G60:
Kexin Fang, 260773389
Pinghsuan Chueh, 260709676
Xiaohui Wang, 260719359
Zhenghua Chen, 260783959
Password: comp421___
(There are three "_"s)

Question 1

Based on the process and the comments provided from the past project, we modified some parts of our relational schema.

Entities:
Users(id, username, password, birthday, district, billing_address, payment_method, email)
Game(game_id, gname, price, id_publisher, id_player) (id_publisher Ref Users, id_player Ref Users)
Versions(game_id, version_num) (game_id Ref Game)
Post(id, time,com_name, game_id, content) (id Ref User, com_name Ref Community, game_id Ref Game)
Community(game_id, com_name, num_mem, create_time, max_admis) (game_id Ref Game)
Activity(activity_name, host_time, rules)

Subclass of community:
High_end_community(game_id, com_name, max_member, min_level) (game_id Ref Game)
VIP_community(game_id, com_name, min_recharge) (game_id Ref Game)

Subclass of User:
Player(id, level, recharge)
Publisher(id, income)

Relations:
friend(user1, user2) (user1 Ref User(id), user2 Ref User(id)) [*Comment: We are concerned about the duplication of friend relationship in the database. In the future implementation, we wish to remove the duplication in the actual database.*]
play(game_id, id, late_play_date) (game_id Ref Game, id Ref User)
send_gift(player1, player2, game_id, send_date) (player1 Ref User(id), player2 Ref User(id), game_id Ref Game)

host(<u>activity_name</u>, <u>host_time</u>, <u>id</u>, game_id) (activity_name Ref Activity, host_time Ref activity, id Ref User, game_id Ref Game)

attend(<u>id</u>, <u>activity_name</u>, <u>host_time</u>, num_attend) (id Ref User, activity_name Ref Activity, host_time Ref activity)

inform(<u>game_id</u>, <u>com_name</u>, <u>id</u>, special_events, special_information) (game_id Ref Game, com_name Ref Community, id Ref User)

administrate(<u>game_id</u>, <u>com_name</u>, <u>id</u>) (game_id Ref Game, com_name Ref Community, id Ref User)

joins(<u>game_id</u>, <u>com_name</u>, <u>id</u>) (game_id Ref Game, com_name Ref Community, id Ref User)

buy(<u>game_id</u>, <u>user_id</u>, order_id, date)(game_id Ref Game, user_id Ref Users)

Question2

```
CREATE TABLE Users(
       id int                  NOT NULL UNIQUE,
       username varchar(30)    NOT NULL UNIQUE,
       password varchar(20)    NOT NULL,
       birthday date           NOT NULL,
       district varchar(100),
       billing_address varchar(100),
       payment_method varchar(30),
       email varchar(50),
       PRIMARY KEY(id)
);

CREATE TABLE Game(
       game_id INT NOT NULL UNIQUE,
       gname VARCHAR(200),
       price INT NOT NULL,
       id_publisher INT UNIQUE,
       id_player INT UNIQUE,
       PRIMARY KEY(game_id),
       FOREIGN KEY(id_publisher)REFERENCES Users(id),
       FOREIGN KEY(id_player)REFERENCES Users(id)
);

CREATE TABLE Versions(
       game_id INT NOT NULL,
       version_num INT NOT NULL,
```

```sql
        PRIMARY KEY(game_id,version_num),
        FOREIGN KEY(game_id)REFERENCES Game(game_id)
);

CREATE TABLE Player(
        id int            NOT NULL UNIQUE,
        level int         NOT NULL,
        recharge  int,
        PRIMARY KEY(id),
        FOREIGN KEY(id) REFERENCES Users(id)
);

CREATE TABLE Publisher(
        id int            NOT NULL UNIQUE,
        income int        NOT NULL,
        PRIMARY KEY(id),
        FOREIGN KEY(id) REFERENCES Users(id)
);

CREATE TABLE Post(
        id INT NOT NULL UNIQUE,
        time TIMESTAMP NOT NULL UNIQUE,
        com_name CHAR(100) NOT NULL UNIQUE,
        game_id INT,
        Content CHAR(200)  NOT NULL,
        PRIMARY KEY(id,time),
        FOREIGN KEY(id)REFERENCES Users(id),
        FOREIGN KEY(com_name)REFERENCES Community(com_name),
        FOREIGN KEY(game_id)REFERENCES Game(game_id)
);

CREATE TABLE Activity(
  activity_name CHAR(100) NOT NULL UNIQUE,
  host_time TIMESTAMP NOT NULL UNIQUE,
 rules VARCHAR(400),
 PRIMARY KEY (activity_name, host_time)
);

CREATE TABLE Community(
  game_id int NOT NULL,
  com_name CHAR(100) NOT NULL UNIQUE,
  num_mem int,
```

```sql
    Create_time DATE NOT NULL,
    max_admis int,
    PRIMARY KEY (game_id, com_name),
    FOREIGN KEY (game_id) REFERENCES Game(game_id)
);

CREATE TABLE VIP_community(
    game_id int NOT NULL,
    com_name CHAR(100) NOT NULL UNIQUE,
    min_recharge int,
    PRIMARY KEY (game_id, com_name),
    FOREIGN KEY (game_id) REFERENCES Game(game_id),
    FOREIGN KEY (com_name) REFERENCES Community(com_name)
);

CREATE TABLE High_end_community(
    game_id int NOT NULL,
    com_name CHAR(100) NOT NULL UNIQUE,
    max_member int,
    min_level int,
    PRIMARY KEY (game_id, com_name),
    FOREIGN KEY (game_id) REFERENCES Game(game_id)
);

CREATE TABLE friend(
    user1 int NOT NULL,
    user2 int NOT NULL,
    PRIMARY KEY (user1,user2),
    FOREIGN KEY (user1) REFERENCES Users(id),
    FOREIGN KEY (user2) REFERENCES Users(id)
);

CREATE TABLE play(
    game_id int NOT NULL,
    id int NOT NULL,
    late_play_date DATE,
    PRIMARY KEY (game_id, id),
    FOREIGN KEY (game_id) REFERENCES Game(game_id),
    FOREIGN KEY (id) REFERENCES Users(id)
);

CREATE TABLE send_gift(
```

```sql
    player1 int NOT NULL,
    player2 int NOT NULL,
    game_id int NOT NULL,
    send_date DATE,
    PRIMARY KEY (player1,player2,game_id),
    FOREIGN KEY (player1) REFERENCES Users(id),
    FOREIGN KEY (player2) REFERENCES Users(id),
    FOREIGN KEY (game_id) REFERENCES Game(game_id)
);

CREATE TABLE joins(
    game_id int NOT NULL,
    com_name varchar(100) NOT NULL,
    id int NOT NULL,
    PRIMARY KEY (game_id,com_name,id),
    FOREIGN KEY (game_id) REFERENCES game(game_id),
    FOREIGN KEY (com_name) REFERENCES community(com_name),
    FOREIGN KEY (id) REFERENCES users(id)
);

CREATE TABLE inform(
    game_id int NOT NULL,
    com_name CHAR(50) NOT NULL,
    id int NOT NULL,
    special_events CHAR(100),
    special_information CHAR(100),
    PRIMARY KEY (game_id, com_name, id),
    FOREIGN KEY (game_id) REFERENCES game(game_id),
    FOREIGN KEY (com_name) REFERENCES community(com_name),
    FOREIGN KEY (id) REFERENCES users(id)
);

CREATE TABLE host(
    activity_name CHAR(100) NOT NULL,
    host_time TIMESTAMP NOT NULL,
    game_id int NOT NULL,
    id int NOT NULL,
    PRIMARY KEY (activity_name, host_time, game_id, id),
    FOREIGN KEY (activity_name) REFERENCES Activity(activity_name),
    FOREIGN KEY (host_time) REFERENCES Activity(host_time),
    FOREIGN KEY (game_id) REFERENCES Game(game_id),
    FOREIGN KEY (id) REFERENCES Users(id)
```

```
);

CREATE TABLE attend(
    id int NOT NULL,
    activity_name CHAR(100) NOT NULL,
    host_time  TIMESTAMP NOT NULL,
    num_attend int,
    PRIMARY KEY (id, activity_name, host_time),
    FOREIGN KEY (id) REFERENCES Users(id),
    FOREIGN KEY (activity_name) REFERENCES Activity(activity_name),
    FOREIGN KEY (host_time) REFERENCES Activity(host_time)
);

CREATE TABLE Administrate(
    game_id INT NOT NULL,
    com_name CHAR(100) NOT NULL,
    id int NOT NULL,
    PRIMARY KEY(game_id,com_name,id),
    FOREIGN KEY(game_id)REFERENCES Game(game_id),
    FOREIGN KEY(com_name)REFERENCES Community(com_name),
    FOREIGN KEY(id)REFERENCES Users(id)
);


CREATE TABLE buy
(
    order_id CHAR(10) NOT NULL,
    date TIMESTAMP NOT NULL,
    game_id INTEGER NOT NULL,
    player_id INTEGER NOT NULL,
    PRIMARY KEY (game_id, player_id),
    UNIQUE (order_id),
    FOREIGN KEY (game_id) REFERENCES game (game_id)
    FOREIGN KEY (player_id) REFERENCES player (id)

);


cs421=> \d activity
          Table "cs421g60.activity"
    Column    |           Type           | Modifiers
--------------+--------------------------+-----------
```

```
 activity_name | character(100)           | not null
 host_time     | timestamp without time zone | not null
 rules         | character varying(400)      |
Indexes:
    "activity_pkey" PRIMARY KEY, btree (activity_name, host_time)
    "activity_activity_name_key" UNIQUE CONSTRAINT, btree (activity_name)
    "activity_host_time_key" UNIQUE CONSTRAINT, btree (host_time)
Referenced by:
    TABLE "attend" CONSTRAINT "attend_activity_name_fkey" FOREIGN KEY
(activity_name) REFERENCES activity(activity_name)
    TABLE "attend" CONSTRAINT "attend_host_time_fkey" FOREIGN KEY
(host_time) REFERENCES activity(host_time)
    TABLE "host" CONSTRAINT "host_activity_name_fkey" FOREIGN KEY
(activity_name) REFERENCES activity(activity_name)
    TABLE "host" CONSTRAINT "host_host_time_fkey" FOREIGN KEY (host_time)
REFERENCES activity(host_time)


cs421=> \d administrate
    Table "cs421g60.administrate"
 Column   |     Type     | Modifiers
----------+--------------+-----------
 game_id  | integer      | not null
 com_name | character(100) | not null
 id       | integer      | not null
Indexes:
    "administrate_pkey" PRIMARY KEY, btree (game_id, com_name, id)
Foreign-key constraints:
    "administrate_com_name_fkey" FOREIGN KEY (com_name) REFERENCES
community(com_name)
    "administrate_game_id_fkey" FOREIGN KEY (game_id) REFERENCES
game(game_id)
    "administrate_id_fkey" FOREIGN KEY (id) REFERENCES users(id)


cs421=> \d attend
          Table "cs421g60.attend"
   Column     |          Type          | Modifiers
--------------+------------------------+-----------
 id           | integer                | not null
 activity_name | character(100)         | not null
 host_time    | timestamp without time zone | not null
```

num_attend | integer |
Indexes:
    "attend_pkey" PRIMARY KEY, btree (id, activity_name, host_time)
Foreign-key constraints:
    "attend_activity_name_fkey" FOREIGN KEY (activity_name) REFERENCES
activity(activity_name)
    "attend_host_time_fkey" FOREIGN KEY (host_time) REFERENCES
activity(host_time)
    "attend_id_fkey" FOREIGN KEY (id) REFERENCES users(id)

cs421=> \d buy
          Table "cs421g60.buy"
 Column | Type | Modifiers
-----------+----------------------------+-----------
 order_id | character(10) | not null
 date | timestamp without time zone | not null
 game_id | integer | not null
 player_id | integer | not null
Indexes:
    "buy_pkey" PRIMARY KEY, btree (game_id, player_id)
    "order id" UNIQUE CONSTRAINT, btree (order_id)
Foreign-key constraints:
    "buy_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
NOT VALID
    "player_id" FOREIGN KEY (player_id) REFERENCES player(id)

cs421=> \d community
      Table "cs421g60.community"
  Column | Type | Modifiers
-------------+----------------+-----------
 game_id | integer | not null
 com_name | character(100) | not null
 num_mem | integer |
 create_time | date | not null
 max_admis | integer |
Indexes:
    "community_pkey" PRIMARY KEY, btree (game_id, com_name)
    "community_com_name_key" UNIQUE CONSTRAINT, btree (com_name)
Foreign-key constraints:

"community_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
Referenced by:
    TABLE "administrate" CONSTRAINT "administrate_com_name_fkey" FOREIGN KEY (com_name) REFERENCES community(com_name)
    TABLE "inform" CONSTRAINT "inform_com_name_fkey" FOREIGN KEY (com_name) REFERENCES community(com_name)
    TABLE "joins" CONSTRAINT "joins_com_name_fkey" FOREIGN KEY (com_name) REFERENCES community(com_name)
    TABLE "post" CONSTRAINT "post_com_name_fkey" FOREIGN KEY (com_name) REFERENCES community(com_name)
    TABLE "vip_community" CONSTRAINT "vip_community_com_name_fkey" FOREIGN KEY (com_name) REFERENCES community(com_name)

cs421=> \d friend
   Table "cs421g60.friend"
 Column | Type | Modifiers
--------+---------+-----------
 user1 | integer | not null
 user2 | integer | not null
Indexes:
    "friend_pkey" PRIMARY KEY, btree (user1, user2)
Foreign-key constraints:
    "friend_user1_fkey" FOREIGN KEY (user1) REFERENCES users(id)
    "friend_user2_fkey" FOREIGN KEY (user2) REFERENCES users(id)

cs421=> \d game
         Table "cs421g60.game"
   Column | Type | Modifiers
-------------+-----------------------+-----------
 game_id | integer | not null
 gname | character varying(200) |
 price | integer | not null
 id_publisher | integer |
Indexes:
    "game_pkey" PRIMARY KEY, btree (game_id)
Foreign-key constraints:
    "game_id_publisher_fkey" FOREIGN KEY (id_publisher) REFERENCES users(id)
Referenced by:

TABLE "administrate" CONSTRAINT "administrate_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "buy" CONSTRAINT "buy_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id) NOT VALID
    TABLE "community" CONSTRAINT "community_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "high_end_community" CONSTRAINT "high_end_community_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "host" CONSTRAINT "host_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "inform" CONSTRAINT "inform_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "joins" CONSTRAINT "joins_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "play" CONSTRAINT "play_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "post" CONSTRAINT "post_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "versions" CONSTRAINT "versions_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    TABLE "vip_community" CONSTRAINT "vip_community_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)

cs421=> \d high_end_community
  Table "cs421g60.high_end_community"
  Column | Type | Modifiers
------------+---------------+-----------
 game_id | integer | not null
 com_name | character(100) | not null
 max_member | integer |
 min_level | integer |
Indexes:
    "high_end_community_pkey" PRIMARY KEY, btree (game_id, com_name)
    "high_end_community_com_name_key" UNIQUE CONSTRAINT, btree (com_name)
Foreign-key constraints:
    "high_end_community_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)

```
cs421=> \d host
           Table "cs421g60.host"
   Column     |           Type            | Modifiers
---------------+---------------------------+-----------
 activity_name | character(100)            | not null
 host_time     | timestamp without time zone | not null
 game_id       | integer                   | not null
 id            | integer                   | not null
Indexes:
    "host_pkey" PRIMARY KEY, btree (activity_name, host_time, game_id, id)
Foreign-key constraints:
    "host_activity_name_fkey" FOREIGN KEY (activity_name) REFERENCES
activity(activity_name)
    "host_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    "host_host_time_fkey" FOREIGN KEY (host_time) REFERENCES
activity(host_time)
    "host_id_fkey" FOREIGN KEY (id) REFERENCES users(id)


cs421=> \d inform
         Table "cs421g60.inform"
     Column        |      Type      | Modifiers
---------------------+----------------+-----------
 game_id             | integer        | not null
 com_name            | character(50)  | not null
 id                  | integer        | not null
 special_events      | character(100) |
 special_information | character(100) |
Indexes:
    "inform_pkey" PRIMARY KEY, btree (game_id, com_name, id)
Foreign-key constraints:
    "inform_com_name_fkey" FOREIGN KEY (com_name) REFERENCES
community(com_name)
    "inform_game_id_fkey" FOREIGN KEY (game_id) REFERENCES
game(game_id)
    "inform_id_fkey" FOREIGN KEY (id) REFERENCES users(id)


cs421=> \d joins
        Table "cs421g60.joins"
 Column  |          Type           | Modifiers
----------+-------------------------+-----------
```

```
 game_id  | integer              | not null
 com_name | character varying(100) | not null
 id       | integer              | not null
Indexes:
    "joins_pkey" PRIMARY KEY, btree (game_id, com_name, id)
Foreign-key constraints:
    "joins_com_name_fkey" FOREIGN KEY (com_name) REFERENCES
community(com_name)
    "joins_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    "joins_id_fkey" FOREIGN KEY (id) REFERENCES users(id)


cs421=> \d play
      Table "cs421g60.play"
    Column    |  Type  | Modifiers
----------------+---------+-----------
 game_id        | integer | not null
 id             | integer | not null
 late_play_date | date    |
Indexes:
    "play_pkey" PRIMARY KEY, btree (game_id, id)
Foreign-key constraints:
    "play_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    "play_id_fkey" FOREIGN KEY (id) REFERENCES users(id)


cs421=> \d player
      Table "cs421g60.player"
 Column  |      Type       | Modifiers
----------+------------------+-----------
 id       | integer         | not null
 level    | integer         | not null
 recharge | double precision |
Indexes:
    "player_pkey" PRIMARY KEY, btree (id)
Foreign-key constraints:
    "player_id_fkey" FOREIGN KEY (id) REFERENCES users(id)
Referenced by:
    TABLE "buy" CONSTRAINT "player_id" FOREIGN KEY (player_id)
REFERENCES player(id)
```

```
cs421=> \d post
         Table "cs421g60.post"
 Column  |           Type            | Modifiers
---------+---------------------------+-----------
 id       | integer                  | not null
 time     | timestamp without time zone | not null
 com_name | character(100)           | not null
 game_id  | integer                  |
 content  | character(200)           | not null
Indexes:
    "post_pkey" PRIMARY KEY, btree (id, "time")
    "post_com_name_key" UNIQUE CONSTRAINT, btree (com_name)
    "post_id_key" UNIQUE CONSTRAINT, btree (id)
    "post_time_key" UNIQUE CONSTRAINT, btree ("time")
Foreign-key constraints:
    "post_com_name_fkey" FOREIGN KEY (com_name) REFERENCES
community(com_name)
    "post_game_id_fkey" FOREIGN KEY (game_id) REFERENCES game(game_id)
    "post_id_fkey" FOREIGN KEY (id) REFERENCES users(id)


cs421=> \d publisher
      Table "cs421g60.publisher"
 Column |      Type        | Modifiers
--------+------------------+-----------
 id     | integer          | not null
 income | double precision | not null
Indexes:
    "publisher_pkey" PRIMARY KEY, btree (id)
Foreign-key constraints:
    "publisher_id_fkey" FOREIGN KEY (id) REFERENCES users(id)


cs421=> \d send_gift
   Table "cs421g60.send_gift"
 Column    | Type    | Modifiers
-----------+---------+-----------
 player1   | integer | not null
 player2   | integer | not null
 game_id   | integer | not null
 send_date | date    |
Indexes:
```

"send_gift_pkey" PRIMARY KEY, btree (player1, player2, game_id)
Foreign-key constraints:
    "send_gift_player1_fkey" FOREIGN KEY (player1) REFERENCES users(id)
    "send_gift_player2_fkey" FOREIGN KEY (player2) REFERENCES users(id)
    "send_gift_game_id_fkey" FOREIGN KEY (game_id) REFERENCES
game(game_id)

cs421=> \d users
            Table "cs421g60.users"
    Column | Type | Modifiers
----------------+--------------------+-----------
 id | integer | not null
 username | character varying(30) | not null
 password | character varying(20) | not null
 birthday | date | not null
 district | character varying(100) |
 billing_address | character varying(100) |
 payment_method | character varying(30) |
 email | character varying(50) |
Indexes:
    "users_pkey" PRIMARY KEY, btree (id)
    "users_id_key" UNIQUE CONSTRAINT, btree (id)
    "users_username_key" UNIQUE CONSTRAINT, btree (username)
Referenced by:
    TABLE "administrate" CONSTRAINT "administrate_id_fkey" FOREIGN KEY (id)
REFERENCES users(id)
    TABLE "attend" CONSTRAINT "attend_id_fkey" FOREIGN KEY (id)
REFERENCES users(id)
    TABLE "friend" CONSTRAINT "friend_user1_fkey" FOREIGN KEY (user1)
REFERENCES users(id)
    TABLE "friend" CONSTRAINT "friend_user2_fkey" FOREIGN KEY (user2)
REFERENCES users(id)
    TABLE "game" CONSTRAINT "game_id_publisher_fkey" FOREIGN KEY
(id_publisher) REFERENCES users(id)
    TABLE "host" CONSTRAINT "host_id_fkey" FOREIGN KEY (id) REFERENCES
users(id)
    TABLE "inform" CONSTRAINT "inform_id_fkey" FOREIGN KEY (id)
REFERENCES users(id)
    TABLE "joins" CONSTRAINT "joins_id_fkey" FOREIGN KEY (id) REFERENCES
users(id)
    TABLE "play" CONSTRAINT "play_id_fkey" FOREIGN KEY (id) REFERENCES
users(id)

TABLE "player" CONSTRAINT "player_id_fkey" FOREIGN KEY (id)
REFERENCES users(id)
    TABLE "post" CONSTRAINT "post_id_fkey" FOREIGN KEY (id) REFERENCES
users(id)
    TABLE "publisher" CONSTRAINT "publisher_id_fkey" FOREIGN KEY (id)
REFERENCES users(id)
    TABLE "send_gift" CONSTRAINT "send_gift_player1_fkey" FOREIGN KEY
(player1) REFERENCES users(id)
    TABLE "send_gift" CONSTRAINT "send_gift_player2_fkey" FOREIGN KEY
(player2) REFERENCES users(id)
   TABLE "send_gift" CONSTRAINT "send_gift_game_id_fkey" FOREIGN KEY
(game_id) REFERENCES game(game_id)

cs421=> \d versions
    Table "cs421g60.versions"
  Column | Type | Modifiers
-------------+---------+-----------
 game_id | integer | not null
 version_num | integer | not null
Indexes:
   "versions_pkey" PRIMARY KEY, btree (game_id, version_num)
Foreign-key constraints:
   "versions_game_id_fkey" FOREIGN KEY (game_id) REFERENCES
game(game_id)

cs421=> \d vip_community
      Table "cs421g60.vip_community"
   Column | Type | Modifiers
-------------+----------------------+-----------
 game_id | integer | not null
 com_name | character varying(50) | not null
 min_recharge | numeric |
Indexes:
   "vip_community_pkey" PRIMARY KEY, btree (game_id)
   "vip_community_com_name_key" UNIQUE CONSTRAINT, btree (com_name)
Foreign-key constraints:
   "vip_community_com_name_fkey" FOREIGN KEY (com_name) REFERENCES
community(com_name)
   "vip_community_game_id_fkey" FOREIGN KEY (game_id) REFERENCES
game(game_id)

Question 3

INSERT INTO Player VALUES (12, 2, 100); -- success
INSERT INTO users VALUES (12345, 'AAAAA BBBBB', 'password', '1979-01-01', 'French', '2999 Avenue Path', 'PAW1234567890', 'aaaaa@gmail.com');

---------------Mavis' Insert--------------
DELETE FROM player;
INSERT INTO player VALUES(16576,1,500);
INSERT INTO player VALUES(12156,3,500);
INSERT INTO player VALUES(11182,11,500);
INSERT INTO player VALUES(13323,67,500);
INSERT INTO player VALUES(12279,20,500);

Printout:
```
 id       | level     | recharge
----------+-----------+----------
 16576 |      1 |   500
 12156 |      3 |   500
 11182 |     11 |   500
 13323 |     67 |   500
 12279 |     20 |   500
(5 rows)
```

-------------------Mavis' insert end--------------------

Question 4

Data imported total count (produced by python, please see the attachment for codes)

Table name	# columns	# rows
users	8	150
player	3	50

publisher	2	30
community	5	15
high_end_community	4	5
vip_community	3	5
friend	2	25
game	5	10
activity	3	10
administrate	3	21
attend	4	100
buy	4	20
host	4	10
inform	5	5
joins	3	20
post	5	10
send_gift	3	10
versions	2	29
play	3	

Screenshots for the first 5-10 tuples for each table:

Users

Data Output   Explain   Messages   Notifications

	id [PK] integer	username character varying (30)	password character varying	birthday date	district character varying (100)	billing_address character varying (100)	payment_method character varying (30)	email character varying (50)
1	16576	Randall Carrillo	pnygkiuiuz	1970-04-02	New Zealand	663 William Plains Apt. 266	BTTA09995321921221	whancock@hotmail.com
2	12156	Lori Allen	urjsfftwjq	2019-01-03	Ukraine	075 Rick Orchard Apt. 656	UIFH46726296210933	christopherbryant@hotmail...
3	11182	Christian Ray	qqxnkhkybu	2017-10-01	New Caledonia	983 Stephanie Mill	TAME68674745822169	csantos@yahoo.com
4	13323	Christopher Willis	xmzzvtsbgw	1991-09-30	Jordan	86239 Michelle Loop	LOIS81702483395843	alec01@hotmail.com
5	12279	Elizabeth Oliver	bvgidirawc	1997-09-04	Trinidad and Tobago	369 Horton Junctions	NYJI97203857610270	lpollard@gmail.com
6	10798	Donald Wilkerson	didxfjszxa	1975-10-07	Maldives	088 Amanda Burgs	NENK78335249029736	jblake@gmail.com
7	11743	Chelsea Simpson	igxsfzrwlq	1979-01-01	Costa Rica	4905 Porter Stream	DVQV47808616158346	andrew57@hotmail.com
8	19962	Norma Hamilton	dxorknszka	1993-06-24	Netherlands Antilles	660 Eric Islands	JNWK17997275723088	kenneth57@gmail.com
9	19233	Roberta Clark	yocsdatxsg	1973-04-23	Belarus	Unit 6423 Box 8778	FMGO066810091730291	kyletorres@hotmail.com
10	12486	George Kidd	eyineauvoc	2002-10-21	Myanmar	37056 Jones Way Suite 770	QHVL40944414980061	mlara@gmail.com

Player

	id [PK] integer	level integer	recharge double precision
1	10720	14	4551.78
2	15488	98	4133.62
3	16322	43	936.66
4	15874	54	2777.57
5	11782	96	4045.94
6	15240	74	4581.66
7	17450	77	9115.33
8	14416	29	7333.65
9	11668	38	7791.39
10	12892	26	932.82

Publisher

	id [PK] integer	income double precision
1	14689	69041.16
2	10275	91674.19
3	11209	66868.16
4	11817	66865.89
5	17161	76951.5
6	14543	63976.81
7	10671	48502.2
8	19377	18938.4
9	15155	51447.45
10	18291	52736.99

Community

	game_id [PK] integer	com_name [PK] character (100)	num_mem integer	create_time date	max_admis integer
1	31088	Let's Minecraft ...	500	2019-11-01	5
2	39917	GTA Forum ...	500	2018-10-18	4
3	39917	Grand Theft Auto Commu...	160	2017-05-12	1
4	37132	Tetris Community ...	200	2016-09-20	2
5	38697	The Skyrim ...	150	2019-07-12	2
6	34385	Chez Sims ...	100	2019-10-07	1
7	32131	ZZZelda ...	60	2018-06-14	1
8	32131	Zelda Players ...	120	2018-03-02	2
9	39206	Phantom Pain Legend ...	50	2019-11-27	2
10	38195	NFL Talk ...	100	2015-06-20	3

High_end_community

	game_id [PK] integer	com_name [PK] character (100)	max_member integer	min_level integer
1	39917	Grand Theft Auto Commu...	160	32
2	37132	Tetris Community ...	200	25
3	34466	Mario Discovery ...	300	12
4	34385	Chez Sims ...	100	10
5	39206	Phantom Pain Legend ...	50	40

Vip_community

	game_id [PK] integer	com_name character varying (50)	min_recharge numeric
1	38697	The Skyrim	67
2	38195	NFL Talk	26
3	32131	ZZZelda	96
4	32551	The A-spec VIPS	49
5	34466	Mario Labo	19

Friend

	user1 [PK] integer	user2 [PK] integer
1	10720	10880
2	16738	15874
3	12452	18084
4	10342	19494
5	17832	16038
6	17322	13896
7	18630	17754
8	13136	11478
9	18902	15288
10	12634	12828

Game

	game_id [PK] integer	gname character varying (200)	price integer	id_publisher integer
1	32551	Gran Turismo 3: A-Spec	89	12892
2	31088	Minecraft	30	14689
3	39917	Grand Theft Auto V	15	10275
4	37132	Tetris	66	11209
5	38697	The Elder Scrolls V: Skyrim	45	11817
6	34466	Mario Bros	78	17161
7	34385	The Sims	51	14543
8	32131	The Legend of Zelda	17	10671
9	39206	Metal Gear Solid V: The Phant...	18	19377
10	38195	Madden NFL 2003	19	15155

Activity

	activity_name [PK] character (100)	host_time [PK] timestamp without time zone	rules character varying (400)
1	National Video Game Chal...	2020-02-28 18:00:00	Complete several missions with your team!
2	Super Mario Party Obstacl...	2020-03-11 11:00:00	Run with these fun DIY ideas to bring Mario-worthy obstacles into your backyard
3	Minecraft the New World ...	2020-01-11 10:00:00	For new designs, 2 hours to bake, 20 minutes to decorate
4	Tearaway Papercraft - Mar...	2019-01-21 10:00:00	In limited time, tear the object into as many pieces as you can
5	Make a Video Game Video...	2019-03-15 14:00:00	Get creative and act out your favorite games!
6	Pirate Escape Raft - GTA ...	2020-04-09 14:00:00	Use a zipper baggie as a flotation device assisting a popsicle stick raft to float until the seas get too rough
7	Game With Anna and Elsa ...	2020-02-01 16:00:00	Learn to ace the special game with characters Anna and Elsa
8	Mario game art ...	2020-11-01 09:00:00	use 1×1 bricks as a pixel to illustrate your creative ideas
9	Diamond Hunting - Mario! ...	2020-07-14 10:00:00	Given 4 hours, players find as many diamonds as they can in teams!
10	Zelda Puzzle Night ...	2020-09-24 20:00:00	See which teams solve the more puzzles in 2 hours!

Administrate

	game_id [PK] integer	com_name [PK] character (100)	id [PK] integer
1	31088	Let's Minecraft	16322
2	31088	Let's Minecraft	11782
3	31088	Let's Minecraft	17450
4	39917	GTA Forum	11668
5	39917	GTA Forum	17832
6	39917	Grand Theft Auto Community	17713
7	37132	Tetris Community	11909
8	38697	The Skyrim	11727
9	34466	Mario Bros' Farm	12729
10	34466	Mario Discovery	12828

Attend

	id [PK] integer	activity_name [PK] character (100)	host_time [PK] timestamp without time zone	num_attend integer
1	17605	National Video Game Chal...	2020-02-28 18:00:00	135
2	19054	National Video Game Chal...	2020-02-28 18:00:00	135
3	11012	National Video Game Chal...	2020-02-28 18:00:00	135
4	13924	National Video Game Chal...	2020-02-28 18:00:00	135
5	10227	National Video Game Chal...	2020-02-28 18:00:00	135
6	18511	National Video Game Chal...	2020-02-28 18:00:00	135
7	16836	National Video Game Chal...	2020-02-28 18:00:00	135
8	15155	National Video Game Chal...	2020-02-28 18:00:00	135
9	16089	National Video Game Chal...	2020-02-28 18:00:00	135
10	14674	National Video Game Chal...	2020-02-28 18:00:00	135

Buy

	order_id character (10)	date timestamp without time zone	game_id [PK] integer	player_id [PK] integer
1	jmyxx69819	2015-10-27 18:15:28	38697	10720
2	cbete19616	2007-10-20 16:03:43	34466	10720
3	hhiur91832	2008-03-10 07:52:12	39206	10720
4	wlxoc49383	2019-09-07 22:23:11	39917	10720
5	oamvl97882	1995-10-01 07:25:08	32551	10720
6	odmyc45680	1995-05-01 07:00:37	32131	16576
7	ggcpz40621	1987-12-29 00:55:00	31088	17176
8	cxqgj89962	1970-09-27 12:09:32	39206	19466
9	lacry64098	1990-11-11 12:40:08	39206	15240
10	bqrdq59492	1997-01-16 08:15:55	39917	16576

Host

	activity_name [PK] character (100)	host_time [PK] timestamp without time zone	game_id [PK] integer	id [PK] integer
1	National Video Game Chal...	2020-02-28 18:00:00	39206	15488
2	Super Mario Party Obstacl...	2020-03-11 11:00:00	34466	12729
3	Minecraft the New World ...	2020-01-11 10:00:00	31088	11782
4	Tearaway Papercraft - Mar...	2019-01-21 10:00:00	34466	15240
5	Make a Video Game Video...	2019-03-15 14:00:00	39917	11668
6	Pirate Escape Raft - GTA ...	2020-04-09 14:00:00	39917	12892
7	Game With Anna and Elsa ...	2020-02-01 16:00:00	34385	10556
8	Mario game art ...	2020-11-01 09:00:00	34466	14465
9	Diamond Hunting - Mario! ...	2020-07-14 10:00:00	34466	12729
10	Zelda Puzzle Night ...	2020-09-24 20:00:00	32131	15161

Inform

game_id [PK] integer	com_name [PK] character (50)		id [PK] integer	special_events character (100)	special_information character (100)
1	38697	The Skyrim	...	15874	off-line meeting in P...	[null]
2	38195	NFL Talk	...	11668	off-line meeting in Sa...	Dress fancy and come da...
3	32551	The A-spec VIPS	...	12892	off-line meeting in To...	audience advised to have ...
4	34466	Mario Labo	...	11782	off-line meeting in N...	[null]
5	32131	ZZZelda	...	17450	video meeting with p...	[null]

Joins

	game_id [PK] integer	com_name [PK] character varying (100)	id [PK] integer
1	31088	Let's Minecraft	16576
2	39917	GTA Forum	12156
3	39917	Grand Theft Auto Community	11182
4	37132	Tetris Community	13323
5	38697	The Skyrim	12279
6	34466	Mario Bros' Farm	10798
7	34466	Mario Discovery	11743
8	34385	Chez Sims	19962
9	32131	ZZZelda	19233
10	39206	Phantom Pain Legend	12486

Post

	id [PK] integer	time [PK] timestamp without time zone	com_name character (100)		game_id integer	content character (200)
1	16576	2017-10-01 04:12:52	Let's Minecraft	...	31088	Hello i'm new here! A...
2	12156	1987-05-16 21:04:35	GTA Forum	...	39917	Anyone know where ...
3	11182	2016-07-29 02:45:42	Grand Theft Auto Co...		39917	How to pass the final...
4	13323	2014-01-31 16:26:40	Tetris Community	...	37132	Memes about Tetris l...
5	12279	1972-07-26 14:23:50	The Skyrim	...	38697	The Skyrim sucks ...
6	10798	1987-11-25 13:57:34	Mario Bros' Farm	...	34466	This character is soo...
7	11743	1985-08-23 09:54:10	Mario Discovery	...	34466	Would love to make t...
8	19962	1984-09-28 13:36:59	Chez Sims	...	34385	Chez Sims is boring.....
9	19233	2001-12-26 21:04:29	ZZZelda	...	32131	9 tips for mastering t...
10	12486	2002-08-23 19:02:59	Phantom Pain Legen...		39206	My favorite game!!! ...

Send_gift

	player1 [PK] integer	player2 [PK] integer	game_id [PK] integer	send_date date
1	19776	18948	31088	2017-12-30
2	11012	19494	39917	2013-02-11
3	18084	11782	37132	2005-12-18
4	17832	17322	38697	2015-02-01
5	17754	11182	34466	2012-03-12
6	12402	19962	34385	2001-10-25
7	18902	11478	32131	2009-05-04
8	17944	15030	39206	2016-09-14
9	15288	12758	38195	2001-08-22
10	12892	17916	32551	2003-03-11

Versions

	game_id [PK] integer	version_num [PK] integer
1	31088	3
2	39917	5
3	37132	1
4	38697	4
5	34466	3
6	34385	5
7	32131	3
8	39206	2
9	38195	3
10	32551	2

Play

	game_id [PK] integer	id [PK] integer	late_play_date date
1	38697	10720	2019-04-21
2	34466	10720	2019-02-20
3	39206	10720	2019-07-08
4	39917	10720	2019-02-16
5	32551	10720	2019-10-25
6	32131	16576	2019-01-23
7	31088	17176	2019-07-31
8	39206	19466	2019-04-06
9	39206	15240	2019-03-30
10	39917	16576	2019-12-25

Question 5

1. Which players have been informed by game publishers of special events (only players in VIP communities have this privilege)?

Select player_id,i.com_name
From inform i JOIN
(
  Select j.id as player_id, v.com_name
  From vip_community v JOIN joins j
  On v.com_name = j.com_name
)x
On i.com_name = x.com_name
;

Data Output   Explain   Messages

	player_id integer	com_name character (50)	
1	12279	The Skyrim	...
2	19233	ZZZelda	...
3	12156	NFL Talk	...
4	10671	Mario Labo	...
5	15327	ZZZelda	...
6	11209	The A-spec VIPS	...
7	19962	NFL Talk	...
8	19391	The Skyrim	...

2.1. What games did the player who bought the maximum number of games bought?

SELECT gname, game_id FROM game WHERE game_id IN
(SELECT game_id FROM buy WHERE player_id IN (
(SELECT player_id FROM
(SELECT player_id, COUNT(player_id) as t FROM buy GROUP BY player_id
ORDER BY t DESC) AS tmp LIMIT 1)));

	gname character varying (200)	game_id [PK] integer	
1	Gran Turismo 3: A-Spec	32551	
2	Mario Bros	34466	
3	The Elder Scrolls V: Skyrim	38697	
4	Metal Gear Solid V: The Phant...	39206	
5	Grand Theft Auto V	39917	

2.2 What activity did this particular player attend?

SELECT id, activity_name FROM attend WHERE id IN
(SELECT player_id FROM
(SELECT player_id, COUNT(player_id) as t FROM buy GROUP BY player_id
ORDER BY t DESC) AS tmp LIMIT 1);

	id integer	activity_name character (100)	
1	10720	Pirate Escape Raft - GTA	...

3.The oldest player on this website
SELECT id,username,birthday
FROM users
WHERE birthday = (SELECT MIN(birthday) FROM users)

	id [PK] integer	username character varying (30)	birthday date
1	12402	Dr. Joan Schmidt	1970-03-23

3.5 The games bought by the people who are under 30 years old, and the times of buying
SELECT buy.game_id, game.gname, count(buy.game_id) AS times
FROM buy
JOIN users ON id = player_id
JOIN game ON game.game_id = buy.game_id
WHERE birthday<='1990-01-01'
GROUP BY buy.game_id, game.gname
ORDER BY times DESC;

	game_id integer	gname character varying (200)	times bigint
1	39917	Grand Theft Auto V	2
2	39206	Metal Gear Solid V: The Phant...	2
3	34466	Mario Bros	1
4	37132	Tetris	1
5	38697	The Elder Scrolls V: Skyrim	1
6	32131	The Legend of Zelda	1

4. Activities about Mario game that happen after 2019-7-10?
SELECT activity_name FROM
host JOIN game ON host.game_id=game.game_id
WHERE gname='Mario Bros' and
host_time > '2019-07-10 00:00:00';

	activity_name character (100)
1	Super Mario Party Obstacle
2	Mario game art
3	Diamond Hunting - Mario!

5. Activities that are hosted by administrator of community?

SELECT activity_name FROM
host JOIN administrate ON
host.id=administrate.id;

	activity_name character (100)
1	Super Mario Party Obstacle
2	Minecraft the New World
3	Make a Video Game Video - GTA
4	Diamond Hunting - Mario!
5	Zelda Puzzle Night

6. Users befriend each other?

SELECT f1 FROM
friend f1 JOIN friend f2
ON f1.user2=f2.user1 AND f1.user1=f2.user2;

	f1 friend
1	(10720,10880)
2	(10880,10720)

7. The most generous sender (The player who spent most money on gifts)

SELECT * FROM
(SELECT player1 AS sender, SUM(price) AS sumSend FROM
send_gift s JOIN game g ON s.game_id=g.game_id
GROUP BY player1
ORDER BY sumSend DESC
) AS tmp
LIMIT 1;

	sender integer	sumsend bigint
1	17832	137

Question 6

1.Delete the friendship between two users if one of the users decides to unfriend another. For example, user 10720 decides to unfriend user 10880, there are two tuples deleted.

Before:

	user1 [PK] integer	user2 [PK] integer
1	10720	10880
2	16738	15874
3	12452	18084
4	10342	19494
5	17832	16038
6	17322	13896
7	18630	17754
8	13136	11478
9	18902	15288
10	12634	12828

DELETE FROM friend WHERE (user1 = 10720 and user2 = 10880) or (user1 = 10880 and user2=10720);

After:

	user1 [PK] integer	user2 [PK] integer
1	16738	15874
2	12452	18084
3	10342	19494
4	17832	16038
5	17322	13896
6	18630	17754
7	13136	11478
8	18902	15288
9	12634	12828
10	10880	11182

2. If a player recharges money larger than 5000.00, he will become a 'prime' user. We add a column called 'prime' to the table 'player'.

Before:

	id [PK] integer	level integer	recharge double precision
1	10720	14	4551.78
2	15488	98	4133.62
3	16322	43	936.66
4	15874	54	2777.57
5	11782	96	4045.94
6	15240	74	4581.66
7	17450	77	9115.33
8	14416	29	7333.65
9	11668	38	7791.39
10	12892	26	932.82

UPDATE player
SET prime = True WHERE recharge >= 5000;

UPDATE player
SET prime = False WHERE recharge < 5000;

After:
Some players are prime, and the others are not (we only showed):

	id [PK] integer	level integer	recharge double precision	prime boolean
1	17450	77	9115.33	true
2	14416	29	7333.65	true
3	11668	38	7791.39	true
4	16836	13	5597.49	true
5	15842	73	6260.14	true
45	17064	30	3850.51	false
46	10330	81	1470.94	false
47	15726	85	3950.3	false
48	16958	83	4138.6	false
49	10880	100	1473.37	false
50	14674	51	2554.13	false

3. Updating prices in the game table when there is a discount (70% off) on games whose original prices are over 55.

Before:

	game_id [PK] integer	gname character varying (200)	price integer	id_publisher integer
1	32551	Gran Turismo 3: A-Spec	90	12892
2	31088	Minecraft	30	14689
3	39917	Grand Theft Auto V	17	10275
4	37132	Tetris	67	11209
5	38697	The Elder Scrolls V: Skyrim	47	11817
6	34466	Mario Bros	77	17161
7	34385	The Sims	50	14543
8	32131	The Legend of Zelda	17	10671
9	39206	Metal Gear Solid V: The Phant...	17	19377
10	38195	Madden NFL 2003	20	15155

UPDATE game SET price = (price * 0.3) WHERE price > 55;

After:

Data Output   Explain   Messages

	game_id [PK] integer	gname character varying (200)	price integer	id_publisher integer
1	32551	Gran Turismo 3: A-Spec	27	12892
2	37132	Tetris	20	11209
3	34466	Mario Bros	23	17161
4	31088	Minecraft	30	14689
5	39917	Grand Theft Auto V	17	10275
6	38697	The Elder Scrolls V: Skyrim	47	11817
7	34385	The Sims	50	14543
8	32131	The Legend of Zelda	17	10671
9	39206	Metal Gear Solid V: The Phant...	17	19377
10	38195	Madden NFL 2003	20	15155

4.Update the last_play_date after the player played the game
Before:
SELECT * FROM play
WHERE game_id = 31088 AND id = 17176;

	game_id [PK] integer	id [PK] integer	late_play_date date
1	31088	17176	2019-07-31

Update the date:
UPDATE play SET late_play_date='2020-02-28'
WHERE game_id = 31088 AND id = 17176;
After:

	game_id [PK] integer	id [PK] integer	late_play_date date
1	31088	17176	2020-02-28

Question 7

There are conditions that must hold to allow a view update:
- View is defined based on only one table;
- View should include the primary key of the table based on which view is created;
- View should not have references to columns of other base tables;
- View should not contain fields made out of aggregate functions;
- View should not contain GROUP BY, Union or HAVING clause in definition.

Creating a view for activities with details between 2020-01-01 to 2020-05-30.

SQL query:
CREATE VIEW Activity_VIEW AS

SELECT activity_name, host_time, rules
FROM Activity
WHERE activity.host_time BETWEEN '2020-01-01 00:00:00' AND '2020-05-30 23:59:59';

SELECT * from Activity_VIEW;

Output:

Try UPDATE:

We tried to update the 'rules' for the activity 'Super Mario Party Obstacle'. Next we tried to update the activity host time, assuming that the activity is delayed. The first example shows that this particular update in view is automatically translated into the base table. The second example shows that the system does not automatically translate the update into base table when the changed item has reference to other tables, thus unupdatable.

UPDATE activity_view
SET rules = 'Form teams of however many people and play!' where activity_name ='Super Mario Party Obstacle';

UPDATE activity_view
SET host_time = '2020-07-07 12:00:00' where activity_name ='Game With Anna and Elsa - Sims';


```
ERROR:  update or delete on table "activity" violates foreign key constraint "host_host_time_fkey" on table "hos
DETAIL:  Key (host_time)=(2020-02-01 16:00:00) is still referenced from table "host".
SQL state: 23503
```

Creating a view on which players can join which high_end communities. They need to satisfy 2 conditions, 1) the player should possess the game of the community 2) the level of the player should be higher than the minimum level defined in the high_end_community.

SQL query:

CREATE VIEW high_end_members AS
SELECT
player.id, player.level, gname, com_name
FROM high_end_community JOIN player ON player.level >=
high_end_community.min_level
JOIN buy ON high_end_community.game_id = buy.game_id AND player.id =
buy.player_id
JOIN game ON high_end_community.game_id = game.game_id

SELECT * FROM high_end_members

Output:

Data Output   Explain   Messages   Notifications

	id integer	level integer	gname character varying (200)	com_name character (100)	
1	10720	14	Mario Bros	Mario Discovery	...
2	15240	74	Metal Gear Solid V: The Phant...	Phantom Pain Legen...	
3	15240	74	Tetris	Tetris Community	...

Try UPDATE:

 We tried to set the levels of the players in the view to 70.

UPDATE high_end_members SET level = 70

And it ran into error:

Notifications   Messages   Data Output   Explain

ERROR: cannot update view "high_end_members"
DETAIL: Views that do not select from a single table or view
are not automatically updatable.
HINT: To enable updating the view, provide an INSTEAD OF
UPDATE trigger or an unconditional ON UPDATE DO INSTEAD rule.
SQL state: 55000

This is because we used join in the definition of view. View can only be updated if the view was created from one single table.

Question 8

As we defined when we generated ids, the user ids should be a 5-digit number between 10000 and 20000. All game ids are between 30000 and 40000. Even numbers are assigned for players and odd numbers assigned for publishers. This was not stated explicitly in the creation of tables, so we decided to add it here.

ALTER TABLE player
ADD CHECK (id <= 20000 and id >= 10000 and id%2=0);
ALTER TABLE publisher
ADD CHECK (id <= 20000 and id >= 10000 and id%2=1);

```
cs421=> \d player
      Table "cs421g60.player"
 Column  |      Type      | Modifiers
----------+------------------+-----------
 id      | integer        | not null
 level   | integer        | not null
 recharge | double precision |
Indexes:
   "player_pkey" PRIMARY KEY, btree (id)
Check constraints:
   "player_id_check" CHECK (id <= 20000 AND id >= 10000 AND (id % 2) = 0)
Foreign-key constraints:
   "player_id_fkey" FOREIGN KEY (id) REFERENCES users(id)
Referenced by:
   TABLE "buy" CONSTRAINT "player_id" FOREIGN KEY (player_id) REFERENCES
player(id)

cs421=> \d publisher
     Table "cs421g60.publisher"
 Column |      Type      | Modifiers
--------+------------------+-----------
 id     | integer        | not null
 income | double precision | not null
Indexes:
   "publisher_pkey" PRIMARY KEY, btree (id)
Check constraints:
```

"publisher_id_check" CHECK (id <= 20000 AND id >= 10000 AND (id % 2) = 1)
Foreign-key constraints:
    "publisher_id_fkey" FOREIGN KEY (id) REFERENCES users(id)

If we try to add an even id that is not in the range of (10000, 20000) in to player and publisher, it will produce error and insertion fails:

INSERT INTO player
VALUES (40000, 90, 100.01)

```
Data Output   Explain   Messages   Notifications

ERROR:  new row for relation "player" violates check constraint
"player_id_check"
DETAIL:  Failing row contains (40000, 90, 100.01).
SQL state: 23514
```

INSERT INTO publisher
VALUES (90000, 1001010)

```
Data Output   Explain   Messages   Notifications

ERROR:  new row for relation "publisher" violates check constraint
"publisher_id_check"
DETAIL:  Failing row contains (90000, 1001010).
SQL state: 23514
```

It also fails with odd ids that are not in range:

INSERT INTO player VALUES (90001,100, 900.03)

```
Data Output   Explain   Messages   Notifications

ERROR:  new row for relation "player" violates check constraint
"player_id_check"
DETAIL:  Failing row contains (90001, 100, 900.03).
SQL state: 23514
```

INSERT INTO publisher VALUES (90001,1001010)

```
Data Output   Explain   Messages   Notifications

ERROR:  new row for relation "publisher" violates check constraint
"publisher_id_check"
DETAIL:  Failing row contains (90001, 1001010).
SQL state: 23514
```

Same thing happens when the id is within the range, but is not even for player or odd for publisher:

INSERT INTO player VALUES (11111, 100, 900.03)

```
Data Output   Explain   Messages   Notifications

ERROR:  new row for relation "player" violates check constraint
"player_id_check"
DETAIL:  Failing row contains (11111, 100, 900.03).
SQL state: 23514
```

INSERT INTO publisher VALUES (11110, 100)

```
Data Output   Explain   Messages   Notifications

ERROR:  new row for relation "publisher" violates check constraint
"publisher_id_check"
DETAIL:  Failing row contains (11110, 100).
SQL state: 23514
```

Just for fun we added another check for a different table. Send_gift should only happen between two different players, so we added a check here as well.

ALTER TABLE send_gift ADD CHECK (player1 != player2)

And when we tried to add a friend to oneself:

INSERT INTO send_gift VALUES (10720, 10720, 38697)

```
Notifications   Messages   Data Output   Explain

ERROR:  new row for relation "send_gift" violates check
constraint "send_gift_check"
DETAIL:  Failing row contains (10720, 10720, 38697, 2015-02-
01).
SQL state: 23514
```

But it will succeed if it's a different person and it's not presented in the list:

INSERT INTO send_gift VALUES (10720, 11478, 38697, '2015-02-01')

```
Notifications   Messages   Data Output   Explain

INSERT 0 1

Query returned successfully in 119 msec.
```

Question 9

Please see the codes attached for specific steps and attributes.
For the creativity part, we tried the following:

- Automated data generation
  - Produce fake ids from website (https://www.random.org/integer-sets/)
  - Produced faked names, addresses, dates with python faker package (https://github.com/joke2k/faker)
- Real data sets
  - Produced random prices, recharges with python random() functions