COMP 421: Project 2

Group 87

Gabriel Alacchi - 260740414
Yunus Can Cukran - 260669715
Rebecca Jaubert - 260720767
Owen Lewis - 260706614

1 - Modified Relational Schema

We did not receive any feedback on our schema, and therefore we have kept it nearly the same, we just added a `banned_date` attribute for players so that we can represent banning a user. We also changed some of the attribute names to be more consistent (i.e. since username is the primary key of players, we rename the attributes which REF players to be named x_username as opposed to x_id).

ENTITIES:

Items(item_id, name, price, weight)
Weapons(item_id, ammo, type, range, damage) (item_id REF Items)
Armors(item_id, protection) (item_id REF Items)
Attachments(item_id, attaches_to_id)
(item_id REF Items
attaches_to_id REF Weapons)
Players(username, email, coin_balance, level, regist_date, exp, banned_date, guild_name, guild_join_date)
(guild_id REF Guilds)
GameSessions(gid, game_type, map_name, start_time, end_time, winning_team_nb)
CoinPurchases(pid, username, coins, amount_paid, trans_date) (username ref Players)
Guilds(name, created_date, admin_username) (admin_username REF Players)

RELATIONSHIPS:

Owns(username, item_id, purchase_date)
(username REF Players,
Item_id REF Items)
InventoryContains(username, name, item_id)
((username, name) REF Inventories,
Item_id REF Items)
Plays(username, inv_name, gid, kills, deaths, assists, team_nb)
((username, inv_name) REF Inventories,
gid REF GameSessions)
Friends(requester_username, requestee_username, request_date, accept_date)
(requester_username REF Players,
requestee_username REF Players)

WEAK ENTITIES:

Inventories(name, username) (username REF Players)

2 - SQL Schema

Note: *The SQL itself can be found in the* ***schema.sql*** *file, also submitted. We have provided the /d outputs here.*

```
cs421=> \d friends
                      Table "cs421g87.friends"
       Column       |            Type             |       Modifiers
--------------------+-----------------------------+------------------------
 requester_username | character varying(255)      | not null
 requestee_username | character varying(255)      | not null
 request_date       | timestamp without time zone | not null default now()
 accept_date        | timestamp without time zone |
Indexes:
    "friends_pkey" PRIMARY KEY, btree (requester_username, requestee_username)
Foreign-key constraints:
    "friends_players_requestee_username_fk" FOREIGN KEY (requestee_username) REF
ERENCES players(username) ON UPDATE CASCADE ON DELETE CASCADE
    "friends_players_requester_username_fk" FOREIGN KEY (requester_username) REF
ERENCES players(username) ON UPDATE CASCADE ON DELETE CASCADE
```

```
cs421=> \d weapons
           Table "cs421g87.weapons"
 Column  |          Type          | Modifiers
---------+------------------------+-----------
 item_id | bigint                 | not null
 ammo    | integer                | not null
 type    | character varying(255) | not null
 range   | integer                | not null
 damage  | integer                | not null
Indexes:
    "weapons_pkey" PRIMARY KEY, btree (item_id)
Check constraints:
    "weapons_ammo_check" CHECK (ammo >= 0)
    "weapons_damage_check" CHECK (damage >= 0)
    "weapons_range_check" CHECK (range >= 0)
    "weapons_type_check" CHECK (type::text = ANY (ARRAY['primary'::character var
ying, 'sidearm'::character varying, 'melee'::character varying]::text[]))
Foreign-key constraints:
    "weapons_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id) ON UPDATE
CASCADE ON DELETE CASCADE
```

```
cs421=> \d plays
                Table "cs421g87.plays"
     Column      |          Type          | Modifiers
-----------------+------------------------+-----------
 username        | character varying(255) | not null
 inventory_name  | character varying(255) | not null
 gid             | bigint                 | not null
 kills           | integer                | default 0
 deaths          | integer                | default 0
 assists         | integer                | default 0
 team_number     | integer                | not null
Indexes:
    "plays_pkey" PRIMARY KEY, btree (username, inventory_name, gid)
Check constraints:
    "plays_assists_check" CHECK (assists >= 0)
    "plays_deaths_check" CHECK (deaths >= 0)
    "plays_kills_check" CHECK (kills >= 0)
Foreign-key constraints:
    "plays_gid_fkey" FOREIGN KEY (gid) REFERENCES game_sessions(gid) ON UPDATE C
ASCADE ON DELETE RESTRICT
    "plays_username_fkey" FOREIGN KEY (username, inventory_name) REFERENCES inve
ntories(username, name) ON UPDATE CASCADE ON DELETE RESTRICT
```

```
                          Table "cs421g87.players"
     Column      |            Type             |           Modifiers
-----------------+-----------------------------+--------------------------------
-
 username        | character varying(255)      | not null
 email           | character varying(255)      | not null
 coin_balance    | integer                     | not null default 0
 level           | integer                     | not null default 1
 register_date   | timestamp without time zone | not null default now()
 experience      | integer                     | not null default 0
 banned_date     | timestamp without time zone |
 guild_name      | character varying(255)      | default NULL::character varying
 guild_join_date | timestamp without time zone |
Indexes:
    "players_pkey" PRIMARY KEY, btree (username)
    "players_email_key" UNIQUE CONSTRAINT, btree (email)
Foreign-key constraints:
    "guild_membership_fkey" FOREIGN KEY (guild_name) REFERENCES guilds(name) ON
UPDATE CASCADE ON DELETE SET NULL
Referenced by:
    TABLE "coin_purchases" CONSTRAINT "coin_purchases_username_fkey" FOREIGN KEY
 (username) REFERENCES players(username) ON UPDATE CASCADE ON DELETE RESTRICT
```

```
cs421=> \d owns
                    Table "cs421g87.owns"
    Column     |            Type             |       Modifiers
---------------+-----------------------------+------------------------
 username      | character varying(255)      | not null
 item_id       | bigint                      | not null
 purchase_date | timestamp without time zone | not null default now()
Indexes:
    "owns_pkey" PRIMARY KEY, btree (username, item_id)
Foreign-key constraints:
    "owns_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id) ON UPDATE CAS
CADE ON DELETE CASCADE
    "owns_players_username_fk" FOREIGN KEY (username) REFERENCES players(usernam
e) ON UPDATE CASCADE ON DELETE CASCADE
```

```
                                Table "cs421g87.items"
 Column |          Type          |                      Modifiers
--------+------------------------+-----------------------------------------------
------
 id     | integer                | not null default nextval('items_id_seq'::regc
lass)
 name   | character varying(255) | not null
 price  | integer                | not null
 weight | integer                | not null
Indexes:
    "items_pkey" PRIMARY KEY, btree (id)
Check constraints:
    "items_price_check" CHECK (price > 0)
    "items_weight_check" CHECK (weight >= 0)
Referenced by:
    TABLE "armors" CONSTRAINT "armors_item_id_fkey" FOREIGN KEY (item_id) REFERE
NCES items(id) ON UPDATE CASCADE ON DELETE CASCADE
    TABLE "attachments" CONSTRAINT "attachments_attaches_to_id_fkey" FOREIGN KEY
 (attaches_to_id) REFERENCES items(id) ON UPDATE CASCADE ON DELETE RESTRICT
    TABLE "attachments" CONSTRAINT "attachments_item_id_fkey" FOREIGN KEY (item_
id) REFERENCES items(id) ON UPDATE CASCADE ON DELETE CASCADE
    TABLE "inventory_contains" CONSTRAINT "inventory_contains_item_id_fkey" FORE
```

```
cs421=> \d inventory_contains
      Table "cs421g87.inventory_contains"
  Column  |          Type          | Modifiers
----------+------------------------+-----------
 username | character varying(255) | not null
 name     | character varying(255) | not null
 item_id  | bigint                 | not null
Indexes:
    "inventory_contains_pkey" PRIMARY KEY, btree (username, name, item_id)
Foreign-key constraints:
    "inventory_contains_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id)
 ON UPDATE CASCADE ON DELETE CASCADE
    "inventory_contains_username_fkey" FOREIGN KEY (username, name) REFERENCES i
nventories(username, name) ON UPDATE CASCADE ON DELETE CASCADE
```

```
cs421=> \d inventories
         Table "cs421g87.inventories"
  Column  |          Type          | Modifiers
----------+------------------------+-----------
 username | character varying(255) | not null
 name     | character varying(255) | not null
Indexes:
    "inventories_pkey" PRIMARY KEY, btree (name, username)
Foreign-key constraints:
    "inventories_players_username_fk" FOREIGN KEY (username) REFERENCES players(
username) ON UPDATE CASCADE ON DELETE CASCADE
Referenced by:
    TABLE "inventory_contains" CONSTRAINT "inventory_contains_username_fkey" FOR
EIGN KEY (username, name) REFERENCES inventories(username, name) ON UPDATE CASCA
DE ON DELETE CASCADE
    TABLE "plays" CONSTRAINT "plays_username_fkey" FOREIGN KEY (username, invent
ory_name) REFERENCES inventories(username, name) ON UPDATE CASCADE ON DELETE RES
TRICT
```

```
cs421=> \d guilds
                      Table "cs421g87.guilds"
     Column     |            Type             |       Modifiers
----------------+-----------------------------+------------------------
 name           | character varying(255)      | not null
 created_date   | timestamp without time zone | not null default now()
 admin_username | character varying(255)      | not null
Indexes:
    "guilds_pkey" PRIMARY KEY, btree (name)
Referenced by:
    TABLE "players" CONSTRAINT "guild_membership_fkey" FOREIGN KEY (guild_name)
REFERENCES guilds(name) ON UPDATE CASCADE ON DELETE SET NULL
```

```
                                    Table "cs421g87.game_sessions"
    Column    |            Type             |                              Modifiers

--------------+-----------------------------+-----------------------------------
-------------------------
 gid          | integer                     | not null default nextval('game_ses
sions_gid_seq'::regclass)
 game_type    | character varying(255)      |
 map_name     | character varying(255)      |
 winning_team | integer                     |
 start_time   | timestamp without time zone | default now()
 end_time     | timestamp without time zone |
Indexes:
    "game_sessions_pkey" PRIMARY KEY, btree (gid)
Referenced by:
    TABLE "plays" CONSTRAINT "plays_gid_fkey" FOREIGN KEY (gid) REFERENCES game_
sessions(gid) ON UPDATE CASCADE ON DELETE RESTRICT
```

```
                                  Table "cs421g87.coin_purchases"
   Column    |            Type             |                            Modifiers
-------------+-----------------------------+-------------------------------------
-------------------------
 pid         | integer                     | not null default nextval('coin_purc
hases_pid_seq'::regclass)
 username    | character varying(255)      | not null
 coins       | integer                     | not null default 0
 amount_paid | integer                     | not null default 0
 trans_date  | timestamp without time zone | default now()
Indexes:
    "coin_purchases_pkey" PRIMARY KEY, btree (pid)
Foreign-key constraints:
    "coin_purchases_username_fkey" FOREIGN KEY (username) REFERENCES players(use
rname) ON UPDATE CASCADE ON DELETE RESTRICT
```

```
cs421=> \d attachments
    Table "cs421g87.attachments"
     Column     |  Type  | Modifiers
----------------+--------+-----------
 item_id        | bigint | not null
 attaches_to_id | bigint | not null
Indexes:
    "attachments_pkey" PRIMARY KEY, btree (item_id)
Foreign-key constraints:
    "attachments_attaches_to_id_fkey" FOREIGN KEY (attaches_to_id) REFERENCES it
ems(id) ON UPDATE CASCADE ON DELETE RESTRICT
    "attachments_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id) ON UPD
ATE CASCADE ON DELETE CASCADE
```

```
cs421=> \d armors
     Table "cs421g87.armors"
   Column   |  Type   | Modifiers
------------+---------+-----------
 item_id    | bigint  | not null
 protection | integer | not null
Indexes:
    "armors_pkey" PRIMARY KEY, btree (item_id)
Check constraints:
    "armors_protection_check" CHECK (protection >= 0)
Foreign-key constraints:
    "armors_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id) ON UPDATE C
ASCADE ON DELETE CASCADE
```

```
cs421=> \d inventory_contains
      Table "cs421g87.inventory_contains"
  Column  |          Type          | Modifiers
----------+------------------------+-----------
 username | character varying(255) | not null
 name     | character varying(255) | not null
 item_id  | bigint                 | not null
Indexes:
    "inventory_contains_pkey" PRIMARY KEY, btree (username, name, item_id)
Foreign-key constraints:
    "inventory_contains_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id)
 ON UPDATE CASCADE ON DELETE CASCADE
    "inventory_contains_username_fkey" FOREIGN KEY (username, name) REFERENCES i
nventories(username, name) ON UPDATE CASCADE ON DELETE CASCADE
```

3 - Insert Commands

```
INSERT INTO players (username, email) VALUES
     ('gabriel', 'gabriel.alacchi@mail.mcgill.ca'),
     ('rebecca', 'rebecca.jaubert@mail.mcgill.ca'),
     ('owen', 'owen@mail.mcgill.ca'),
     ('yunus', 'yunus@mail.mcgill.ca'),
     ('dimdim75', 'dim@hotmail.fr'),
     ('lyrink', 'linus@gmail.com'),
     ('pestoHu94', 'paul.huchede@gmail.com'),
     ('rouDouBreh', 'richard@yahoo.ca'),
     ('inesK22', 'ines.kheyar@hotmail.fr'),
     ('gadu94zer', 'gabrielradwan@gmail.com'),
     ('mamounek', 'mamoune.kellal@gmail.com'),
     ('jokazer', 'arthur.roc@hotmail.fr'),
     ('zingalax', 'romaincharpentier@gmail.com'),
     ('conduit420', 'shadara@mail.mcgill.ca')
     ON CONFLICT DO NOTHING
```

```
----- Items ----- (id,name,price, weight)
INSERT INTO items VALUES
(1,'DL Q33 - Loup solitaire', 1000, 55),
(2, 'S36 - Requin abyssal',1600, 60),
(3,'HG 40 - hallow', 1600, 25),
(4,'couteau sanglant',560, 10),
(5, 'grenade paralysante', 350, 3),
(6,'type 25 - Tropiques', 800, 40),
(7, 'M21 EBR - Tropiques', 800, 55),
(8, 'couteau - Pasteque', 200, 10),
(9, 'grenade fumigène - vague bleue', 320, 3),
(10, 'AK-47', 500, 40),
(11, 'M21 EBR', 700, 55),
(12, 'Firebreak', 700, 45),
(13, 'Sac à dos - Firebreak', 120, 10),
(14, 'Uniforme Mante', 800, 55),
(15, 'Eclaireur Mante', 320, 3),
```

```
(16, 'Bombe clown Mante', 190, 18),
(17, 'Camouflage d hiver', 400, 20),
(18, 'Parachute - Glacier onirique', 350, 10),
(19, 'Eclaireur Bleu froid', 320, 3),
(20, 'M4LMG - Triangle rouge', 550, 60),
(21, 'POW-57 - Gene zombie', 1200, 25),
(22, 'RPD', 250, 60),
(23, 'BY15', 700, 25),
(24, 'Silencieux - Fusil à pompe', 550, 0),
(25, 'Chargeur étendu - Fusil à pompe', 500, 0),
(26, 'Viseur laser - Fusil à pompe', 500, 0),
(27, 'Lunette scopique - DL Q33', 600, 0),
(28, 'Viseur laser - DL Q33', 425, 0),
(29, 'Silencieux - S36', 550, 7),
(30, 'Poignée avant - S36', 350, 3),
(31, 'Chargeur étendu - HG 40', 500, 0),
(32, 'Canon long - HG 40', 100, 10),
(33, 'MW11', 200, 10),
(34, 'M19', 390, 10),
(35, '1911', 350, 10),
(36, 'Revolver 357', 500, 30),
(37, 'X16', 400, 10),
(38, 'Forces spéciales 1', 100, 30),
(39, 'Forces spéciales 2', 100, 30),
(40, 'Eclaireur', 150, 20)
ON CONFLICT DO NOTHING;
```

```
-- Weapons (id, ammo,type, range, damage)
INSERT INTO weapons VALUES
(1, 15, 'primary', 95, 90),
(2,75,'primary', 46, 75),
(3, 40,'primary', 42, 75),
(4, 10, 'melee', 10, 50),
(5,0,'melee', 60, 0),
(6, 70, 'primary', 35, 55),
(7,35,'primary',95,80),
(8, 10, 'melee', 10, 30) ,
(9, 0, 'melee', 65, 0),
(10, 55, 'primary', 66, 70),
(11, 35, 'primary', 95, 80),
(20, 55, 'primary', 45, 60),
(21, 50, 'primary', 25, 90),
(22, 65, 'primary', 50, 65),
(23, 20,'primary', 30, 85),
(33, 40, 'sidearm', 20, 55),
(34, 60, 'sidearm', 20, 35),
(35, 45, 'sidearm', 35, 45),
(36, 15, 'sidearm', 40, 70),
(37, 40, 'sidearm', 10, 50)
ON CONFLICT DO NOTHING;
```

```
-- Armors (id, protection)
INSERT INTO armors VALUES
(12, 55),
(14, 60),
(17, 32),
(38,10),
(39,10),
(40,10)
ON CONFLICT DO NOTHING;
```


```
-- Armors (id, protection)
INSERT INTO armors VALUES
(12, 55),
(14, 60),
(17, 32),
(38,10),
(39,10),
(40,10)
ON CONFLICT DO NOTHING;
```


Log Output when Running the 5 Commands

```
[2020-02-28 20:40:05] 14 rows affected in 11 ms
sql> INSERT INTO friends VALUES
     ('dimdim75','rebecca','2019-09-12 13:05:01','2019-09-12 17:05:01'),
     ('mamounek', 'dimdim75', '2019-05-09 11:23:13', '2019-05-15 18:45:05'),
     ('gadu94zer', 'dimdim75', '2019-05-22 14:03:16', '2019-06-02 19:48:54'),
     ('dimdim75', 'pestoHu94', '2019-10-16 08:37:17', '2019-10-17 20:33:14')
     ON CONFLICT DO NOTHING
[2020-02-28 20:40:05] 4 rows affected in 8 ms
sql> INSERT INTO items VALUES
     (1,'DL Q33 - Loup solitaire', 1000, 55),
     ...
     (40, 'Eclaireur', 150, 20)
     ON CONFLICT DO NOTHING
[2020-02-28 20:40:05] 40 rows affected in 9 ms
sql> INSERT INTO weapons VALUES
     (1, 15, 'primary', 95, 90),
     ...
     (37, 40, 'sidearm', 10, 50)
     ON CONFLICT DO NOTHING
[2020-02-28 20:40:05] 20 rows affected in 12 ms
sql> INSERT INTO armors VALUES
     (12, 55),
     ...
     (40,10)
     ON CONFLICT DO NOTHING
[2020-02-28 20:40:05] 6 rows affected in 12 ms
sql> INSERT INTO attachments VALUES
     (13, 12),
     ...
     (32,3)
```

```
    ON CONFLICT DO NOTHING
[2020-02-28 20:40:05] 14 rows affected in 13 ms
```

4 - Data Insertion

Owns

	username	item_id	purchase_date
1	gabriel	22	2020-02-28 14:03:05.015601
2	gabriel	34	2020-02-28 14:03:05.015601
3	gabriel	9	2020-02-28 14:03:05.015601
4	gabriel	11	2020-02-28 14:03:05.015601
5	gabriel	36	2020-02-28 14:03:05.015601

Players

	username	email	coin_balance	level	register_date	experience	banned_date	guild_name	guild_join_date
1	mamounek	mamoune.kellal@gmail.com	17900	57	2018-05-25 02:54:26.295636	0	<null>	Honored Admirals	2020-02-28 17:38:27.267850
2	zingalax	romaincharpentier@gmail.com	6100	22	2018-01-24 22:28:49.733603	0	<null>	<null>	<null>
3	rouDouBreh	richard@yahoo.ca	10300	49	2018-01-27 00:20:49.404473	0	<null>	<null>	<null>
4	dimdim75	dim@hotmail.fr	6700	55	2018-09-30 11:37:36.939617	0	<null>	Tempest Noobs	2020-02-28 17:38:27.267850
5	gabriel	gabriel.alacchi@mail.mcgill.ca	24100	2	2018-09-12 03:21:48.937529	0	<null>	Tempest Noobs	2020-02-28 17:38:27.267850

Plays

	username	inventory_name	gid	kills	deaths	assists	team_number
1	rebecca	rebecca-1	0	12	6	2	1
2	zingalax	zingalax-1	0	8	14	3	1
3	pestoHu94	pestoHu94-0	0	8	12	3	2
4	owen	owen-0	0	12	8	3	2
5	rouDouBreh	rouDouBreh-0	1	9	9	2	1
6	zingalax	zingalax-0	1	8	7	2	1
7	mamounek	mamounek-0	1	7	9	1	2
8	dimdim75	dimdim75-0	1	9	8	1	2
9	gabriel	gabriel-0	2	11	6	1	1
10	lyrink	lyrink-0	2	7	13	3	1

Weapons

	item_id	ammo	type	range	damage
1	1	15	primary	95	90
2	2	75	primary	46	75
3	3	40	primary	42	75
4	4	10	melee	10	50
5	5	0	melee	60	0

Armors

	item_id	protection
1	12	55
2	14	60
3	17	32
4	38	10
5	39	10

Attachments

	item_id	attaches_to_id
1	13	12
2	15	14
3	16	14
4	18	17
5	19	17

Coin_Purchases

	pid	username	coins	amount_paid	trans_date
1	1102	yunus	200	4	2019-10-25 02:26:27.091154
2	1103	yunus	200	4	2019-04-26 05:20:47.263537
3	1104	yunus	4800	96	2019-12-27 02:48:55.392702
4	1105	yunus	3600	72	2019-03-19 12:24:45.641769
5	1106	yunus	1900	38	2019-02-24 19:11:38.097365

Friends

	requester_username	requestee_username	request_date	accept_date
1	dimdim75	rebecca	2019-09-12 13:05:01.000000	2019-09-12 17:05:01.000000
2	mamounek	dimdim75	2019-05-09 11:23:13.000000	2019-05-15 18:45:05.000000
3	gadu94zer	dimdim75	2019-05-22 14:03:16.000000	2019-06-02 19:48:54.000000
4	dimdim75	pestoHu94	2019-10-16 08:37:17.000000	2019-10-17 20:33:14.000000

Game_Sessions

	gid	game_type	map_name	winning_team	start_time	end_time
1	0	search-and-destroy	Rust	1	2019-11-23 19:42:19.684102	2019-11-23 19:58:53.827717
2	1	search-and-destroy	Verdun	2	2019-07-12 17:07:42.684102	2019-07-12 17:22:43.185946
3	2	team-deathmatch	Verdun	2	2019-07-12 16:45:15.684102	2019-07-12 17:03:34.179163
4	3	search-and-destroy	Verdun	2	2019-08-21 13:40:20.684102	2019-08-21 14:02:51.056745
5	4	team-deathmatch	Rust	2	2019-06-15 23:04:10.684102	2019-06-15 23:26:09.576033

Guilds

	name	created_date	admin_username
1	Tempest Noobs	2020-02-28 14:03:09.355576	gabriel
2	Abandoned Helix	2020-02-28 14:03:09.355576	yunus
3	Hallowed Apocalypse	2020-02-28 14:03:09.355576	owen
4	Honored Admirals	2020-02-28 14:03:09.355576	rebecca

Inventories

	username	name
1	gabriel	gabriel-0
2	gabriel	gabriel-1
3	rebecca	rebecca-0
4	rebecca	rebecca-1
5	owen	owen-0

Inventory_Contains

	username	name	item_id
1	gabriel	gabriel-0	22
2	gabriel	gabriel-0	34
3	gabriel	gabriel-1	11
4	gabriel	gabriel-1	36
5	gabriel	gabriel-1	8

Items

	id	name	price	weight
1	1	DL Q33 - Loup solitaire	1000	55
2	2	S36 - Requin abyssal	1600	60
3	3	HG 40 - hallow	1600	25
4	4	couteau sanglant	560	10
5	5	grenade paralysante	350	3

5 - Queries

1. Display players having the weapon AK47 in their inventories

```
                                Table "cs421g87.items"
 Column |          Type          |                      Modifiers
--------+------------------------+-----------------------------------------------
------
 id     | integer                | not null default nextval('items_id_seq'::regc
lass)
 name   | character varying(255) | not null
 price  | integer                | not null
 weight | integer                | not null
Indexes:
    "items_pkey" PRIMARY KEY, btree (id)
Check constraints:
    "items_price_check" CHECK (price > 0)
    "items_weight_check" CHECK (weight >= 0)
Referenced by:
    TABLE "armors" CONSTRAINT "armors_item_id_fkey" FOREIGN KEY (item_id) REFERE
NCES items(id) ON UPDATE CASCADE ON DELETE CASCADE
    TABLE "attachments" CONSTRAINT "attachments_attaches_to_id_fkey" FOREIGN KEY
 (attaches_to_id) REFERENCES items(id) ON UPDATE CASCADE ON DELETE RESTRICT
    TABLE "attachments" CONSTRAINT "attachments_item_id_fkey" FOREIGN KEY (item_
id) REFERENCES items(id) ON UPDATE CASCADE ON DELETE CASCADE
    TABLE "inventory_contains" CONSTRAINT "inventory_contains_item_id_fkey" FORE
```

```
cs421=> \d inventory_contains
      Table "cs421g87.inventory_contains"
  Column  |          Type          | Modifiers
----------+------------------------+-----------
 username | character varying(255) | not null
 name     | character varying(255) | not null
 item_id  | bigint                 | not null
Indexes:
    "inventory_contains_pkey" PRIMARY KEY, btree (username, name, item_id)
Foreign-key constraints:
    "inventory_contains_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id)
 ON UPDATE CASCADE ON DELETE CASCADE
    "inventory_contains_username_fkey" FOREIGN KEY (username, name) REFERENCES i
nventories(username, name) ON UPDATE CASCADE ON DELETE CASCADE
```

```sql
--Display players having the weapon AK47 in their inventories
SELECT DISTINCT(username)
FROM inventory_contains c, items i
WHERE c.item_id = i.id AND i.name='AK-47';
```

4 rows

	username
1	gadu94zer
2	rouDouBreh
3	mamounek
4	dimdim75

2. Display the guilds that supplied us with the highest revenue in a given (month, year)

```
                        Table "cs421g87.players"
    Column      |            Type             |           Modifiers
----------------+-----------------------------+--------------------------------
-
 username       | character varying(255)      | not null
 email          | character varying(255)      | not null
 coin_balance   | integer                     | not null default 0
 level          | integer                     | not null default 1
 register_date  | timestamp without time zone | not null default now()
 experience     | integer                     | not null default 0
 banned_date    | timestamp without time zone |
 guild_name     | character varying(255)      | default NULL::character varying
 guild_join_date | timestamp without time zone |
Indexes:
    "players_pkey" PRIMARY KEY, btree (username)
    "players_email_key" UNIQUE CONSTRAINT, btree (email)
Foreign-key constraints:
    "guild_membership_fkey" FOREIGN KEY (guild_name) REFERENCES guilds(name) ON
UPDATE CASCADE ON DELETE SET NULL
Referenced by:
    TABLE "coin_purchases" CONSTRAINT "coin_purchases_username_fkey" FOREIGN KEY
 (username) REFERENCES players(username) ON UPDATE CASCADE ON DELETE RESTRICT
```

```
                                  Table "cs421g87.coin_purchases"
   Column    |            Type             |                          Modifiers
-------------+-----------------------------+-------------------------------------
-------------------------
 pid         | integer                     | not null default nextval('coin_purc
hases_pid_seq'::regclass)
 username    | character varying(255)      | not null
 coins       | integer                     | not null default 0
 amount_paid | integer                     | not null default 0
 trans_date  | timestamp without time zone | default now()
Indexes:
    "coin_purchases_pkey" PRIMARY KEY, btree (pid)
Foreign-key constraints:
    "coin_purchases_username_fkey" FOREIGN KEY (username) REFERENCES players(use
rname) ON UPDATE CASCADE ON DELETE RESTRICT
```

```sql
--Display the guilds that supplied us with the highest revenue in a given (month, year)
SELECT guild_name,  SUM(amount_paid) AS total
FROM
   (SELECT *
   FROM coin_purchases
   WHERE
       EXTRACT(year from trans_date) = 2019 AND EXTRACT(month from trans_date) = 3)
           t, players p
WHERE p.username = t.username AND p.guild_name IS NOT NULL
GROUP BY p.guild_name
ORDER BY total DESC;
```

Output | Display the guilds t...a given (month, year)

2 rows

	guild_name	total
1	Abandoned Helix	128
2	Hallowed Apocalypse	36

3. Weapons which have the highest kill to death ratio.

```
cs421=> \d plays
                Table "cs421g87.plays"
     Column     |          Type          | Modifiers
----------------+------------------------+-----------
 username       | character varying(255) | not null
 inventory_name | character varying(255) | not null
 gid            | bigint                 | not null
 kills          | integer                | default 0
 deaths         | integer                | default 0
 assists        | integer                | default 0
 team_number    | integer                | not null
Indexes:
    "plays_pkey" PRIMARY KEY, btree (username, inventory_name, gid)
Check constraints:
    "plays_assists_check" CHECK (assists >= 0)
    "plays_deaths_check" CHECK (deaths >= 0)
    "plays_kills_check" CHECK (kills >= 0)
Foreign-key constraints:
    "plays_gid_fkey" FOREIGN KEY (gid) REFERENCES game_sessions(gid) ON UPDATE C
ASCADE ON DELETE RESTRICT
    "plays_username_fkey" FOREIGN KEY (username, inventory_name) REFERENCES inve
ntories(username, name) ON UPDATE CASCADE ON DELETE RESTRICT
```

```
cs421=> \d inventory_contains
      Table "cs421g87.inventory_contains"
  Column  |          Type          | Modifiers
----------+------------------------+-----------
 username | character varying(255) | not null
 name     | character varying(255) | not null
 item_id  | bigint                 | not null
Indexes:
    "inventory_contains_pkey" PRIMARY KEY, btree (username, name, item_id)
Foreign-key constraints:
    "inventory_contains_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id)
 ON UPDATE CASCADE ON DELETE CASCADE
    "inventory_contains_username_fkey" FOREIGN KEY (username, name) REFERENCES i
nventories(username, name) ON UPDATE CASCADE ON DELETE CASCADE
```

```
cs421=> \d weapons
          Table "cs421g87.weapons"
 Column  |          Type          | Modifiers
---------+------------------------+-----------
 item_id | bigint                 | not null
 ammo    | integer                | not null
 type    | character varying(255) | not null
 range   | integer                | not null
 damage  | integer                | not null
Indexes:
    "weapons_pkey" PRIMARY KEY, btree (item_id)
Check constraints:
    "weapons_ammo_check" CHECK (ammo >= 0)
    "weapons_damage_check" CHECK (damage >= 0)
    "weapons_range_check" CHECK (range >= 0)
    "weapons_type_check" CHECK (type::text = ANY (ARRAY['primary'::character var
ying, 'sidearm'::character varying, 'melee'::character varying]::text[]))
Foreign-key constraints:
    "weapons_item_id_fkey" FOREIGN KEY (item_id) REFERENCES items(id) ON UPDATE
CASCADE ON DELETE CASCADE
```

```
                          Table "cs421g87.items"
 Column |          Type          |                     Modifiers
--------+------------------------+-----------------------------------------------
------
 id     | integer                | not null default nextval('items_id_seq'::regc
lass)
 name   | character varying(255) | not null
 price  | integer                | not null
 weight | integer                | not null
Indexes:
    "items_pkey" PRIMARY KEY, btree (id)
Check constraints:
    "items_price_check" CHECK (price > 0)
    "items_weight_check" CHECK (weight >= 0)
Referenced by:
    TABLE "armors" CONSTRAINT "armors_item_id_fkey" FOREIGN KEY (item_id) REFERE
NCES items(id) ON UPDATE CASCADE ON DELETE CASCADE
    TABLE "attachments" CONSTRAINT "attachments_attaches_to_id_fkey" FOREIGN KEY
 (attaches_to_id) REFERENCES items(id) ON UPDATE CASCADE ON DELETE RESTRICT
    TABLE "attachments" CONSTRAINT "attachments_item_id_fkey" FOREIGN KEY (item_
id) REFERENCES items(id) ON UPDATE CASCADE ON DELETE CASCADE
    TABLE "inventory_contains" CONSTRAINT "inventory_contains_item_id_fkey" FORE
```

Kill-Death Ratio (KDR) is defined as kills divided by deaths in a game, and is an important measure of skill of a player. Generally speaking it can be helpful to visualize trends of KDR with respect to items in the game, as this generally speaks towards potential items which are too powerful and need to be weakened to balance the game.

```sql
-- Weapons which have the highest kill to death ratio.
SELECT i.name, w.type, round(SUM(kills)::NUMERIC / SUM(deaths)::NUMERIC, 2) as kdr FROM
plays ps
INNER JOIN inventory_contains ic
 ON ps.username = ic.username AND ps.inventory_name = ic.name
INNER JOIN weapons w
 ON ic.item_id = w.item_id
INNER JOIN items i
```

```
 ON ic.item_id = i.id
GROUP BY i.name, w.type
ORDER BY kdr DESC;
```

18 rows

	name	type	kdr
1	HG 40 - hallow	primary	2.23
2	DL Q33 - Loup solitaire	primary	2.2
3	couteau sanglant	melee	1.75
4	POW-57 - Gene zombie	primary	1.2
5	1911	sidearm	1.19
6	S36 - Requin abyssal	primary	1.13
7	M21 EBR	primary	1.08
8	RPD	primary	1
9	couteau - Pasteque	melee	0.97
10	grenade fumigène - vague bleue	melee	0.97
11	MW11	sidearm	0.94
12	X16	sidearm	0.93
13	M19	sidearm	0.9
14	M4LMG - Triangle rouge	primary	0.82
15	Revolver 357	sidearm	0.8
16	BY15	primary	0.8
17	M21 EBR - Tropiques	primary	0.77
18	AK-47	primary	0.74

4. List details of games that were played between a particular player and all of their friends.

```
cs421=> \d friends
                    Table "cs421g87.friends"
       Column       |            Type             |       Modifiers
--------------------+-----------------------------+-----------------------
 requester_username | character varying(255)      | not null
 requestee_username | character varying(255)      | not null
 request_date       | timestamp without time zone | not null default now()
 accept_date        | timestamp without time zone |
Indexes:
    "friends_pkey" PRIMARY KEY, btree (requester_username, requestee_username)
Foreign-key constraints:
    "friends_players_requestee_username_fk" FOREIGN KEY (requestee_username) REF
ERENCES players(username) ON UPDATE CASCADE ON DELETE CASCADE
    "friends_players_requester_username_fk" FOREIGN KEY (requester_username) REF
ERENCES players(username) ON UPDATE CASCADE ON DELETE CASCADE
```

```
cs421=> \d plays
                Table "cs421g87.plays"
     Column      |          Type          | Modifiers
-----------------+------------------------+-----------
 username        | character varying(255) | not null
 inventory_name  | character varying(255) | not null
 gid             | bigint                 | not null
 kills           | integer                | default 0
 deaths          | integer                | default 0
 assists         | integer                | default 0
 team_number     | integer                | not null
Indexes:
    "plays_pkey" PRIMARY KEY, btree (username, inventory_name, gid)
Check constraints:
    "plays_assists_check" CHECK (assists >= 0)
    "plays_deaths_check" CHECK (deaths >= 0)
    "plays_kills_check" CHECK (kills >= 0)
Foreign-key constraints:
    "plays_gid_fkey" FOREIGN KEY (gid) REFERENCES game_sessions(gid) ON UPDATE C
ASCADE ON DELETE RESTRICT
    "plays_username_fkey" FOREIGN KEY (username, inventory_name) REFERENCES inve
ntories(username, name) ON UPDATE CASCADE ON DELETE RESTRICT
```

```sql
-- List details of games that were played between a particular player and all of their
friends.
WITH
 friend_list AS
   (SELECT requester_username AS friend FROM friends WHERE requestee_username =
'rebecca'
    UNION
    SELECT requestee_username AS friend FROM friends WHERE requester_username =
'rebecca'),
 games_played_together AS (SELECT gid FROM plays
                           WHERE username IN (SELECT friend from friend_list)
                           INTERSECT
                           SELECT gid FROM plays
                           WHERE username = 'rebecca')
SELECT gs.gid, username, kills, deaths, assists, team_number, gs.game_type, gs.map_name,
gs.winning_team FROM plays
INNER JOIN game_sessions gs on plays.gid = gs.gid
WHERE gs.gid IN (SELECT gid FROM games_played_together)
ORDER BY gs.gid;
```

	gid	username	kills	deaths	assists	team_number	game_type	map_name	winning_team
1	16	dimdim75	10	8	2	2	team-deathmatch	Rust	2
2	16	gadu94zer	7	6	2	2	team-deathmatch	Rust	2
3	16	rebecca	9	7	1	1	team-deathmatch	Rust	2
4	16	mamounek	5	10	2	1	team-deathmatch	Rust	2
5	22	rebecca	10	8	2	1	capture-the-flag	Vimy Ridge	2
6	22	owen	9	9	1	2	capture-the-flag	Vimy Ridge	2
7	22	dimdim75	8	10	1	2	capture-the-flag	Vimy Ridge	2
8	22	361060d6cf291f3b0dd04904fb724ca8	9	9	3	1	capture-the-flag	Vimy Ridge	2
9	71	dimdim75	13	7	2	1	capture-the-flag	Rust	1
10	71	rebecca	13	8	2	2	capture-the-flag	Rust	1
11	71	lyrink	8	13	3	2	capture-the-flag	Rust	1
12	71	zingalax	8	14	3	1	capture-the-flag	Rust	1
13	78	pestoHu94	7	10	1	2	team-deathmatch	Vimy Ridge	2
14	78	mamounek	9	14	1	1	team-deathmatch	Vimy Ridge	2
15	78	dimdim75	7	8	2	1	team-deathmatch	Vimy Ridge	2
16	78	rebecca	15	6	1	2	team-deathmatch	Vimy Ridge	2
17	90	mamounek	7	9	2	1	capture-the-flag	Verdun	2
18	90	dimdim75	7	6	1	2	capture-the-flag	Verdun	2
19	90	rebecca	10	5	1	1	capture-the-flag	Verdun	2
20	90	361060d6cf291f3b0dd04904fb724ca8	7	11	2	2	capture-the-flag	Verdun	2
21	102	yunus	12	5	1	2	search-and-destroy	Rust	1
22	102	dimdim75	8	11	3	2	search-and-destroy	Rust	1
23	102	rebecca	11	8	1	1	search-and-destroy	Rust	1
24	102	rouDouBreh	5	12	2	1	search-and-destroy	Rust	1
25	121	dimdim75	6	8	1	2	capture-the-flag	Verdun	1
26	121	361060d6cf291f3b0dd04904fb724ca8	7	10	1	2	capture-the-flag	Verdun	1
27	121	yunus	11	7	1	1	capture-the-flag	Verdun	1
28	121	rebecca	7	6	2	1	capture-the-flag	Verdun	1

5. Active Monthly Players, where an active player is defined as having participated in at least 1 game_session in that (month, year)

```
cs421=> \d plays
                Table "cs421g87.plays"
    Column     |          Type          | Modifiers
---------------+------------------------+-----------
 username      | character varying(255) | not null
 inventory_name | character varying(255) | not null
 gid           | bigint                 | not null
 kills         | integer                | default 0
 deaths        | integer                | default 0
 assists       | integer                | default 0
 team_number   | integer                | not null
Indexes:
    "plays_pkey" PRIMARY KEY, btree (username, inventory_name, gid)
Check constraints:
    "plays_assists_check" CHECK (assists >= 0)
    "plays_deaths_check" CHECK (deaths >= 0)
    "plays_kills_check" CHECK (kills >= 0)
Foreign-key constraints:
    "plays_gid_fkey" FOREIGN KEY (gid) REFERENCES game_sessions(gid) ON UPDATE C
ASCADE ON DELETE RESTRICT
    "plays_username_fkey" FOREIGN KEY (username, inventory_name) REFERENCES inve
ntories(username, name) ON UPDATE CASCADE ON DELETE RESTRICT
```

```
                                    Table "cs421g87.game_sessions"
    Column    |            Type             |                          Modifiers
--------------+-----------------------------+-------------------------------------
-------------------------
 gid          | integer                     | not null default nextval('game_ses
sions_gid_seq'::regclass)
 game_type    | character varying(255)      |
 map_name     | character varying(255)      |
 winning_team | integer                     |
 start_time   | timestamp without time zone | default now()
 end_time     | timestamp without time zone |
Indexes:
    "game_sessions_pkey" PRIMARY KEY, btree (gid)
Referenced by:
    TABLE "plays" CONSTRAINT "plays_gid_fkey" FOREIGN KEY (gid) REFERENCES game_
sessions(gid) ON UPDATE CASCADE ON DELETE RESTRICT
```

```sql
--Active Monthly Players, where an active player is defined as having participated in at
least 1 game_session in that (month, year)
SELECT EXTRACT(year from start_time) as year,
       EXTRACT(month from start_time) as month,
       COUNT( DISTINCT username) as active_users
FROM plays p, game_sessions g
WHERE p.gid = g.gid
GROUP BY (year, month)
ORDER BY year, month
```

	year	month	active_users
1	2019	2	4
2	2019	3	14
3	2019	4	14
4	2019	5	10
5	2019	6	14
6	2019	7	14
7	2019	8	14
8	2019	9	13
9	2019	10	13
10	2019	11	14
11	2019	12	14
12	2020	1	14
13	2020	2	12

6- Data Modification Commands

1. Deleting a Guild

To accomplish this we first set the guild_name and guild_join_date of every member of the guild to null, then deleting the guild in a transaction.

```
BEGIN;
UPDATE players SET guild_name = NULL, guild_join_date = NULL
WHERE guild_name = 'Some Guild';
DELETE FROM guilds WHERE name = 'Some Guild';
COMMIT;
```

Log output after running the transaction

```
sql> BEGIN
[2020-02-28 18:10:08] completed in 26 ms
sql> UPDATE players SET guild_name = NULL, guild_join_date = NULL
     WHERE guild_name = 'Some Guild'
[2020-02-28 18:10:08] 2 rows affected in 21 ms
sql> DELETE FROM guilds WHERE name = 'Some Guild'
[2020-02-28 18:10:08] 1 row affected in 25 ms
sql> COMMIT
[2020-02-28 18:10:08] completed in 21 ms
```

2. Banning a Player

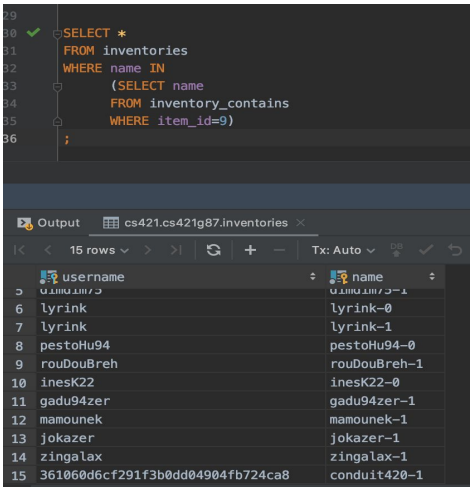
```
UPDATE players SET
 banned_date = now(),
 username = md5(random()::text),
 guild_name = NULL,
 guild_join_date = NULL
WHERE username = 'conduit420';
```

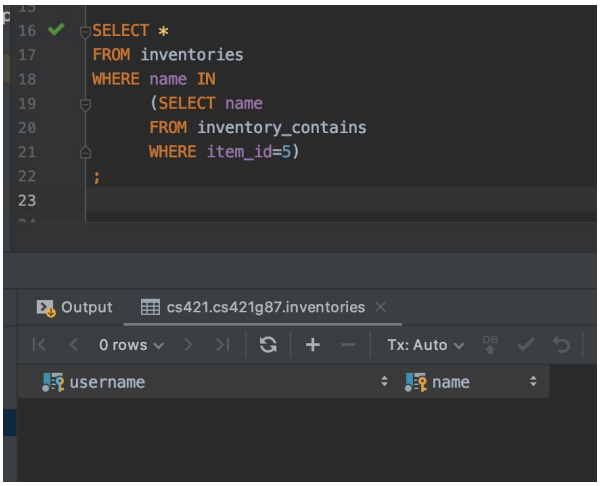
Banning a player is interesting in that since we require keeping logs of purchases associated to a user for auditing purposes (as we are handling monetary transactions), we want to actually keep the user in our database. For all intents and purposes however this user should not exist anymore, and thus we use an md5 hash to randomize the username, and set the banned_date field to indicate that the user is banned. Changing the username causes a lot of cascading updates.

```
sql> BEGIN
[2020-02-28 19:20:15] completed in 20 ms
sql> UPDATE players SET
       banned_date = now(),
       username = md5(random()::text),
       guild_name = NULL,
       guild_join_date = NULL
     WHERE username = 'conduit420'
[2020-02-28 19:20:15] 1 row affected in 22 ms
sql> COMMIT
[2020-02-28 19:20:15] completed in 23 ms
sql> END
[2020-02-28 19:20:16] [25P01] there is no transaction in progress
[2020-02-28 19:20:16] completed in 25 ms
```

3. Modifying all inventories that have a specific item

In this example we want to modify all inventories that have a Grenade Fumigène (id: 9) with a Grenade paralysante (id: 5).
Initially we have 15 inventories with a Grenade Fumigène (id: 9) and 0 with a Grenade paralysante (id: 5)

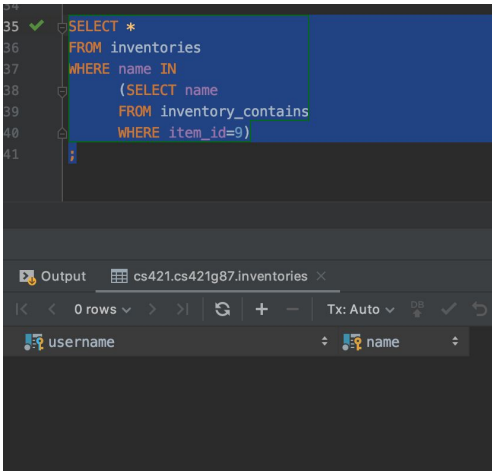


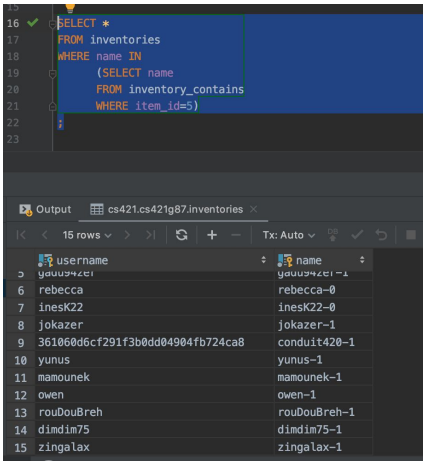


Now let's run the update command (see updates.sql) :

```
UPDATE inventory_contains
   SET item_id = 5
 WHERE item_id = 9;
```

And let's run the previous commands to verify that the change did occur :





Everything worked well.

4. Update coin balance for all players that purchased coins

We sum the coins purchased per player and update coin balance for each

Let's run :

```
UPDATE players
   SET coin_balance = subToSum.total
   FROM(SELECT username, COALESCE(SUM(coins),0) AS total
       FROM coin_purchases GROUP BY username) AS subToSum
WHERE players.username= subToSum.username;
```

We can verify that indeed the coin_balance for each player is the same as the sum of all the transactions they did (knowing that the default coin_balance set was 0)

	username	email	coin_balance	level	register_date
1	mamounek	mamoune.kellal@gmail.com	17900	57	2018-05-25 02:54:26.295636
2	conduit420	shadara@mail.mcgill.ca	0	5	2018-03-18 01:13:38.138708
3	zingalax	romaincharpentier@gmail.com	6100	22	2018-01-24 22:28:49.733603
4	rouDouBreh	richard@yahoo.ca	10300	49	2018-01-27 00:20:49.404473
5	dimdim75	dim@hotmail.fr	6700	55	2018-09-30 11:37:36.939617
6	gabriel	gabriel.alacchi@mail.mcgill.ca	24100	2	2018-09-12 03:21:48.937529
7	inesK22	ines.kheyar@hotmail.fr	10900	57	2018-01-20 15:02:06.584400
8	yunus	yunus@mail.mcgill.ca	36000	13	2018-01-20 20:47:29.240238
9	gadu94zer	gabrielradwan@gmail.com	6600	84	2018-05-14 04:19:24.503339
10	jokazer	arthur.roc@hotmail.fr	15900	47	2018-03-21 18:36:50.485201
11	owen	owen@mail.mcgill.ca	30800	56	2018-02-05 17:28:07.633079
12	lyrink	linus@gmail.com	6300	38	2018-08-19 20:05:13.481845
13	rebecca	rebecca.jaubert@mail.mcgill.ca	13200	5	2018-07-12 03:47:01.593270
14	pestoHu94	paul.huchede@gmail.com	200	82	2018-04-28 11:46:15.513927

```
SELECT username, COALESCE(SUM(coins),0) AS totalCoinsBought
    FROM coin_purchases GROUP BY username;
```

Output | Result 4

12 rows

	username	totalcoinsbought
1	pestoHu94	200
2	owen	30800
3	dimdim75	6700
4	inesK22	10900
5	gadu94zer	6600
6	gabriel	24100
7	rebecca	13200
8	zingalax	6100
9	yunus	36000
10	jokazer	15900
11	rouDouBreh	10300
12	lyrink	6300

7 - Views

1. The Leaderboard with Respect to Win Rate Ratio

```
CREATE OR REPLACE VIEW leaderboard AS
(SELECT
 ps.username,
 count(*) as games_played,
 count(*) filter (where ps.team_number = gs.winning_team) as wins,
 round(count(*) filter (where ps.team_number = gs.winning_team)::NUMERIC /
count(*)::NUMERIC, 2) as win_loss_ratio,
 SUM(kills) as total_kills,
 SUM(deaths) as total_deaths,
 SUM(assists) as total_assists,
```

```
 round(SUM(kills)::NUMERIC / SUM(deaths)::NUMERIC, 2) AS kdr
FROM plays ps
INNER JOIN game_sessions gs ON ps.gid = gs.gid
INNER JOIN players p ON p.username = ps.username
WHERE p.banned_date IS NULL
GROUP BY ps.username
ORDER BY win_loss_ratio DESC);
```

Result of running the query.

```
sql> CREATE OR REPLACE VIEW leaderboard AS
     (SELECT
       ps.username,
       count(*) as games_played,
       count(*) filter (where ps.team_number = gs.winning_team) as wins,
       round(count(*) filter (where ps.team_number = gs.winning_team)::NUMERIC /
count(*)::NUMERIC, 2) as win_loss_ratio,
       SUM(kills) as total_kills,
       SUM(deaths) as total_deaths,
       SUM(assists) as total_assists,
       round(SUM(kills)::NUMERIC / SUM(deaths)::NUMERIC, 2) AS kdr
     FROM plays ps
     INNER JOIN game_sessions gs ON ps.gid = gs.gid
     INNER JOIN players p ON p.username = ps.username
     WHERE p.banned_date IS NULL
     GROUP BY ps.username
     ORDER BY win_loss_ratio DESC)
[2020-02-28 18:20:45] completed in 26 ms
```

```
SELECT * FROM leaderboard LIMIT 10;
```

10 rows DDL

console 38 ms

	username	games_played	wins	win_loss_ratio	total_kills	total_deaths	total_assists	kdr
1	[illegible]	32	26	0.81	415	162	60	2.56
2	gabriel	35	28	0.8	427	187	61	2.28
3	yunus	33	25	0.76	408	182	61	2.24
4	rebecca	33	21	0.64	423	180	52	2.35
5	gadu94zer	37	20	0.54	297	290	61	1.02
6	inesK22	40	21	0.53	336	291	66	1.15
7	pestoHu94	48	22	0.46	303	528	81	0.57
8	rouDouBreh	35	15	0.43	232	360	68	0.64
9	dimdim75	33	14	0.42	279	241	52	1.16
10	zingalax	34	12	0.35	222	387	59	0.57

2. Revenues Aggregated by Month

```
CREATE OR REPLACE VIEW monthly_revenue AS (
SELECT SUM(amount_paid) as revenue, EXTRACT(year from trans_date) as year, EXTRACT(month
from trans_date) as month FROM coin_purchases
GROUP BY year, month
ORDER BY year, month ASC);
```

Result of running the query

```
sql>
     CREATE OR REPLACE VIEW monthly_revenue AS (
```

```
     SELECT SUM(amount_paid) as revenue, EXTRACT(year from trans_date) as year,
EXTRACT(month from trans_date) as month FROM coin_purchases
     GROUP BY year, month
     ORDER BY year, month ASC)
[2020-02-28 18:28:14] completed in 30 ms
```

```
SELECT
 year, month, revenue
FROM monthly_revenue
ORDER BY revenue DESC
LIMIT 1;
```

	year	month	revenue
1	2018	12	546

Conditions For Updating A View:

1. The view is defined based on a single table.

2. The view must include the PRIMARY KEY of the table based upon which the view has been created.

3. The view should not have any field made out of aggregate functions.

4. The view must not have any DISTINCT clause in its definition.

5. The view must not have any GROUP BY or HAVING clause in its definition.

6. The view must not have any SUBQUERIES in its definitions.

7. If the view you want to update is based upon another view, the latter should be updatable.

8. Any of the selected output fields (of the view) must not use constants, strings or value expressions.

Source: https://www.w3resource.com/sql/update-views/sql-update-views.php

8- Checks

See the Modified Relational Schema section for the CHECKS. We show failed queries on the items and plays tables.

Fails price > 0 Check

```
sql> INSERT INTO items (id, name, price, weight) VALUES
     (1000, 'This Will Not Work', -10, 1)
[2020-02-28 20:46:51] [23514] ERROR: new row for relation "items" violates check
constraint "items_price_check"
[2020-02-28 20:46:51] Detail: Failing row contains (1000, This Will Not Work, -10, 1).
```

Failing Assists >= 0 Check

```
sql> INSERT INTO plays (username, inventory_name, gid, kills, deaths, assists,
team_number) VALUES
     ('gabriel', 'gabriel-0', 1, -1, -2, -3, 1)
[2020-02-28 20:50:07] [23514] ERROR: new row for relation "plays" violates check
constraint "plays_assists_check"
[2020-02-28 20:50:07] Detail: Failing row contains (gabriel, gabriel-0, 1, -1, -2, -3,
1).
```

9 - Creativity

Automated data generation

The automation script can be found in the 3 python scripts in the **generation** folder. For most of the data generation we used Python as we wanted to generate a somewhat realistic sample of actual game data that separates skill levels of players. Attached to our submission will be the code we used to accomplish this.

We used python to generate the following

1. Guilds
2. Inventories
3. Games

For guilds, we simply selected the accounts linked to the four of us and made us each admins of our own guild, and randomly selected a balanced number of other players to be additional members of the guild.

For inventories, for each player we randomly created 2 inventories consisting each of a primary, sidearm and melee weapon, a random armor and a random attachment for each item that had one available. Additionally as required we had to automatically create tuples in

the “owns” relationship table to ensure that the items in these inventories are actually owned by the players.

For games we generated data based on tiers of players. We made 4 players top tier, half the remaining players mid tier and bottom tier. We used teams of 2 for each game so that we may create 125 games in total while respecting the 500 rows hard limit set by the requirements of the project. To generate a game we first generate a game session, randomly select 4 players to form teams, and then select game type and map randomly. Finally we randomly select a winning team and kills/deaths, all of that is randomized by a distribution which heavily favors the stronger players, that way our leaderboard query shows an actual variance in the strengths of players.

Complex Analytical Queries:

See Section 5: Queries