'Anguilla');
INSERT INTO country VALUES (8, 'Antarctica');
INSERT INTO country VALUES (9, 'Antigua and
Barbuda');
INSERT INTO country VALUES (10, 'Argentina');
INSERT INTO country VALUES (11, 'Armenia');
INSERT INTO country VALUES (12, 'Aruba');
```

```

INSERT INTO country VALUES (13, 'Australia');
INSERT INTO country VALUES (14, 'Austria');
INSERT INTO country VALUES (15, 'Azerbaijan');
INSERT INTO country VALUES (16, 'Bahamas');
INSERT INTO country VALUES (17, 'Bahrain');
INSERT INTO country VALUES (18, 'Bangladesh');
INSERT INTO country VALUES (19, 'Barbados');
INSERT INTO country VALUES (20, 'Belarus');
INSERT INTO country VALUES (21, 'Belgium');
INSERT INTO country VALUES (22, 'Belize');
INSERT INTO country VALUES (23, 'Benin');
INSERT INTO country VALUES (24, 'Bermuda');
INSERT INTO country VALUES (25, 'Bhutan');
INSERT INTO country VALUES (26, 'Bolivia');
INSERT INTO country VALUES (27, 'Bosnia and
Herzegovina');
INSERT INTO country VALUES (28, 'Botswana');
INSERT INTO country VALUES (29, 'Bouvet Island');
INSERT INTO country VALUES (30, 'Brazil');
INSERT INTO country VALUES (31, 'British Indian Ocean
Territory');
INSERT INTO country VALUES (32, 'Brunei Darussalam');
INSERT INTO country VALUES (33, 'Bulgaria');
INSERT INTO country VALUES (34, 'Burkina Faso');
INSERT INTO country VALUES (35, 'Burundi');
INSERT INTO country VALUES (36, 'Cambodia');
INSERT INTO country VALUES (37, 'Cameroon');
INSERT INTO country VALUES (38, 'Canada');
INSERT INTO country VALUES (39, 'Cape Verde');
INSERT INTO country VALUES (40, 'Cayman Islands');
INSERT INTO country VALUES (41, 'Central African
Republic');
INSERT INTO country VALUES (42, 'Chad');
INSERT INTO country VALUES (43, 'Chile');
INSERT INTO country VALUES (44, 'China');
INSERT INTO country VALUES (45, 'Christmas Island');
INSERT INTO country VALUES (46, 'Cocos (Keeling)
Islands');
INSERT INTO country VALUES (47, 'Colombia');

```

```

INSERT INTO country VALUES (48, 'Comoros');
INSERT INTO country VALUES (49, 'Congo');
INSERT INTO country VALUES (50, 'Cook Islands');
INSERT INTO country VALUES (51, 'Costa Rica');
INSERT INTO country VALUES (52, 'Croatia
(Hrvatska)');
INSERT INTO country VALUES (53, 'Cuba');
INSERT INTO country VALUES (54, 'Cyprus');
INSERT INTO country VALUES (55, 'Czech Republic');
INSERT INTO country VALUES (56, 'Denmark');
INSERT INTO country VALUES (57, 'Djibouti');
INSERT INTO country VALUES (58, 'Dominica');
INSERT INTO country VALUES (59, 'Dominican
Republic');
INSERT INTO country VALUES (60, 'East Timor');
INSERT INTO country VALUES (61, 'Ecuador');
INSERT INTO country VALUES (62, 'Egypt');
INSERT INTO country VALUES (63, 'El Salvador');
INSERT INTO country VALUES (64, 'Equatorial Guinea');
INSERT INTO country VALUES (65, 'Eritrea');
INSERT INTO country VALUES (66, 'Estonia');
INSERT INTO country VALUES (67, 'Ethiopia');
INSERT INTO country VALUES (68, 'Falkland Islands
(Malvinas)');
INSERT INTO country VALUES (69, 'Faroe Islands');
INSERT INTO country VALUES (70, 'Fiji');
INSERT INTO country VALUES (71, 'Finland');
INSERT INTO country VALUES (72, 'France');
INSERT INTO country VALUES (73, 'France,
Metropolitan');
INSERT INTO country VALUES (74, 'French Guiana');
INSERT INTO country VALUES (75, 'French Polynesia');
INSERT INTO country VALUES (76, 'French Southern
Territories');
INSERT INTO country VALUES (77, 'Gabon');
INSERT INTO country VALUES (78, 'Gambia');
INSERT INTO country VALUES (79, 'Georgia');
INSERT INTO country VALUES (80, 'Germany');
INSERT INTO country VALUES (81, 'Ghana');

```



```

INSERT INTO country VALUES (82, 'Gibraltar');
INSERT INTO country VALUES (83, 'Guernsey');
INSERT INTO country VALUES (84, 'Greece');
INSERT INTO country VALUES (85, 'Greenland');
INSERT INTO country VALUES (86, 'Grenada');
INSERT INTO country VALUES (87, 'Guadeloupe');
INSERT INTO country VALUES (88, 'Guam');
INSERT INTO country VALUES (89, 'Guatemala');
INSERT INTO country VALUES (90, 'Guinea');
INSERT INTO country VALUES (91, 'Guinea-Bissau');
INSERT INTO country VALUES (92, 'Guyana');
INSERT INTO country VALUES (93, 'Haiti');
INSERT INTO country VALUES (94, 'Heard and Mc Donald
Islands');
INSERT INTO country VALUES (95, 'Honduras');
INSERT INTO country VALUES (96, 'Hong Kong');
INSERT INTO country VALUES (97, 'Hungary');
INSERT INTO country VALUES (98, 'Iceland');
INSERT INTO country VALUES (99, 'India');
INSERT INTO country VALUES (100, 'Isle of Man');
INSERT INTO country VALUES (101, 'Indonesia');
INSERT INTO country VALUES (102, 'Iran (Islamic
Republic of)');
INSERT INTO country VALUES (103, 'Iraq');
INSERT INTO country VALUES (104, 'Ireland');
INSERT INTO country VALUES (105, 'Israel');
INSERT INTO country VALUES (106, 'Italy');
INSERT INTO country VALUES (107, 'Ivory Coast');
INSERT INTO country VALUES (108, 'Jersey');
INSERT INTO country VALUES (109, 'Jamaica');
INSERT INTO country VALUES (110, 'Japan');
INSERT INTO country VALUES (111, 'Jordan');
INSERT INTO country VALUES (112, 'Kazakhstan');
INSERT INTO country VALUES (113, 'Kenya');
INSERT INTO country VALUES (114, 'Kiribati');
INSERT INTO country VALUES (117, 'Kosovo');
INSERT INTO country VALUES (118, 'Kuwait');
INSERT INTO country VALUES (119, 'Kyrgyzstan');
INSERT INTO country VALUES (120, 'Lao People''''s

```

```

Democratic Republic');
INSERT INTO country VALUES (121, 'Latvia');
INSERT INTO country VALUES (122, 'Lebanon');
INSERT INTO country VALUES (123, 'Lesotho');
INSERT INTO country VALUES (124, 'Liberia');
INSERT INTO country VALUES (125, 'Libyan Arab
Jamahiriya');
INSERT INTO country VALUES (126, 'Liechtenstein');
INSERT INTO country VALUES (127, 'Lithuania');
INSERT INTO country VALUES (128, 'Luxembourg');
INSERT INTO country VALUES (129, 'Macau');
INSERT INTO country VALUES (130, 'Macedonia');
INSERT INTO country VALUES (131, 'Madagascar');
INSERT INTO country VALUES (132, 'Malawi');
INSERT INTO country VALUES (133, 'Malaysia');
INSERT INTO country VALUES (134, 'Maldives');
INSERT INTO country VALUES (135, 'Mali');
INSERT INTO country VALUES (136, 'Malta');
INSERT INTO country VALUES (137, 'Marshall Islands');
INSERT INTO country VALUES (138, 'Martinique');
INSERT INTO country VALUES (139, 'Mauritania');
INSERT INTO country VALUES (140, 'Mauritius');
INSERT INTO country VALUES (141, 'Mayotte');
INSERT INTO country VALUES (142, 'Mexico');
INSERT INTO country VALUES (143, 'Micronesia,
Federated States of');
INSERT INTO country VALUES (144, 'Moldova, Republic
of');
INSERT INTO country VALUES (145, 'Monaco');
INSERT INTO country VALUES (146, 'Mongolia');
INSERT INTO country VALUES (147, 'Montenegro');
INSERT INTO country VALUES (148, 'Montserrat');
INSERT INTO country VALUES (149, 'Morocco');
INSERT INTO country VALUES (150, 'Mozambique');
INSERT INTO country VALUES (151, 'Myanmar');
INSERT INTO country VALUES (152, 'Namibia');
INSERT INTO country VALUES (153, 'Nauru');
INSERT INTO country VALUES (154, 'Nepal');
INSERT INTO country VALUES (155, 'Netherlands');

```

```

INSERT INTO country VALUES (156, 'Netherlands
Antilles');
INSERT INTO country VALUES (157, 'New Caledonia');
INSERT INTO country VALUES (158, 'New Zealand');
INSERT INTO country VALUES (159, 'Nicaragua');
INSERT INTO country VALUES (160, 'Niger');
INSERT INTO country VALUES (161, 'Nigeria');
INSERT INTO country VALUES (162, 'Niue');
INSERT INTO country VALUES (163, 'Norfolk Island');
INSERT INTO country VALUES (164, 'Northern Mariana
Islands');
INSERT INTO country VALUES (165, 'Norway');
INSERT INTO country VALUES (166, 'Oman');
INSERT INTO country VALUES (167, 'Pakistan');
INSERT INTO country VALUES (168, 'Palau');
INSERT INTO country VALUES (169, 'Palestine');
INSERT INTO country VALUES (170, 'Panama');
INSERT INTO country VALUES (171, 'Papua New Guinea');
INSERT INTO country VALUES (172, 'Paraguay');
INSERT INTO country VALUES (173, 'Peru');
INSERT INTO country VALUES (174, 'Philippines');
INSERT INTO country VALUES (116, 'South Korea');
INSERT INTO country VALUES (175, 'Pitcairn');
INSERT INTO country VALUES (176, 'Poland');
INSERT INTO country VALUES (177, 'Portugal');
INSERT INTO country VALUES (178, 'Puerto Rico');
INSERT INTO country VALUES (179, 'Qatar');
INSERT INTO country VALUES (180, 'Reunion');
INSERT INTO country VALUES (181, 'Romania');
INSERT INTO country VALUES (182, 'Russian
Federation');
INSERT INTO country VALUES (183, 'Rwanda');
INSERT INTO country VALUES (184, 'Saint Kitts and
Nevis');
INSERT INTO country VALUES (185, 'Saint Lucia');
INSERT INTO country VALUES (186, 'Saint Vincent and
the Grenadines');
INSERT INTO country VALUES (187, 'Samoa');
INSERT INTO country VALUES (188, 'San Marino');

```

```

INSERT INTO country VALUES (189, 'Sao Tome and
Principe');
INSERT INTO country VALUES (190, 'Saudi Arabia');
INSERT INTO country VALUES (191, 'Senegal');
INSERT INTO country VALUES (192, 'Serbia');
INSERT INTO country VALUES (193, 'Seychelles');
INSERT INTO country VALUES (194, 'Sierra Leone');
INSERT INTO country VALUES (195, 'Singapore');
INSERT INTO country VALUES (196, 'Slovakia');
INSERT INTO country VALUES (197, 'Slovenia');
INSERT INTO country VALUES (198, 'Solomon Islands');
INSERT INTO country VALUES (199, 'Somalia');
INSERT INTO country VALUES (200, 'South Africa');
INSERT INTO country VALUES (201, 'South Georgia South
Sandwich Islands');
INSERT INTO country VALUES (202, 'Spain');
INSERT INTO country VALUES (203, 'Sri Lanka');
INSERT INTO country VALUES (204, 'St. Helena');
INSERT INTO country VALUES (205, 'St. Pierre and
Miquelon');
INSERT INTO country VALUES (206, 'Sudan');
INSERT INTO country VALUES (207, 'Suriname');
INSERT INTO country VALUES (208, 'Svalbard and Jan
Mayen Islands');
INSERT INTO country VALUES (209, 'Swaziland');
INSERT INTO country VALUES (210, 'Sweden');
INSERT INTO country VALUES (211, 'Switzerland');
INSERT INTO country VALUES (212, 'Syrian Arab
Republic');
INSERT INTO country VALUES (213, 'Taiwan');
INSERT INTO country VALUES (214, 'Tajikistan');
INSERT INTO country VALUES (215, 'Tanzania, United
Republic of');
INSERT INTO country VALUES (216, 'Thailand');
INSERT INTO country VALUES (217, 'Togo');
INSERT INTO country VALUES (218, 'Tokelau');
INSERT INTO country VALUES (219, 'Tonga');
INSERT INTO country VALUES (220, 'Trinidad and
Tobago');

```

```

INSERT INTO country VALUES (221, 'Tunisia');
INSERT INTO country VALUES (222, 'Turkey');
INSERT INTO country VALUES (223, 'Turkmenistan');
INSERT INTO country VALUES (224, 'Turks and Caicos
Islands');
INSERT INTO country VALUES (225, 'Tuvalu');
INSERT INTO country VALUES (226, 'Uganda');
INSERT INTO country VALUES (227, 'Ukraine');
INSERT INTO country VALUES (228, 'United Arab
Emirates');
INSERT INTO country VALUES (229, 'United Kingdom');
INSERT INTO country VALUES (230, 'United States');
INSERT INTO country VALUES (231, 'United States minor
outlying islands');
INSERT INTO country VALUES (232, 'Uruguay');
INSERT INTO country VALUES (233, 'Uzbekistan');
INSERT INTO country VALUES (234, 'Vanuatu');
INSERT INTO country VALUES (235, 'Vatican City
State');
INSERT INTO country VALUES (236, 'Venezuela');
INSERT INTO country VALUES (237, 'Vietnam');
INSERT INTO country VALUES (238, 'Virgin Islands
(British)');
INSERT INTO country VALUES (239, 'Virgin Islands
(U.S.)');
INSERT INTO country VALUES (240, 'Wallis and Futuna
Islands');
INSERT INTO country VALUES (241, 'Western Sahara');
INSERT INTO country VALUES (242, 'Yemen');
INSERT INTO country VALUES (243, 'Zaire');
INSERT INTO country VALUES (244, 'Zambia');
INSERT INTO country VALUES (245, 'Zimbabwe');
INSERT INTO country VALUES (115, 'North Korea');

```

```
--
```

```
-- Data for Name: city; Type: TABLE DATA; Schema:
public; Owner: lbaw1716
```

```
--
```

```
INSERT INTO city VALUES (1, 'Dongguan', 44);
INSERT INTO city VALUES (2, 'Hanoi', 237);
INSERT INTO city VALUES (3, 'Ankara', 222);
INSERT INTO city VALUES (4, 'Suzhou', 44);
INSERT INTO city VALUES (5, 'Yokohama', 110);
INSERT INTO city VALUES (6, 'Moscow', 182);
INSERT INTO city VALUES (7, 'New York City', 230);
INSERT INTO city VALUES (8, 'Cairo', 62);
INSERT INTO city VALUES (9, 'Beijing', 44);
INSERT INTO city VALUES (10, 'Berlin', 80);
INSERT INTO city VALUES (11, 'Lima', 173);
INSERT INTO city VALUES (12, 'Johannesburg', 200);
INSERT INTO city VALUES (13, 'Saint Petersburg',
182);
INSERT INTO city VALUES (14, 'Los Angeles', 230);
INSERT INTO city VALUES (15, 'Singapore', 195);
INSERT INTO city VALUES (17, 'Istanbul', 222);
INSERT INTO city VALUES (18, 'Madrid', 202);
INSERT INTO city VALUES (19, 'London', 229);
INSERT INTO city VALUES (20, 'Lagos', 161);
INSERT INTO city VALUES (21, 'Kolkata', 99);
INSERT INTO city VALUES (22, 'Jaipur', 99);
INSERT INTO city VALUES (23, 'Mumbai', 99);
INSERT INTO city VALUES (24, 'Busan', 116);
INSERT INTO city VALUES (25, 'Pyongyang', 115);
INSERT INTO city VALUES (26, 'Seoul', 116);
INSERT INTO city VALUES (27, 'Sao Paulo', 30);
INSERT INTO city VALUES (28, 'Rio de Janeiro', 30);
INSERT INTO city VALUES (29, 'Salvador', 30);
INSERT INTO city VALUES (30, 'Porto Alegre', 30);
INSERT INTO city VALUES (31, 'Natal', 30);
INSERT INTO city VALUES (32, 'Joinville', 30);
INSERT INTO city VALUES (33, 'Lisbon', 177);
INSERT INTO city VALUES (34, 'Porto', 177);
INSERT INTO city VALUES (35, 'Casablanca', 149);
INSERT INTO city VALUES (36, 'Cape Town', 200);
INSERT INTO city VALUES (37, 'Hyderabad', 99);
INSERT INTO city VALUES (38, 'Jakarta', 101);
```



```

INSERT INTO city VALUES (39, 'Delhi', 99);
INSERT INTO city VALUES (40, 'Frankfurt', 80);
INSERT INTO city VALUES (41, 'Bucharest', 181);
INSERT INTO city VALUES (42, 'Kiev', 227);
INSERT INTO city VALUES (43, 'Rome', 106);
INSERT INTO city VALUES (44, 'Paris', 72);
INSERT INTO city VALUES (45, 'Minsk', 20);
INSERT INTO city VALUES (46, 'Vienna', 14);
INSERT INTO city VALUES (47, 'Budapest', 97);
INSERT INTO city VALUES (48, 'Hamburg', 80);
INSERT INTO city VALUES (49, 'Warsaw', 176);
INSERT INTO city VALUES (50, 'Munich', 80);
INSERT INTO city VALUES (51, 'Kharkiv', 227);
INSERT INTO city VALUES (52, 'Prague', 55);
INSERT INTO city VALUES (53, 'Sofia', 33);
INSERT INTO city VALUES (54, 'Nizhny Novgorod', 182);
INSERT INTO city VALUES (55, 'Kazan', 182);
INSERT INTO city VALUES (56, 'Brussels', 21);
INSERT INTO city VALUES (57, 'Columbus ', 230);
INSERT INTO city VALUES (58, 'Boston ', 230);
INSERT INTO city VALUES (59, 'Bakersfield ', 230);
INSERT INTO city VALUES (60, 'Tucson ', 230);
INSERT INTO city VALUES (61, 'Fort Worth ', 230);
INSERT INTO city VALUES (62, 'Lexington-Fayette ',
230);
INSERT INTO city VALUES (63, 'Arlington ', 230);
INSERT INTO city VALUES (64, 'Montgomery ', 230);
INSERT INTO city VALUES (65, 'Greensboro ', 230);
INSERT INTO city VALUES (66, 'Minneapolis ', 230);
INSERT INTO city VALUES (67, 'Virginia Beach ', 230);
INSERT INTO city VALUES (68, 'Sacramento ', 230);
INSERT INTO city VALUES (69, 'Honolulu ', 230);
INSERT INTO city VALUES (70, 'Norfolk ', 230);
INSERT INTO city VALUES (71, 'Cleveland ', 230);
INSERT INTO city VALUES (72, 'Washington D.C', 230);
INSERT INTO city VALUES (73, 'Chicago ', 230);
INSERT INTO city VALUES (74, 'Birmingham', 229);
INSERT INTO city VALUES (75, 'Bath', 229);
INSERT INTO city VALUES (76, 'Belfast', 229);

```

```

INSERT INTO city VALUES (77, 'Bristol', 229);
INSERT INTO city VALUES (78, 'Cambridge', 229);
INSERT INTO city VALUES (79, 'Canterbury', 229);
INSERT INTO city VALUES (80, 'Chester', 229);
INSERT INTO city VALUES (81, 'Derby', 229);
INSERT INTO city VALUES (82, 'Edinburgh', 229);
INSERT INTO city VALUES (83, 'Glasgow', 229);
INSERT INTO city VALUES (84, 'Hereford', 229);
INSERT INTO city VALUES (85, 'Lancaster', 229);
INSERT INTO city VALUES (86, 'Leicester', 229);
INSERT INTO city VALUES (87, 'Liverpool', 229);
INSERT INTO city VALUES (88, 'Manchester', 229);
INSERT INTO city VALUES (89, 'Oxford', 229);
INSERT INTO city VALUES (90, 'Nottingham', 229);
INSERT INTO city VALUES (91, 'Peterborough', 229);
INSERT INTO city VALUES (92, 'Plymouth', 229);
INSERT INTO city VALUES (93, 'Portsmouth', 229);
INSERT INTO city VALUES (94, 'Winchester', 229);
INSERT INTO city VALUES (95, 'Westminster', 229);
INSERT INTO city VALUES (96, 'York', 229);
INSERT INTO city VALUES (97, 'Stockholm', 210);
INSERT INTO city VALUES (98, 'Gothenburg', 210);
INSERT INTO city VALUES (99, 'Malmö', 210);
INSERT INTO city VALUES (100, 'Uppsala', 210);
INSERT INTO city VALUES (101, 'Västerås', 210);
INSERT INTO city VALUES (102, 'Örebro', 210);
INSERT INTO city VALUES (103, 'Lund', 210);
INSERT INTO city VALUES (104, 'Helsingborg', 210);
INSERT INTO city VALUES (105, 'Umeå', 210);
INSERT INTO city VALUES (107, 'Milan', 106);
INSERT INTO city VALUES (108, 'Turin', 106);
INSERT INTO city VALUES (109, 'Palermo', 106);
INSERT INTO city VALUES (110, 'Bologna', 106);
INSERT INTO city VALUES (111, 'Venice', 106);
INSERT INTO city VALUES (112, 'Cagliari', 106);
INSERT INTO city VALUES (113, 'Oslo', 165);
INSERT INTO city VALUES (114, 'Bergen', 165);
INSERT INTO city VALUES (115, 'Trondheim', 165);

```



```

INSERT INTO city VALUES (116, 'Stavanger', 165);
INSERT INTO city VALUES (117, 'Sandnes', 165);
INSERT INTO city VALUES (118, 'Fredrikstad', 165);
INSERT INTO city VALUES (119, 'Ålesund', 165);
INSERT INTO city VALUES (120, 'Maastricht', 155);
INSERT INTO city VALUES (121, 'Kerkrade', 155);
INSERT INTO city VALUES (122, 'Nieuwstadt', 155);
INSERT INTO city VALUES (123, 'Helmond', 155);
INSERT INTO city VALUES (124, 'Roosendaal', 155);
INSERT INTO city VALUES (125, 'Monnickendam', 155);
INSERT INTO city VALUES (126, 'Amsterdam', 155);
INSERT INTO city VALUES (127, 'Edam', 155);
INSERT INTO city VALUES (128, 'Enschede', 155);
INSERT INTO city VALUES (129, 'Utrecht', 155);
INSERT INTO city VALUES (130, 'Sluis', 155);
INSERT INTO city VALUES (131, 'Rotterdam', 155);
INSERT INTO city VALUES (132, 'The Hague', 155);
INSERT INTO city VALUES (133, 'Copenhagen', 56);
INSERT INTO city VALUES (134, 'Bangkok', 216);
INSERT INTO city VALUES (135, 'Abidjan', 107);
INSERT INTO city VALUES (136, 'Durban', 200);
INSERT INTO city VALUES (137, 'Alexandria', 62);
INSERT INTO city VALUES (138, 'Santiago', 43);
INSERT INTO city VALUES (139, 'Riyadh', 190);
INSERT INTO city VALUES (140, 'Tianjin', 44);
INSERT INTO city VALUES (142, 'Shenzhen', 44);
INSERT INTO city VALUES (141, 'Shanghai', 44);
INSERT INTO city VALUES (143, 'Pune', 99);
INSERT INTO city VALUES (144, 'Kinshasa', 49);
INSERT INTO city VALUES (145, 'Karachi', 167);
INSERT INTO city VALUES (146, 'Dhaka', 18);
INSERT INTO city VALUES (147, 'Addis Ababa', 67);
INSERT INTO city VALUES (148, 'Malanje', 6);
INSERT INTO city VALUES (149, 'Luanda ', 6);
INSERT INTO city VALUES (150, 'Lubango', 6);

```

--

-- Data for Name: authenticated_user; Type: TABLE

DATA; Schema: public; Owner: lbaw1716

--

```
INSERT INTO authenticated_user VALUES (5, 'Normal',
'erika_wilk1978',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Mia S Burke',
'purus.Maecenas@ipsumdolorsit.net', '1996-11-03',
NULL, '73 York Road', 'OX1 4PP', 9140, 89);
INSERT INTO authenticated_user VALUES (1,
'Administrator', 'NadiaCarvalho',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Nádia Carvalho', NULL, NULL, NULL,
NULL, NULL, 9800, NULL);
INSERT INTO authenticated_user VALUES (3, 'Normal',
'DavisonNaomi',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Naomi L Davison',
'non.lobortis@ultricesposuere.org', '1975-05-17',
NULL, '90 Cunnery Rd', 'M2 1HE', 9800, 88);
INSERT INTO authenticated_user VALUES (4, 'Normal',
'langeles',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Adam M Shah',
'quis.accumsan.convallis@pellentesquetellus.org',
'1976-03-16', NULL, '62 Cunnery Rd', 'M60 9DW', 9800,
88);
INSERT INTO authenticated_user VALUES (7, 'Normal',
'fearfultruck',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Naomi L Davison',
'0manan.bhalla.mbz@myfaceb00k.ml', '1975-05-17',
NULL, '90 Cunnery Rd', 'M2 1HE', 9800, 88);
INSERT INTO authenticated_user VALUES (25, 'Normal',
'frillymammoth',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Mia S Burke',
'dictum@ridiculusmus.co.uk', '1996-11-03', NULL, '73
York Road', 'OX1 4PP', 9800, 89);
```

```

INSERT INTO authenticated_user VALUES (6, 'Normal',
'assetsopengl',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', 'http://4.bp.blogspot.com/-SoLmj_KD-
oE/VCbiHfUzMjI/AAAAAAAAD-A/mLZ-3j-
TAvk/s1600/Ian%2BH%2B(7).jpg', 'Max H Harding',
'Enswearry45@dayrep.com', '1981-05-28', NULL, '82
Guild Street', 'NW6 3LF', 9800, 19);
INSERT INTO authenticated_user VALUES (9, 'Normal',
'switchsparkling',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Mia S Burke',
'a.tortor@estNunc.co.uk', '1996-11-03', NULL, '73
York Road', 'OX1 4PP', 9800, 89);
INSERT INTO authenticated_user VALUES (12, 'Normal',
'nippykrypton',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Adam M Shah',
'Integer.aliquam.adipiscing@non.org', '1976-03-16',
NULL, '62 Cunnery Rd', 'M60 9DW', 9800, 88);
INSERT INTO authenticated_user VALUES (13, 'Normal',
'informdunghill',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Mia S Burke',
'lorem.sit@Namnulla.org', '1996-11-03', NULL, '73
York Road', 'OX1 4PP', 9800, 89);
INSERT INTO authenticated_user VALUES (14, 'Normal',
'grossboars',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Naomi L Davison',
'sociis.natoque@iaculisaliquetdiam.co.uk', '1975-05-
17', NULL, '90 Cunnery Rd', 'M2 1HE', 9800, 88);
INSERT INTO authenticated_user VALUES (15, 'Normal',
'rubberspinner',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Adam M Shah', 'velit@orci.co.uk',
'1976-03-16', NULL, '62 Cunnery Rd', 'M60 9DW', 9800,
88);
INSERT INTO authenticated_user VALUES (16, 'Normal',

```

```

'branchgreat',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Mia S Burke',
'leo.in.lobortis@litoratorquentper.net', '1996-11-
03', NULL, '73 York Road', 'OX1 4PP', 9800, 89);
INSERT INTO authenticated_user VALUES (17, 'Normal',
'evilpublic',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Naomi L Davison',
'nisl.arcu@velnislQuisque.ca', '1975-05-17', NULL,
'90 Cunnery Rd', 'M2 1HE', 9800, 88);
INSERT INTO authenticated_user VALUES (18, 'Normal',
'mizunaspider',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Adam M Shah',
'Nulla@aliquetdiam.com', '1976-03-16', NULL, '62
Cunnery Rd', 'M60 9DW', 9800, 88);
INSERT INTO authenticated_user VALUES (20, 'Normal',
'dilationwaals',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Naomi L Davison',
'ultrices@malesuadavelvenenatis.co.uk', '1975-05-17',
NULL, '90 Cunnery Rd', 'M2 1HE', 9800, 88);
INSERT INTO authenticated_user VALUES (21, 'Normal',
'watchinggeminii',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Adam M Shah',
'et.eros.Proin@vitaemaurissit.edu', '1976-03-16',
NULL, '62 Cunnery Rd', 'M60 9DW', 9800, 88);
INSERT INTO authenticated_user VALUES (22, 'Normal',
'cambrianend',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Mia S Burke',
'volutpat.ornare@Suspendisse.edu', '1996-11-03',
NULL, '73 York Road', 'OX1 4PP', 9800, 89);
INSERT INTO authenticated_user VALUES (23, 'Normal',
'pastiedash',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Naomi L Davison',

```

```

'nec.urna@ascelerisquesed.com', '1975-05-17', NULL,
'90 Cunnery Rd', 'M2 1HE', 9800, 88);
INSERT INTO authenticated_user VALUES (24, 'Normal',
'jockeypruning',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Adam M Shah',
'ut.ipsuM.ac@quismassaMauris.net', '1976-03-16',
NULL, '62 Cunnery Rd', 'M60 9DW', 9800, 88);
INSERT INTO authenticated_user VALUES (26,
'Moderator', 'starkshell',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Nádia Carvalho', NULL, NULL, NULL,
NULL, NULL, 9800, NULL);
INSERT INTO authenticated_user VALUES (27,
'Moderator', 'bolidesorrowful',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Nádia Carvalho', NULL, NULL, NULL,
NULL, NULL, 9800, NULL);
INSERT INTO authenticated_user VALUES (28,
'Moderator', 'valuemonitoring',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Nádia Carvalho', NULL, NULL, NULL,
NULL, NULL, 9800, NULL);
INSERT INTO authenticated_user VALUES (29,
'Moderator', 'abovedoubtful',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Nádia Carvalho', NULL, NULL, NULL,
NULL, NULL, 9800, NULL);
INSERT INTO authenticated_user VALUES (30,
'Moderator', 'expectantridge',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Nádia Carvalho', NULL, NULL, NULL,
NULL, NULL, 9800, NULL);
INSERT INTO authenticated_user VALUES (2, 'Normal',
'Ritmock',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', 'http://4.bp.blogspot.com/-SoLmj_KD-
oE/VCbiHfUzMjI/AAAAAAAAAD-A/mlZ-3j-
TAvk/s1600/Ian%2BH%2B(7).jpg', 'Max H Harding',

```

```

'lectus.sit@ligula.co.uk', '1981-05-28', NULL, '82
Guild Street', 'NW6 3LF', 9800, 19);
INSERT INTO authenticated_user VALUES (10, 'Normal',
'rabbitsfootortolan',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', 'http://4.bp.blogspot.com/-SoLmj_KD-
oE/VCbiHfUzMjI/AAAAAAAAD-A/mlZ-3j-
TAvk/s1600/Ian%2BH%2B(7).jpg', 'Max H Harding',
'Curabitur.sed.tortor@quisdiamPellentesque.net',
'1981-05-28', 4, '82 Guild Street', 'NW6 3LF', 9800,
19);
INSERT INTO authenticated_user VALUES (19, 'Normal',
'unbecominglabour',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Mia S Burke',
'Nullam@ullamcorpervelit.net', '1996-11-03', NULL,
'73 York Road', 'OX1 4PP', 500, 89);
INSERT INTO authenticated_user VALUES (8, 'Normal',
'helicopteruk',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Adam M Shah',
'panama@smellypotato.tk', '1976-03-16', 1, '62
Cunnery Rd', 'M60 9DW', 9800, 88);
INSERT INTO authenticated_user VALUES (11, 'Normal',
'laggedcolony',
'$2y$12$QKkHP76S.Z38LSMDzbUHF.c5s2.hvancdh2vJMx4Vozmm
9RdX5C.G', NULL, 'Naomi L Davison',
'Vivamus@Etiam.ca', '1975-05-17', NULL, '90 Cunnery
Rd', 'M2 1HE', 8995, 88);

--
-- Data for Name: add_credits; Type: TABLE DATA;
Schema: public; Owner: lbaw1716
--

INSERT INTO add_credits VALUES (1, 150, '2018-03-26
16:10:10+01', 'lectus.sit@ligula.co.uk', 2);
INSERT INTO add_credits VALUES (3, 20, '2018-03-26

```



```

16:10:10+01',
'quis.accumsan.convallis@pellentesquetellus.org', 7);
INSERT INTO add_credits VALUES (4, 300, '2018-03-27
16:10:10+01', 'lectus.sit@ligula.co.uk', 2);
INSERT INTO add_credits VALUES (2, 130, '2018-03-26
16:10:10+01', 'non.lobortis@ultricesposuere.org', 3);

```

```

--
-- Data for Name: auction; Type: TABLE DATA; Schema:
public; Owner: lbaw1716
--

```

```

INSERT INTO auction VALUES (1, 'Active', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'not used
at the moment', 'https://images-na.ssl-images-
amazon.com/images/I/51P963Jq89L._AC_SR201,266_.jpg',
150, 500, 600, '2018-03-31 16:10:10+01', '2016-04-07
16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL, 10,
NULL, 30);
INSERT INTO auction VALUES (3, 'Rejected', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'New, with some factory malformation but does not
influence performance.', 'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 300, 400, 600, '2018-04-26 16:10:10+01', '2018-04-
30 16:10:10+01', '2018-04-26 22:10:10+01', NULL, 'How
can you prove that it has no malfunctions? We are not
convinced...', NULL, NULL, NULL, 3, NULL, 28);
INSERT INTO auction VALUES (5, 'Pending', 'Apple
MacBook Pro 13'' ', 'Apple MacBook Pro 13'' ',
'New!!!!', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-
Bar-e-Touch-ID-Cinzeno-Sideral.jpg', 800, 1200,
1500, '2018-03-26 16:10:10+01', '2018-03-31

```

```

16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL, 9,
NULL, NULL);
INSERT INTO auction VALUES (6, 'Rejected', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'New, with some factory malformation but does not
influence performance.', 'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 300, 400, 600, '2018-04-26 16:10:10+01', '2018-04-
30 16:10:10+01', '2018-04-26 22:10:10+01', NULL, 'How
can you prove that it has no malfunctions? We are not
convinced...', NULL, NULL, NULL, 3, NULL, 28);
INSERT INTO auction VALUES (8, 'Pending', 'Apple
MacBook Pro 13'' ', 'Apple MacBook Pro 13'' ',
'New!!!!', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-
Bar-e-Touch-ID-Cinzeno-Sideral.jpg', 800, 1200,
1500, '2018-03-26 16:10:10+01', '2018-03-31
16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL, 9,
NULL, NULL);
INSERT INTO auction VALUES (9, 'Rejected', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'New, with some factory malformation but does not
influence performance.', 'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 300, 400, 600, '2018-04-26 16:10:10+01', '2018-04-
30 16:10:10+01', '2018-04-26 22:10:10+01', NULL, 'How
can you prove that it has no malfunctions? We are not
convinced...', NULL, NULL, NULL, 3, NULL, 28);
INSERT INTO auction VALUES (10, 'Active', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'not used
at the moment', 'https://images-na.ssl-images-
amazon.com/images/I/51P963Jq89L._AC_SR201,266_.jpg',
150, 500, 600, '2018-03-31 16:10:10+01', '2016-04-07
16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL, 10,

```



```

NULL, 30);
INSERT INTO auction VALUES (11, 'Pending', 'Apple
MacBook Pro 13'' ', 'Apple MacBook Pro 13'' ',
'New!!!!', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-
Bar-e-Touch-ID-Cinzeno-Sideral.jpg', 800, 1200,
1500, '2018-03-26 16:10:10+01', '2018-03-31
16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL, 9,
NULL, NULL);
INSERT INTO auction VALUES (12, 'Rejected', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'New, with some factory malformation but does not
influence performance.', 'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 300, 400, 600, '2018-04-26 16:10:10+01', '2018-04-
30 16:10:10+01', '2018-04-26 22:10:10+01', NULL, 'How
can you prove that it has no malfunctions? We are not
convinced...', NULL, NULL, NULL, 3, NULL, 28);
INSERT INTO auction VALUES (14, 'Pending', 'Apple
MacBook Pro 13'' ', 'Apple MacBook Pro 13'' ',
'New!!!!', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-
Bar-e-Touch-ID-Cinzeno-Sideral.jpg', 800, 1200,
1500, '2018-03-26 16:10:10+01', '2018-03-31
16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL, 9,
NULL, NULL);
INSERT INTO auction VALUES (15, 'Rejected', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'New, with some factory malformation but does not
influence performance.', 'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 300, 400, 600, '2018-04-26 16:10:10+01', '2018-04-
30 16:10:10+01', '2018-04-26 22:10:10+01', NULL, 'How

```

```

can you prove that it has no malfunctions? We are not
convinced...', NULL, NULL, NULL, 3, NULL, 28);
INSERT INTO auction VALUES (16, 'Active', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'not used
at the moment', 'https://images-na.ssl-images-
amazon.com/images/I/51P963Jq89L._AC_SR201,266_.jpg',
150, 500, 600, '2018-03-31 16:10:10+01', '2016-04-07
16:10:10+01', NULL, NULL, NULL, NULL, NULL, 10,
NULL, 30);
INSERT INTO auction VALUES (17, 'Pending', 'Apple
MacBook Pro 13'' ', 'Apple MacBook Pro 13'' ',
'New!!!!', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-
Bar-e-Touch-ID-Cinzeno-Sideral.jpg', 800, 1200,
1500, '2018-03-26 16:10:10+01', '2018-03-31
16:10:10+01', NULL, NULL, NULL, NULL, NULL, 9,
NULL, NULL);
INSERT INTO auction VALUES (18, 'Rejected', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'New, with some factory malformation but does not
influence performance.', 'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 300, 400, 600, '2018-04-26 16:10:10+01', '2018-04-
30 16:10:10+01', '2018-04-26 22:10:10+01', NULL, 'How
can you prove that it has no malfunctions? We are not
convinced...', NULL, NULL, NULL, 3, NULL, 28);
INSERT INTO auction VALUES (20, 'Pending', 'Apple
MacBook Pro 13'' ', 'Apple MacBook Pro 13'' ',
'New!!!!', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-
Bar-e-Touch-ID-Cinzeno-Sideral.jpg', 800, 1200,
1500, '2018-03-26 16:10:10+01', '2018-03-31
16:10:10+01', NULL, NULL, NULL, NULL, NULL, 9,

```

```

NULL, NULL);
INSERT INTO auction VALUES (21, 'Rejected', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'New, with some factory malformation but does not
influence performance.', 'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 300, 400, 600, '2018-04-26 16:10:10+01', '2018-04-
30 16:10:10+01', '2018-04-26 22:10:10+01', NULL, 'How
can you prove that it has no malfunctions? We are not
convinced...', NULL, NULL, NULL, 3, NULL, 28);
INSERT INTO auction VALUES (23, 'Pending', 'Apple
MacBook Pro 13'' ', 'Apple MacBook Pro 13'' ',
'New!!!!', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-
Bar-e-Touch-ID-Cinzeno-Sideral.jpg', 800, 1200,
1500, '2018-03-26 16:10:10+01', '2018-03-31
16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL, 9,
NULL, NULL);
INSERT INTO auction VALUES (4, 'Active', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'not used
at the moment', 'https://images-na.ssl-images-
amazon.com/images/I/51P963Jq89L._AC_SR201,266_.jpg',
150, 500, 600, '2018-03-31 16:10:10+01', '2016-04-07
16:10:10+01', NULL, NULL, NULL, NULL, NULL, 5, 10,
NULL, 30);
INSERT INTO auction VALUES (13, 'Active', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'not used at the moment',
'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 150, 500, 600, '2018-03-31 16:10:10+01', '2016-04-
07 16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL,
10, NULL, 30);
INSERT INTO auction VALUES (22, 'Active', 'Dell

```

```

Premium Desktop Tower with Keyboard&Mouse', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'not used
at the moment', 'https://images-na.ssl-images-
amazon.com/images/I/51P963Jq89L._AC_SR201,266_.jpg',
150, 500, 600, '2018-03-31 16:10:10+01', '2016-04-07
16:10:10+01', NULL, NULL, NULL, NULL, NULL, NULL, 9,
NULL, 30);
INSERT INTO auction VALUES (19, 'Active', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'not used
at the moment', 'https://images-na.ssl-images-
amazon.com/images/I/51P963Jq89L._AC_SR201,266_.jpg',
150, 500, 600, '2018-03-31 16:10:10+01', '2016-04-07
16:10:10+01', NULL, NULL, NULL, NULL, NULL, 1, 8,
NULL, 30);
INSERT INTO auction VALUES (24, 'Rejected', 'HDD -
Toshiba 2TB 7200RPM ', 'HDD - Toshiba 2TB 7200RPM ',
'New, with some factory malformation but does not
influence performance.', 'https://1.bp.blogspot.com/-
wrxKaUtdJPI/WRNCILD7jhI/AAAAAAAAIpc/mBn7HHLv80wwqALCx
pqxeG7c6RxV63upACK4B/s320/Toshiba%2B2TB%2B7200RPM.jpg
', 300, 400, 600, '2018-04-26 16:10:10+01', '2018-04-
30 16:10:10+01', '2018-04-26 22:10:10+01', NULL, 'How
can you prove that it has no malfunctions? We are not
convinced...', NULL, NULL, NULL, 3, NULL, 28);
INSERT INTO auction VALUES (26, 'Active', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'Dell
Premium Desktop Tower with Keyboard&Mouse', 'not used
at the moment', 'https://images-na.ssl-images-
amazon.com/images/I/51P963Jq89L._AC_SR201,266_.jpg',
150, 350, 400, '2018-04-03 14:07:14.311868+01',
'2018-04-10 10:40:28.876944+01', NULL, NULL, NULL,
NULL, NULL, NULL, 2, NULL, 1);
INSERT INTO auction VALUES (2, 'Pending', 'Apple
MacBook Pro 13'' ', 'Apple MacBook Pro 13'' ',
'New!!!!', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-

```

```

Bar-e-Touch-ID-Cinzento-Sideral.jpg', 800, 1200,
1500, '2018-03-26 16:10:10+01', '2018-03-31
16:10:10+01', NULL, NULL, NULL, NULL, NULL, 9,
NULL, NULL);
INSERT INTO auction VALUES (25, 'Over', 'Dell Premium
Desktop Tower with Keyboard&Mouse', 'Dell Premium
Desktop Tower with Keyboard&Mouse', 'not used at the
moment', 'https://images-na.ssl-images-
amazon.com/images/I/51P963Jq89L._AC_SR201,266_.jpg',
150, 500, 600, '2018-03-31 16:10:10+01', '2016-04-07
16:10:10+01', NULL, NULL, NULL, '2016-04-07
16:10:10+01', 570, 3, 10, 2, 30);
INSERT INTO auction VALUES (7, 'Over', 'Apple MacBook
Pro 13'' ', 'Apple MacBook Pro 13'' ', 'not used at
the moment', 'https://static.fnac-
static.com/multimedia/Images/PT/NR/d9/16/12/1185497/1
540-1/tsp20170623132310/Apple-MacBook-Pro-13-Retina-
i5-3-1GHz-8GB-256GB-Intel-Iris-Plus-650-com-Touch-
Bar-e-Touch-ID-Cinzento-Sideral.jpg', 150, 500, 600,
'2018-03-31 16:10:10+01', '2016-04-07 16:10:10+01',
NULL, NULL, NULL, '2018-04-04 09:45:40.80046+01',
600, NULL, 10, 11, 30);

--
-- Name: auto_increment_auction; Type: SEQUENCE SET;
Schema: public; Owner: lbaw1716
--

SELECT pg_catalog.setval('auto_increment_auction',
28, true);

--
-- Name: auto_increment_category; Type: SEQUENCE SET;
Schema: public; Owner: lbaw1716
--

SELECT pg_catalog.setval('auto_increment_category',

```

```

10, true);

--
-- Name: auto_increment_city; Type: SEQUENCE SET;
Schema: public; Owner: lbaw1716
--

SELECT pg_catalog.setval('auto_increment_city', 150,
true);

--
-- Name: auto_increment_comment; Type: SEQUENCE SET;
Schema: public; Owner: lbaw1716
--

SELECT pg_catalog.setval('auto_increment_comment', 6,
true);

--
-- Name: auto_increment_credits; Type: SEQUENCE SET;
Schema: public; Owner: lbaw1716
--

SELECT pg_catalog.setval('auto_increment_credits', 4,
true);

--
-- Name: auto_increment_notification; Type: SEQUENCE
SET; Schema: public; Owner: lbaw1716
--

SELECT
pg_catalog.setval('auto_increment_notification', 1,
true);

```

```

--
-- Name: auto_increment_user; Type: SEQUENCE SET;
Schema: public; Owner: lbaw1716
--

SELECT pg_catalog.setval('auto_increment_user', 30,
true);

--
-- Data for Name: bid; Type: TABLE DATA; Schema:
public; Owner: lbaw1716
--

INSERT INTO bid VALUES ('2018-03-31 20:10:10+01',
165, 1, 3, NULL);
INSERT INTO bid VALUES ('2018-03-31 20:20:10+01',
160, 1, 5, NULL);
INSERT INTO bid VALUES ('2018-03-31 16:30:10+01',
175, 1, 6, NULL);
INSERT INTO bid VALUES ('2018-03-31 16:15:10+01',
160, 13, 9, NULL);
INSERT INTO bid VALUES ('2018-03-31 16:20:10+01',
160, 4, 11, NULL);
INSERT INTO bid VALUES ('2016-04-07 16:10:10+01',
570, 25, 2, NULL);
INSERT INTO bid VALUES ('2018-03-31 16:15:10+01',
150, 22, 4, NULL);
INSERT INTO bid VALUES ('2018-04-04
01:57:47.609897+01', 500, 1, 19, NULL);
INSERT INTO bid VALUES ('2018-04-04
09:34:44.568129+01', 200, 7, 11, false);
INSERT INTO bid VALUES ('2018-04-04
09:38:20.780618+01', 205, 7, 11, NULL);
INSERT INTO bid VALUES ('2018-04-04
09:45:40.80046+01', 600, 7, 11, true);
INSERT INTO bid VALUES ('2018-04-04
10:18:45.488189+01', 505, 1, 6, NULL);

```

```
INSERT INTO bid VALUES ('2018-04-04
10:20:12.186704+01', 600, 1, 8, NULL);
INSERT INTO bid VALUES ('2018-04-04
10:21:40.598228+01', 650, 1, 11, NULL);
INSERT INTO bid VALUES ('2018-04-04
10:29:30.442617+01', 660, 1, 5, NULL);
```

```
--
-- Data for Name: blocks; Type: TABLE DATA; Schema:
public; Owner: lbaw1716
--
```

```
INSERT INTO blocks VALUES ('Blocked', 'This user did
not respect the rules of the site in doing comments
with insults.', '2018-03-26 16:10:10+01', 25, 29);
```

```
--
-- Data for Name: category; Type: TABLE DATA; Schema:
public; Owner: lbaw1716
--
```

```
INSERT INTO category VALUES (1, 'Towers w/
Components', NULL);
INSERT INTO category VALUES (2, 'Towers', NULL);
INSERT INTO category VALUES (3, 'Laptops', NULL);
INSERT INTO category VALUES (4, 'Components
', NULL);
INSERT INTO category VALUES (5, 'Peripherals', NULL);
INSERT INTO category VALUES (6, 'Motherboards', 4);
INSERT INTO category VALUES (7, 'Processors', 4);
INSERT INTO category VALUES (8, 'RAM', 4);
INSERT INTO category VALUES (9, 'Graphics/Video Cards
', 4);
INSERT INTO category VALUES (10, 'Power Supplies ',
4);
```



```

--
-- Data for Name: categoryofauction; Type: TABLE
DATA; Schema: public; Owner: lbaw1716
--

INSERT INTO categoryofauction VALUES (1, 1);
INSERT INTO categoryofauction VALUES (3, 2);
INSERT INTO categoryofauction VALUES (4, 3);
INSERT INTO categoryofauction VALUES (1, 4);
INSERT INTO categoryofauction VALUES (3, 5);
INSERT INTO categoryofauction VALUES (1, 25);

--
-- Data for Name: comment; Type: TABLE DATA; Schema:
public; Owner: lbaw1716
--

INSERT INTO comment VALUES (1, '2018-03-31
16:10:10+01', 'Engaging. It keeps your mind occupied
while you wait.', 1, 2);
INSERT INTO comment VALUES (2, '2018-03-31
16:12:10+01', 'Overly slick m8!', 10, 10);
INSERT INTO comment VALUES (4, '2018-03-31
16:17:10+01', 'It''s nice not just beastly!', 7, 10);
INSERT INTO comment VALUES (3, '2018-03-31
16:15:10+01', 'Just magnificent, friend.', 7, 8);
INSERT INTO comment VALUES (5, '2018-03-31
16:20:10+01', 'Bold. So neat.', 10, 11);
INSERT INTO comment VALUES (6, '2018-03-31
16:20:10+01', 'Incredibly excellent shapes!!', 13,
10);

--
-- Data for Name: edit_categories; Type: TABLE DATA;
Schema: public; Owner: lbaw1716
--

```

```

INSERT INTO edit_categories VALUES (1, 1);
INSERT INTO edit_categories VALUES (2, 1);
INSERT INTO edit_categories VALUES (3, 1);
INSERT INTO edit_categories VALUES (4, 1);
INSERT INTO edit_categories VALUES (5, 1);
INSERT INTO edit_categories VALUES (6, 1);
INSERT INTO edit_categories VALUES (7, 1);
INSERT INTO edit_categories VALUES (8, 1);
INSERT INTO edit_categories VALUES (9, 1);
INSERT INTO edit_categories VALUES (10, 1);

--
-- Data for Name: edit_moderator; Type: TABLE DATA;
Schema: public; Owner: lbaw1716
--

INSERT INTO edit_moderator VALUES (26, 1);
INSERT INTO edit_moderator VALUES (27, 1);
INSERT INTO edit_moderator VALUES (28, 1);
INSERT INTO edit_moderator VALUES (29, 1);
INSERT INTO edit_moderator VALUES (30, 1);

--
-- Data for Name: lastbidder; Type: TABLE DATA;
Schema: public; Owner: lbaw1716
--

INSERT INTO lastbidder VALUES (19);
INSERT INTO lastbidder VALUES (6);
INSERT INTO lastbidder VALUES (3);
INSERT INTO lastbidder VALUES (5);

--
-- Data for Name: notification; Type: TABLE DATA;
Schema: public; Owner: lbaw1716
--

```

```
INSERT INTO notification VALUES (1, '2018-04-04
10:29:30.442617+01', 'Your bid on this auction was
surpassed. Try again!', 'Bid Exceeded', 1, 11);
```

```
--
```

```
-- Data for Name: report; Type: TABLE DATA; Schema:
public; Owner: lbaw1716
```

```
--
```

```
INSERT INTO report VALUES ('2018-03-31 20:10:10+01',
'"It may continue to grow in China, but not in Europe
or the U.S. We don''t need the imaginary outlet to
feel a sense of accomplishment here."
```

```
', 1, 25);
```

```
INSERT INTO report VALUES ('2018-03-31 20:20:10+01',
'"To believe that Apple can somehow succeed where all
others have failed is to ignore some fundamental
realities of tablet computing."
```

```
', 7, 8);
```

GROUP1716, 04/04/2018

- Diogo Peixoto Pereira, diogopeixotopereira@gmail.com
- Igor Bernardo Amorim Silveira, igorasilveira@gmail.com
- Nádia de Sousa Varela de Carvalho, nadiacarvalho118@gmail.com
- Pedro Miguel Ferraz Nogueira da Silva, pedronogueirasilva5@gmail.com