University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: __2___

Date: February 22, 2024

Group Number: Group 3

Name	Student Number	CS Alias (Userid)	Preferred Email Address
Nicholas Zhang	53521472	k2n2p	nicholaszhang0817@gmail.com
Erik Lin	13855572	c7d7n	erik.s.lin.2011@gmail.com
Jay Park	96589361	e4m4p	jayp@student.ubc.ca

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your email address, and then let us assign you to a TA for your project supervisor.)

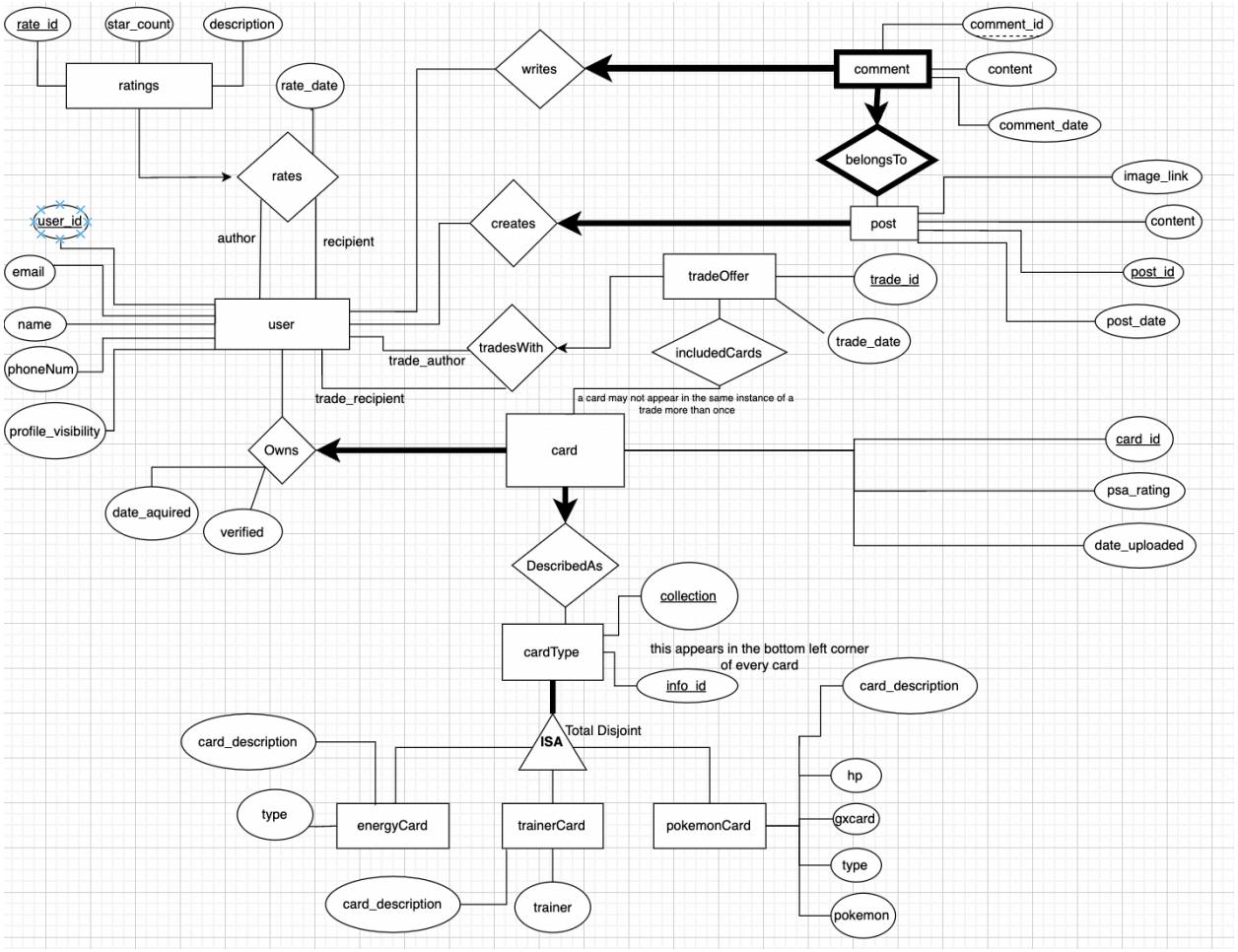
In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia.

University of British Columbia, Vancouver
Department of Computer Science

Project Summary:

Pokéswap is a database designed to streamline the trading experience for Pokémon card fans. We aim to create a database to assist users with a comprehensive tool to manage and view their card collections, create trade offers, and comment on other collectors' posts in a seamless manner. Users can easily upload and sort through their personal Pokémon card collections, as well as view other's collections. The database is designed to model several aspects of the card trading domain, including users, cards of a user, and details of a specific card, including collector number, pokémon number, type, and other specifications of the card. This application domain is perfect as a CPSC 304 project since it focuses on designing a relational database to manage complex relationships between users, their card collections, trade offers, and post-interactions. It provides opportunities to explore database indexing, querying, and management – all essential concepts covered in CPSC 304.

ER Diagram:



University of British Columbia, Vancouver
Department of Computer Science

Relational Schema:

- cardOwnsDescribedAs(card_id : INTEGER, psa_rating : INTEGER, date_uploaded : DATE, **user_id** : INTEGER, date_aquired : DATE, verified : BIT, **info_id** : VARCHAR, **collection** : VARCHAR)
    - date_uploaded NOT NULL
    - date_aquired NOT NULL
    - info_id NOT NULL
    - collection NOT NULL
    - verified NOT NULL
- cardType(info_id: VARCHAR, collection : VARCHAR)
- energyCard(**info_id** : VARCHAR, **collection** : VARCHAR, card_description : VARCHAR, type : VARCHAR)
    - type NOT NULL
- trainerCard(**info_id** : VARCHAR, **collection** : VARCHAR, card_description : VARCHAR, trainer : VARCHAR)
    - trainer NOT NULL
    - card_description NOT NULL
- pokemonCard(**info_id** : VARCHAR, **collection** : VARCHAR, card_description : VARCHAR, hp : INTEGER, gxcard : BIT, type : VARCHAR, pokemon : VARCHAR)
    - hp NOT NULL
    - gxcard NOT NULL
    - type NOT NULL
    - pokemon NOT NULL
    - card_description NOT NULL
- tradeOfferTradesWith(trade_id : INTEGER, trade_date : DATE, **trade_author_id** : INTEGER, **trade_recipient_id** : INTEGER)
    - trade_date NOT NULL
    - trade_author_id NOT NULL
    - trade_recipient_id NOT NULL
- includedCards(**trade_id** : INTEGER, **card_id** : INTEGER)
    - trade_id NOT NULL
    - card_id NOT NULL
- user(user_id : INTEGER, email : VARCHAR, name : VARCHAR, phone_num : CHAR(10), profile_visibility : BIT)
    - CANDIDATE KEY(email)
    - email UNIQUE NOT NULL
    - profile_visibility NOT NULL
- ratingsRates(rate_id : INTEGER, star_count : INTEGER, description : VARCHAR, **rate_author_id** : INTEGER**, rate_recipient_id** : INTEGER, rate_date : DATE)
    - CANDIDATE KEY(rate_author_id, rate_recipient_id)
    - star_count NOT NULL
    - rate_author_id NOT NULL

    - rate_recipient_id NOT NULL
    - (rate_author_id, rate_recipient_id) UNIQUE
    - rate_date NOT NULL
- commentsWritesBelongsTo(**post_id** : INTEGER, comment_id[partial key] : INTEGER, **user_id** : INTEGER, content : VARCHAR, comment_date : DATE)
    - content NOT NULL
    - comment_date NOT NULL
    - user_id NOT NULL
- postCreates(post_id : INTEGER, image_link : VARCHAR, content : VARCHAR, post_date : DATE, **user_id** : INTEGER)
    - content NOT NULL
    - post_date NOT NULL
    - user_id NOT NULL

University of British Columbia, Vancouver
Department of Computer Science

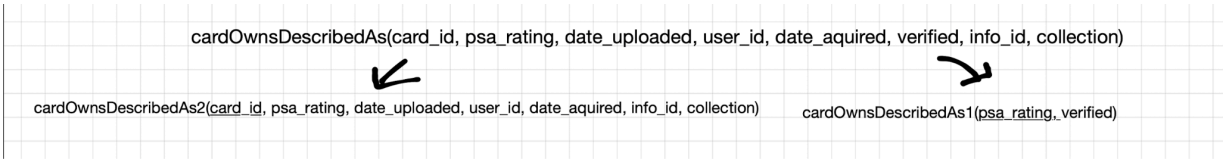
Functional Dependencies and Normalization:

cardOwnsDescribedAs:

Functional Dependencies:

- card_id → psa_rating, date_uploaded, user_id, date_aquired, verified, info_id, collection
- psa_rating → verified

Normalization:



- cardOwnsDescribedAs(card_id : INTEGER, psa_rating : INTEGER, date_uploaded : DATE, **user_id** : INTEGER, date_aquired : DATE, **info_id** : VARCHAR, **collection** : VARCHAR)
- cardOwnsDescribedAs(psa_rating : INTEGER, verified : BIT)

cardType:

Functional Dependencies:

- no non-trivial functional dependencies

Normalization: Already in BCNF Form

- includedCards(info_id **:** VARCHAR, collection: VARCHAR)

energyCard:

Functional Dependencies:

- info_id, collection → card_description, type
- type → card_description

Normalization:

- energyCard(**info_id** : VARCHAR, **collection** : VARCHAR, type : VARCHAR)
- energyCardDescriptions(card_description : VARCHAR, type : VARCHAR)

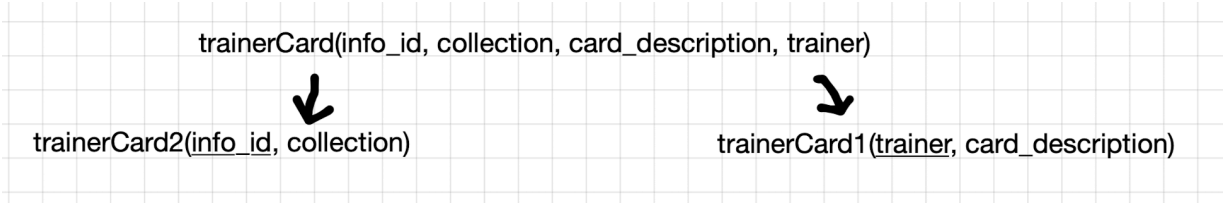
University of British Columbia, Vancouver
Department of Computer Science

trainerCard:

Functional Dependencies:

- info_id, collection → card_description, trainer
- trainer → card_description

Normalization:



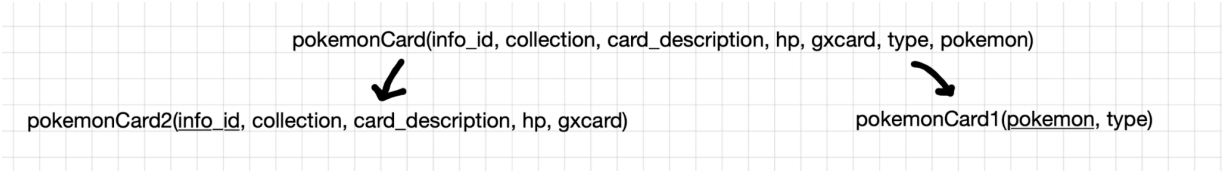
- trainerCard(**info_id** : VARCHAR, **collection** : VARCHAR, trainer : VARCHAR)
- trainerCardDescriptions(card_description : VARCHAR, trainer : VARCHAR)

pokemonCard:

Functional Dependencies:

- info_id, collection → card_description, hp, gxcard, type, pokemon
- pokemon → type

Normalization:



- pokemonCard(**info_id** : VARCHAR, **collection** : VARCHAR, card_description : VARCHAR, hp : INTEGER, gxcard : BIT, pokemon : VARCHAR)
- pokemonTypes(type : VARCHAR, pokemon : VARCHAR)

tradeOfferTradesWith:

Functional Dependencies:

- trade_id → trade_date, trade_author_id, trade_recipient_id

Normalization: Already in BCNF Form

- tradeOfferTradesWith(trade_id : INTEGER, trade_date : DATE, **trade_author_id** : INTEGER, **trade_recipient_id** : INTEGER)

University of British Columbia, Vancouver
Department of Computer Science

includedCards:

Functional Dependencies:
- no non-trivial functional dependencies

Normalization: Already in BCNF Form
- includedCards(**trade_id** : INTEGER, **card_id** : INTEGER)

user:

Functional Dependencies:
- user_id → email, name, phone_num, profile_visibility
- email → user_id, name, phone_num, profile_visibility

Normalization: Already in BCNF Form
- user(user_id : INTEGER, email : VARCHAR, name : VARCHAR, phone_num : CHAR(10), profile_visibility : BIT)
    - CANDIDATE KEY(email)

ratingsRates:

Functional Dependencies: Already in BCNF Form
- rate_id → star_count, description, rate_author_id, rate_recipient_id, rate_date
- rate_author_id, rate_recipient_id → rate_id, star_count, description, rate_date

Normalization: Already in BCNF Form
- ratingsRates(rate_id : INTEGER, star_count : INTEGER, description : VARCHAR, rate_author_id : INTEGER, rate_recipient_id : INTEGER, rate_date : DATE)
    - CANDIDATE KEY(rate_author_id, rate_recipient_id)

commentsWritesBelongsTo:

Functional Dependencies:
- post_id, comment_id → user_id, content, comment_date

Normalization: Already in BCNF Form
- commentsWritesBelongsTo(**post_id** : INTEGER, comment_id[partial key] : INTEGER, **user_id** : INTEGER, content : VARCHAR, comment_date : DATE)

University of British Columbia, Vancouver
Department of Computer Science

postCreates:

Functional Dependencies:

- post_id → image_link, content, post_date, user_id

Normalization: Already in BCNF Form

- postCreates(<u>post_id</u> : INTEGER, image_link : VARCHAR, content : VARCHAR, post_date : DATE, **user_id** : INTEGER)

University of British Columbia, Vancouver
Department of Computer Science

SQL DDL statements and Insertions:

```
CREATE TABLE cardOwnsDescribedAs(
     card_id int PRIMARY KEY,
     psa_rating int,
     date_uploaded date NOT NULL,
     user_id int,
     date_aquired date NOT NULL,
     info_id varchar NOT NULL,
     collection varchar NOT NULL,
     FOREIGN KEY (user_id) REFERENCES user ON DELETE SET NULL,
     FOREIGN KEY (info_id, collection) REFERENCES cardType
);

5 tuples:
INSERT INTO cardOwnsDescribedAs (card_id, psa_rating,
date_uploaded, user_id, date_aquired, info_id, collection)
VALUES
(1, 9, '2024-03-01', 1, '2024-02-01', 1, 'BS'),
(2, 8, '2024-03-02', 2, '2024-02-02', 4, 'BS'),
(3, NULL, '2024-03-03', 3, '2024-02-03', 2, 'BS'),
(4, 10, '2024-03-04', 4, '2024-02-04', 3, 'BS'),
(5, NULL, '2024-03-05', NULL, '2024-02-05', 140, 'KSS');
```

University of British Columbia, Vancouver
Department of Computer Science

```
CREATE TABLE cardType(
     info_id varchar,
     collection varchar,
     PRIMARY KEY (info_id, collection)
);

15 tuples:
INSERT INTO cardType (info_id, collection) VALUES
(3, 'BS'),
(2, 'BS'),
(1, 'BS'),
(4, 'BS'),
(140, 'KSS'),
('178','CES'),
('181','ASR'),
('124','CRZ'),
('102','G2'),
('91','FLF'),
('42','XY'),
('239','CEC'),
('154','UNB'),
('22','AOR'),
('92','GRI');
```

```
CREATE TABLE energyCard(
     info_id varchar,
     collection varchar,
     type varchar NOT NULL,
     PRIMARY KEY (info_id, collection),
     FOREIGN KEY (info_id, collection) REFERENCES cardType
     FOREIGN KEY (type) REFERENCES energyCardDescriptions
);

5 tuples:
INSERT INTO energyCard (info_id, collection, type) VALUES
(3, 'BS', 'water'),
(2, 'BS', 'fire'),
(1, 'BS', 'grass'),
(4, 'BS', 'electric'),
(140, 'KSS', 'fairy');
```

```
CREATE TABLE energyCardDescriptions(
     type varchar PRIMARY KEY,
     card_description varchar
);

5 Tuples:
INSERT INTO energyCardDescriptions (type, card_description)
VALUES
('lightning', NULL),
('fairy', NULL),
('scramble_energy', 'Scramble Energy can be attached only to an
Evolved Pokémon (excluding Pokémon-ex). Scramble Energy
provides   Colorless Energy.  While in play, if you have more
Prize cards left than your opponent, Scramble Energy provides
every  type of Energy but provides only 3 in any combination at
a time. If the Pokémon Scramble Energy is  attached to isn't an
Evolved Pokémon (or evolves into Pokémon-ex), discard Scramble
Energy.'),
('water', NULL),
('r_energy', 'R Energy can be attached only to a Pokémon that
has Dark or Rocket's in its name. While in play, R Energy
provides 2 Darkness Energy. (Doesn't count as a basic Energy
card.) If the Pokémon R Energy is attached to attacks, the
attack does 10 more damage to the Active Pokémon (before
applying Weakness and Resistance). When your turn ends, discard
R Energy.');
```

```
CREATE TABLE trainerCard(
     info_id varchar,
     collection varchar,
     trainer varchar NOT NULL,
     PRIMARY KEY (info_id, collection),
     FOREIGN KEY (info_id, collection) REFERENCES cardType
     FOREIGN KEY (trainer) REFERENCES trainerCardDescriptions
);

5 Tuples:
INSERT INTO trainerCard (info_id, collection, trainer) VALUES
('178','CES', 'acro_bike'),
('181','ASR', 'adaman'),
('124','CRZ', 'bede'),
('102','G2', 'chaos_gym'),
('91','FLF', 'magenetic_storm');
);
```

```
CREATE TABLE trainerCardDescriptions(
     trainer varchar PRIMARY KEY,
     card_description varchar NOT NULL
);

5 Tuples:
INSERT INTO trainerCardDescriptions (info_id, collection,
trainer) VALUES
('acro_bike','Look at the top 2 cards of your deck and put 1 of
them into your hand. Discard the other card.'),
('adaman','You can use this card only if you discard 2 Metal
Energy cards from your hand. Search your deck for up to 2 cards
and put them into your hand. Then, shuffle your deck.'),
('bede','Attach a basic Energy card from your hand to 1 of your
Benched Pokémon.'),
('chaos_gym','This card stays in play after being played.
Discard this card if another Stadium card comes into play.
Whenever a player plays a Trainer card other than a Stadium
card, he or she flips a coin. If heads, that player plays that
card normally. If tails, the player can't play that card. If
the card isn't put into play, the player's opponent may use
that card instead, if he or she does everything required in
order to play that card (like discarding cards). Either way,
the card goes to its owner's discard pile.
'),
('magenetic_storm','Each Pokémon in play has no Resistance.');
);
```

University of British Columbia, Vancouver
Department of Computer Science

```
CREATE TABLE pokemonCard(
     info_id varchar,
     collection varchar,
     pokemon varchar NOT NULL,
     card_description varchar NOT NULL,
     hp int NOT NULL,
     gxcard bit NOT NULL,
     PRIMARY KEY (info_id, collection),
     FOREIGN KEY (info_id, collection) REFERENCES cardType
     FOREIGN KEY (pokemon) REFERENCES pokemonTypes
);

5 Tuples:
INSERT INTO pokemonCard (info_id, collection, pokemon,
card_description, hp, gxcard) VALUES
('42','XY','pikachu','Nuzzle',60,0),
('239','CEC','piplup','Bubble Hold',60,0),
('154','UNB','porygon','Digicharge',50,0),
('22','AOR','vaporeon','Aqua Effect',90,0),
('92','GRI','sylveon','Magical Ribbon',200,1);
```

```
CREATE TABLE pokemonTypes(
     pokemon varchar PRIMARY KEY,
     type varchar NOT NULL
);

5 Tuples:
INSERT INTO pokemonTypes (pokemon, type) VALUES
('pikachu','lightning'),
('sylveon','fairy'),
('vaporeon','water'),
('piplup','water'),
('porygon','normal');
```

```
CREATE TABLE tradeOfferTradesWith(
     trade_id int PRIMARY KEY,
     trade_date date NOT NULL,
     trade_author_id int NOT NULL,
     trade_recipient_id int NOT NULL,
     FOREIGN KEY (trade_author_id) REFERENCES user(user_id) ON
DELETE CASCADE,
     FOREIGN KEY (trade_recipient_id) REFERENCES user(user_id)
ON DELETE CASCADE
);

5 Tuples:
INSERT INTO tradeOfferTradesWith (trade_id, trade_date,
trade_author_id, trade_recipient_id) VALUES
(1, '2024-02-28', 1, 2),
(2, '2024-02-27', 3, 4),
(3, '2024-02-26', 5, 2),
(4, '2024-02-25', 1, 3),
(5, '2024-02-24', 5, 4);
```

```
CREATE TABLE includedCards(
     trade_id int,
     card_id int,
     PRIMARY KEY (trade_id, card_id),
     FOREIGN KEY (trade_id) REFERENCES tradeOfferTradesWith ON
DELETE CASCADE,
     FOREIGN KEY (card_id) REFERENCES cardOwnsDescribedAs ON
DELETE CASCADE,
);

INSERT INTO includedCards (trade_id, card_id) VALUES
(1, 1),
(1, 2),
(2, 3),
(3, 4),
(4, 5);
```

```
CREATE TABLE user (
     user_id int PRIMARY KEY,
     email varchar UNIQUE NOT NULL,
     name varchar,
     phone_num char(10),
     profile_visibility bit NOT NULL DEFAULT 1
);

5 Tuples:
INSERT INTO user (user_id, email, name, phone_num,
profile_visibility) VALUES
(1, 'erik.lin@gmail.com', 'Erik Lin', '1234567890', 1),
(2, 'nich.zhang@gmail.com', 'Nechael Zhang', '2345678901', 1),
(3, 'jay.park@gmail.com', 'Jay Park', '3456789012', 0),
(4, 'jamie.kim@gmail.com', 'Jamie Kim', NULL, 1),
(5, 'anon@gmail.com', NULL, NULL, NULL);
```

```
CREATE TABLE ratingsRates (
     rate_id int PRIMARY KEY,
     star_count int NOT NULL,
     description varchar,
     rate_author_id int NOT NULL,
     rate_recipient_id int NOT NULL,
     rate_date date NOT NULL,
     FOREIGN KEY (rate_author_id) REFERENCES user(user_id) ON
DELETE CASCADE,
     FOREIGN KEY (rate_recipient_id) REFERENCES user(user_id)
ON DELETE CASCADE,
     UNIQUE (rate_author_id, rate_recipient_id)
);

5 Tuples:
INSERT INTO ratingsRates (rate_id, star_count, description,
rate_author_id, rate_recipient_id, rate_date) VALUES
(1, 5, 'Perfect!', 1, 2, '2024-02-28'),
(2, 4, 'Smooth experience and fast communication', 2, 3,
'2024-02-27'),
(3, 1, 'Scammer!', 3, 4, '2024-02-26'),
(4, 2, NULL, 4, 5, '2024-02-25'),
(5, 5, 'Friendly and fast.', 5, 1, '2024-02-24');
```

```
CREATE TABLE commentsWritesBelongsTo (
     post_id int,
     comment_id int,
     user_id int NOT NULL,
     content varchar NOT NULL,
     comment_date date NOT NULL,
     PRIMARY KEY (post_id, comment_id),
     FOREIGN KEY (post_id) REFERENCES postCreates ON DELETE
CASCADE,
     FOREIGN KEY (user_id) REFERENCES user ON DELETE CASCADE
);

5 Tuples:
INSERT INTO commentsWritesBelongsTo (post_id, comment_id,
user_id, content, comment_date) VALUES
(1, 1, 1, 'Such a cool card!', '2024-02-28'),
(2, 1, 1, 'I think this is a fake!', '2024-02-27'),
(3, 3, 2, 'Did you get scammed?', '2024-02-26'),
(4, 4, 3, 'I completely agree.', '2024-02-25'),
(4, 5, 4, 'I disagree with your opinion...', '2024-02-24');
```

```
CREATE TABLE postCreates (
     post_id int PRIMARY KEY,
     image_link varchar,
     content varchar NOT NULL,
     post_date date NOT NULL,
     user_id int NOT NULL,
     FOREIGN KEY (user_id) REFERENCES user ON DELETE CASCADE
);

5 Tuples:
INSERT INTO postCreates (post_id, image_link, content,
post_date, user_id) VALUES
(1, 'http://pokeswap.com/images/pikachu.jpg', 'Look at this
cute pikachu!', '2024-02-28', 1),
(2, NULL, 'Are the new holographic cards worth it?',
'2024-02-27', 1),
(3, 'http://pokeswap.com/images/charizard.jpg', 'Charizard's
card design over the years', '2024-02-26', 2),
(4, 'http://pokeswap.com/images/greninja.jpg', 'Check this
Greninja out.', '2024-02-25', 3),
(5, 'http://pokeswap.com/images/mewtwo.jpg', 'Mewtwo EX's
design is possibly flawed!?', '2024-02-24', 4);
```