

A6: Indexes, triggers, user functions, transactions and population

Our project, answerly, is a web application for collaborative questions and answers.

This artifact contains the Indexes, triggers, user functions, transactions and population specifications of the database.

1. Database Workload

1.1. Tuple Estimation

Relation reference	Relation Name	Order of magnitude	Estimated growth
R01	user	thousands	units per day
R02	label	dozens	units per day
R03	notification	thousands	thousands per day
R04	notification	thousands	thousands per day
R05	user_management	thousands	units per day
R06	question	thousands	thousands per day
R07	answers	thousands	thousands per day
R08	comment	thousands	hundreds per day
R09	vote	thousands	thousands per day
R010	report	hundreds	dozens per day
R012	following	thousands	units per day
R013	about	thousands	thousands per day

1.2. Frequent Queries

Most important queries (SELECT) and their frequency.

Query	SELECT01
Description	Gets questions with most likes
Frequency	thousands per day

SQL code

```
SELECT * FROM question
ORDER BY question.nr_likes desc
```

```
LIMIT 30;
```

Query	SELECT02
-------	----------

Description	Gets most recent questions
-------------	----------------------------

Frequency	thousands per day
-----------	-------------------

SQL code

```
SELECT * FROM question
ORDER BY question.question_date desc
LIMIT 30;
```

Query	SELECT03
-------	----------

Description	Gets most scored answers
-------------	--------------------------

Frequency	hundreds per day
-----------	------------------

SQL code

```
SELECT * FROM question
ORDER BY answer.nr_likes desc
LIMIT 30;
```

Query	SELECT04
-------	----------

Description	Gets most recent answers
-------------	--------------------------

Frequency	hundreds per day
-----------	------------------

SQL code

```
SELECT * FROM question
ORDER BY answer.answer_date desc
LIMIT 30;
```

Query	SELECT05
-------	----------

Description	Gets most recent comments
-------------	---------------------------

Frequency	hundreds per day
-----------	------------------

SQL code

```
SELECT * FROM question
ORDER BY comment.comment_date desc
LIMIT 30;
```

Query	SELECT06
Description	Gets most popular labels
Frequency	hundreds per day

SQL code

```
SELECT *
FROM label join about ON label.id = about.label_id
JOIN question ON question.id = about.question_id
GROUP BY label.id
LIMIT 5;
```

Query	SELECT07
Description	Gets most popular questions within label
Frequency	hundreds per day

SQL code

```
SELECT label.name
FROM label join about ON label.id = about.label_id
JOIN question ON question.id = about.question_id
WHERE label.name = $label_name
ORDER BY question.nr_likes desc
LIMIT 20;
```

Query	SELECT08
Description	Gets most recent questions within label
Frequency	hundreds per day

SQL code

```
SELECT label.name
FROM label join about ON label.id = about.label_id
JOIN question ON question.id = about.question_id
WHERE label.name = $label_name
ORDER BY question.question_date desc
LIMIT 20;
```

Query	SELECT09
Description	Gets all user's question, score and answers
Frequency	hundreds per day

SQL code

```
SELECT *  
FROM "user" u join question ON u.id = question.user_id  
JOIN answer ON u.id = answer.user_id  
WHERE u.id = $id;
```

Query	SELECT10
Description	Gets all notifications order by date of a given user
Frequency	hundreds per day

SQL code

```
SELECT *  
FROM "user" u join notification ON u.id = notification.user_id  
WHERE u.id = $id  
ORDER BY notification.date desc;
```

Query	SELECT11
Description	Gets user info
Frequency	hundreds per day

SQL code

```
SELECT name, bio, first_name, last_name, score  
FROM "user" u  
WHERE u.id = $id
```

Query	SELECT12
Description	Gets a label with string
Frequency	dozens per day

SQL code

```
SELECT name
FROM label
WHERE name LIKE '%$string%'
ORDER BY notification.date desc;
```

Query	SELECT13
--------------	-----------------

Description	Gets a question with string
--------------------	-----------------------------

Frequency	dozens per day
------------------	----------------

SQL code

```
SELECT name
FROM question
WHERE description LIKE '%$string%'
ORDER BY notification.date desc;
```

Query	SELECT14
--------------	-----------------

Description	Gets questions reports
--------------------	------------------------

Frequency	dozens per day
------------------	----------------

SQL code

```
SELECT state, comment, responsible_user
FROM report
WHERE answer_id IS NULL
AND comment_id IS NULL;
```

Query	SELECT15
--------------	-----------------

Description	Gets answers reports
--------------------	----------------------

Frequency	dozens per day
------------------	----------------

SQL code

```
SELECT state, comment, responsible_user
FROM report
WHERE question_id IS NULL
AND comment_id IS NULL;
```

Query	SELECT16
--------------	-----------------

Query	SELECT16
Description	Gets comments reports
Frequency	dozens per day

SQL code

```
SELECT state, comment, responsible_user
FROM report
WHERE question_id IS NULL
AND answer_id IS NULL;
```

Query	SELECT17
Description	Gets user reports
Frequency	dozens per day

SQL code

```
SELECT *
FROM report
WHERE user_id IS NOT NULL
GROUP BY user_id;
```

Query	SELECT18
Description	Gets users with most scores
Frequency	dozens per day

SQL code

```
SELECT *
FROM "user" u join user_management on u.id = user_management.user_id
where user_management.status == 'normal'
ORDER BY score desc
LIMIT 30;
```

Query	SELECT19
Description	Gets moderators with most scores
Frequency	dozens per day

SQL code

```
SELECT *
FROM "user" u join user_management on u.id = user_management.user_id
where user_management.status == 'moderator'
ORDER BY score desc
LIMIT 30;
```

Query	SELECT20
Description	Get question info
Frequency	dozens per day

SQL code

```
SELECT *
FROM question WHERE id = $id;
```

Query	SELECT21
Description	Get comment info
Frequency	dozens per day

SQL code

```
SELECT *
FROM comment WHERE id = $id;
```

Query	SELECT22
Description	Get a user's score
Frequency	dozens per day

SQL code

```
SELECT score
FROM "user" u WHERE u.id = $id;
```

1.3. Frequent Updates

This section features the most important updates and their respective convenience.

Query	INSERT01
-------	----------

Query	INSERT01
Description	Adding a new question
Frequency	hundreds per day

SQL code

```
INSERT INTO questions (user_id, title, description)
VALUES ($user_id, $title, $description);
```

Query	INSERT02
Description	Adding a new answer
Frequency	hundreds per day

SQL code

```
INSERT INTO answer (user_id, question_id, content)
VALUES ($user_id, $question_id, $content);
```

Query	INSERT03
Description	Adding a new coment
Frequency	dozens per day

SQL code

```
INSERT INTO comment (user_id, question_id, answer_id, content)
VALUES ($user_id, $question_id, $answer_id $content);
```

Query	INSERT04
Description	Adding a new user
Frequency	dozens per day

SQL code

```
INSERT INTO "user" (first_name, last_name, email, bio, username, password, bio)
VALUES ($first_name, $last_name, $email, $bio, $username, $password, $bio);
```

Query	UPDATE01
-------	----------

Query	UPDATE01
Description	Updating user info
Frequency	dozens per day

SQL code

```
UPDATE "user"  
SET username = $username, bio = $bio, first_name = $first_name, last_name =  
$last_name  
VALUES id = $id;
```

Query	UPDATE02
Description	Change user status
Frequency	dozens per day

SQL code

```
UPDATE user_management  
SET status = $status  
VALUES user_id = $user_idid;
```

Query	UPDATE03
Description	Edit question
Frequency	dozens per day

SQL code

```
UPDATE question  
SET title = $title, description = $description  
VALUES id = $id;
```

Query	UPDATE04
Description	Edit question
Frequency	dozens per day

SQL code

```
UPDATE answer
SET content = $content
VALUES id = $id;
```

Query	UPDATE05
Description	Edit question
Frequency	dozens per day

SQL code

```
UPDATE comment
SET content = $content
VALUES id = $id;
```

2. Proposed Indices

2.1. Performance Indices

Indices proposed to improve performance of the identified queries. B-Tree is used when we want to grab chunks of data, whilst Hash is used to grab specific database elements.

Index	IDX01
Related queries	SELECT01, SELECT07, SELECT23
Relation	question
Attribute	nr_likes
Type	B-tree
Cardinality	high
Clustering	No
Justification	To allow searching questions ordered by score faster.

```
CREATE INDEX question_score ON question USING btree(nr_likes)
```

Index	IDX02
Related queries	SELECT02, SELECT08
Relation	question
Attribute	question_date

Index	IDX02
Type	B-tree
Cardinality	high
Clustering	No
Justification	To allow searching questions ordered by date faster.

```
CREATE INDEX question_date ON question USING btree(question_date)
```

Index	IDX03
Related queries	SELECT03
Relation	answer
Attribute	question_id, nr_likes
Type	B-tree
Cardinality	medium
Clustering	No
Justification	To allow searching answers to a specific question ordered by score faster.

```
CREATE INDEX answer_score ON answer USING btree(question_id, nr_likes)
```

Index	IDX04
Related queries	SELECT04
Relation	answer
Attribute	question_id, answer_date
Type	B-tree
Cardinality	medium
Clustering	No
Justification	To allow searching answers to a specific question ordered by date faster.

```
CREATE INDEX answer_date ON answer USING btree(question_id, answer_date)
```

Index	IDX05
-------	-------

Index	IDX05
Related queries	SELECT05
Relation	comment
Attribute	question_id, comment_date
Type	B-tree
Cardinality	medium
Clustering	No
Justification	To allow searching comments to a specific question ordered by date faster.

```
CREATE INDEX comment_date ON comment USING btree(question_id, comment_date)
```

Index	IDX06
Related queries	SELECT06
Relation	following
Attribute	label_id
Type	B-tree
Cardinality	medium
Clustering	No
Justification	To allow searching for the most popular labels faster.

```
CREATE INDEX label_popularity ON following USING btree(label_id)
```

Index	IDX07
Related queries	SELECT09
Relation	question
Attribute	user_id
Type	B-tree
Cardinality	medium
Clustering	No
Justification	To allow searching for a user's number of questions faster.

```
CREATE INDEX question_user ON question USING btree(user_id)
```

Index	IDX08
Related queries	SELECT09
Relation	answer
Attribute	user_id
Type	B-tree
Cardinality	medium
Clustering	No
Justification	To allow searching for a user's number of answers faster.

```
CREATE INDEX answer_user ON answer USING btree(user_id)
```

Index	IDX09
Related queries	SELECT10
Relation	notification
Attribute	user_id, date
Type	B-tree
Cardinality	medium
Clustering	No
Justification	To allow searching for a user's number of notifications ordered by date faster.

```
CREATE INDEX notification_user_date ON notification USING btree(user_id, date)
```

Index	IDX10
Related queries	SELECT11
Relation	user
Attribute	username
Type	Hash
Cardinality	medium
Clustering	No

Index	IDX10
Justification	To allow retrieving user information faster, based on the email

```
CREATE INDEX user_username ON "user" USING hash(username)
```

Index	IDX11
Related queries	SELECT17
Relation	report
Attribute	user_id
Type	B-tree
Cardinality	low
Clustering	No
Justification	To allow searching for the number of reports a user has received

```
CREATE INDEX report_user ON report USING btree(user_id)
```

Index	IDX12
Related queries	SELECT18, SELECT19
Relation	user
Attribute	score
Type	B-tree
Cardinality	low
Clustering	No
Justification	To allow searching for users and moderators ordered by score faster

```
CREATE INDEX user_score ON "user" USING btree(score)
```

2.2. Full-text Search Indices

GIN is used for infrequently updated tables, leading to faster lookups. GiST is better for dynamic data.

Index	IDX13
Related queries	SELECT12

Index	IDX13
Relation	label
Attribute	name
Type	GIN
Cardinality	high
Clustering	No
Justification	To enhance performance of full text searches on label names

```
CREATE INDEX label_name ON label USING gin(to_tsvector('english', name));
```

Index	IDX14
Related queries	SELECT13
Relation	question
Attribute	title
Type	GiST
Cardinality	high
Clustering	No
Justification	To enhance performance of full text searches on question titles

```
CREATE INDEX question_title ON question USING gist(to_tsvector('english', title))
```

3. Triggers

Trigger	TRIGGER01
Description	When a user votes on a question, the number of likes or dislikes of that question needs to be updated, as well as the owner's score.
SQL code	

```
CREATE FUNCTION update_score_question() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT vote.id FROM vote WHERE NEW."vote" = FALSE) THEN
        UPDATE question
        SET nr_dislikes = nr_dislikes+1
        WHERE NEW.question_id = id;
```

```

        UPDATE "user"
        SET score = score-1
        FROM question
        WHERE NEW.question_id = question.id AND question.user_id = "user".id;
    ELSE IF EXISTS (SELECT vote.id FROM vote WHERE NEW."vote" = TRUE) THEN
        UPDATE question
        SET nr_likes = nr_likes+1
        WHERE NEW.question_id = id;
        UPDATE "user"
        SET score = score+1
        FROM question
        WHERE NEW.question_id = question.id AND question.user_id = "user".id;
    END IF;
END IF;
RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER update_score_question
AFTER INSERT OR UPDATE ON vote
FOR EACH ROW
EXECUTE PROCEDURE update_score_question();

```

Trigger**TRIGGER02****Description**

When a user votes on an answer, the number of likes or dislikes of that answer needs to be updated, as well as the owner's score.

SQL code

```

CREATE FUNCTION update_score_answer() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT vote.id FROM vote WHERE NEW."vote" = FALSE) THEN
        UPDATE answer
        SET nr_dislikes = nr_dislikes+1
        WHERE NEW.answer_id = id;
        UPDATE "user"
        SET score = score-1
        FROM answer
        WHERE NEW.answer_id = answer.id AND answer.user_id = "user".id;
    ELSE IF EXISTS (SELECT vote.id FROM vote WHERE NEW."vote" = TRUE) THEN
        UPDATE answer
        SET nr_likes = nr_likes+1
        WHERE NEW.answer_id = id;
        UPDATE "user"
        SET score = score+1
        FROM answer
        WHERE NEW.answer_id = answer.id AND answer.user_id = "user".id;
    END IF;

```



```

        END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER update_score_answer
    AFTER INSERT OR UPDATE ON vote
    FOR EACH ROW
    EXECUTE PROCEDURE update_score_answer();

```

Trigger	TRIGGER03
---------	-----------

Description	Users cannot vote on their own questions.
--------------------	---

SQL code

```

CREATE FUNCTION vote_own_question() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT question.user_id
                FROM question
                WHERE NEW.user_id = question.user_id AND NEW.question_id =
question.id) THEN
        RAISE EXCEPTION 'A user cannot vote on his own question';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER vote_own_question
    BEFORE INSERT OR UPDATE ON vote
    FOR EACH ROW
    EXECUTE PROCEDURE vote_own_question();

```

Trigger	TRIGGER04
---------	-----------

Description	Users cannot vote on their own answers.
--------------------	---

SQL code

```

CREATE FUNCTION vote_own_answer() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT answer.user_id
                FROM answer
                WHERE NEW.user_id = answer.user_id AND NEW.answer_id = answer.id)
    THEN

```

```

        RAISE EXCEPTION 'A user cannot vote on his own answer';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER vote_own_answer
    BEFORE INSERT OR UPDATE ON vote
    FOR EACH ROW
    EXECUTE PROCEDURE vote_own_answer();

```

Trigger	TRIGGER05
---------	-----------

Description	The date of an answer must be greater than or equal to the date of the respective question.
--------------------	---

SQL code

```

CREATE FUNCTION answer_date() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT question.question_date
               FROM question
               WHERE NEW.question_id = question.id AND NEW.answer_date <
question.question_date) THEN
        RAISE EXCEPTION 'The date of an answer cannot be earlier than the date of
its question';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER answer_date
    BEFORE INSERT OR UPDATE ON answer
    FOR EACH ROW
    EXECUTE PROCEDURE answer_date();

```

Trigger	TRIGGER06
---------	-----------

Description	The date of a comment must be greater than or equal to the date of the respective answer.
--------------------	---

SQL code

```

CREATE FUNCTION comment_date_answer() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT answer.answer_date
               FROM answer

```

```

        WHERE NEW.answer_id = answer.id AND NEW.comment_date <
answer.answer_date) THEN
    RAISE EXCEPTION 'The date of a comment cannot be earlier than the date of
its answer';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER comment_date_answer
    BEFORE INSERT OR UPDATE ON comment
    FOR EACH ROW
    EXECUTE PROCEDURE comment_date_answer();

```

Trigger	TRIGGER07
---------	-----------

Description	The date of a comment must be greater than or equal to the date of the respective question.
--------------------	---

SQL code

```

CREATE FUNCTION comment_date_question() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT question.question_date
                FROM question
                WHERE NEW.question_id = question.id AND NEW.comment_date <
question.question_date) THEN
        RAISE EXCEPTION 'The date of a comment cannot be earlier than the date of
its question';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER comment_date_question
    BEFORE INSERT OR UPDATE ON comment
    FOR EACH ROW
    EXECUTE PROCEDURE comment_date_question();

```

Trigger	TRIGGER08
---------	-----------

Description	An answer or question can only be voted once by the same user.
--------------------	--

SQL code

```

CREATE FUNCTION vote_once() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT * FROM vote
                WHERE ((NEW.user_id = vote.user_id AND NEW.question_id =
vote.question_id) OR
                        (NEW.user_id = vote.user_id AND NEW.answer_id =
vote.answer_id))) THEN
        RAISE EXCEPTION 'An element can only be voted once by the same user';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER vote_once
    BEFORE INSERT ON vote
    FOR EACH ROW
    EXECUTE PROCEDURE vote_once();

```

Trigger	TRIGGER09
---------	-----------

Description	A report can only be handled by an administrator or moderator.
--------------------	--

SQL code

```

CREATE FUNCTION report_status_responsible() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF NOT EXISTS (SELECT user_management.user_id
                    FROM user_management
                    WHERE ((user_management.status = 'moderator' OR
user_management.status = 'administrator')
                        AND
                        (user_management.user_id = NEW.responsible_user))) THEN
        RAISE EXCEPTION 'A report can only be handled by an administrator or
moderator';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER report_status_responsible
    BEFORE INSERT OR UPDATE ON report_status
    FOR EACH ROW
    EXECUTE PROCEDURE report_status_responsible();

```

Trigger	TRIGGER10
---------	-----------

Trigger	TRIGGER10
Description	There should be only one marked correct answer per question.
SQL code	

```

CREATE FUNCTION marked_answer() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT marked_answer FROM answer WHERE answer.id = NEW.id AND
NEW.marked_answer = TRUE) THEN
        UPDATE answer
        SET marked_answer = FALSE
        WHERE (answer.question_id = NEW.question_id AND answer.id != NEW.id);
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER marked_answer
AFTER UPDATE OF marked_answer ON answer
FOR EACH ROW
EXECUTE PROCEDURE marked_answer();

```

4. Transactions

T01	Insert a new user
Justification	When we add a new user, we need to guarantee that both inserts on table user and table user_management are consistently successful, so that we use a transaction. The Isolation Level is Repeatable Read because we want to ensure that the user_management doesn't get a wrong user foreign key, if a new user is inserted at the same time. This Isolation Level sees a snapshot as of the start of the transaction, not seeing changes made by other transactions that committed after his own transaction started.
Isolation level	REPEATABLE READ

SQL code

```

BEGIN TRANSACTION;
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ
INSERT INTO "user" (first_name, last_name, email, bio, username, password,
score)
VALUES ($first_name, $last_name, $email, $bio, $username,
$hashed_password, $score);

INSERT INTO user_management (user_id)

```

```
VALUES (currval('user_id_seq'));

COMMIT;
```

T02 Insert a new report

Justification

When we add a new report, we need to guarantee that both inserts on table report and table report_status are consistently successful, so that we use a transaction. The Isolation Level is Repeatable Read because we want to ensure that the report_status doesn't get a wrong report foreign key, if a new report is inserted at the same time. This Isolation Level sees a snapshot as of the start of the transaction, not seeing changes made by other transactions that committed after his own transaction started.

Isolation level

REPEATABLE READ

SQL code

```
BEGIN TRANSACTION;
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ
INSERT INTO report (user_id, question_id, answer_id, comment_id)
VALUES ($user_id, $question_id, $answer_id, $comment_id);

INSERT INTO report_status (report_id, comment, responsible_user)
VALUES (currval('report_id_seq'), $comment, $responsible_user);

COMMIT;
```

5. SQL Code

5.1. Database schema

```
-----
-- Drop old schmemma
-----

DROP TABLE IF EXISTS "user" CASCADE;
DROP TABLE IF EXISTS label CASCADE;
DROP TABLE IF EXISTS notification CASCADE;
DROP TABLE IF EXISTS user_management CASCADE;
DROP TABLE IF EXISTS question CASCADE;
DROP TABLE IF EXISTS answer CASCADE;
DROP TABLE IF EXISTS comment CASCADE;
DROP TABLE IF EXISTS vote CASCADE;
DROP TABLE IF EXISTS report CASCADE;
DROP TABLE IF EXISTS report_status CASCADE;
DROP TABLE IF EXISTS following CASCADE;
DROP TABLE IF EXISTS about CASCADE;
```

```

-----
-- Tables
-----

-- Table: user
CREATE TABLE "user" (
    id          SERIAL          PRIMARY KEY,
    first_name  TEXT            NOT NULL,
    last_name   TEXT            NOT NULL,
    email       TEXT            NOT NULL UNIQUE,
    bio         TEXT,
    username    TEXT            NOT NULL UNIQUE,
    password    TEXT            NOT NULL,
    score       INTEGER         NOT NULL DEFAULT 0
);

-- Table: label
CREATE TABLE label (
    id          SERIAL          PRIMARY KEY,
    name        TEXT            NOT NULL
);

-- Table: user_management
CREATE TABLE user_management (
    id          SERIAL          PRIMARY KEY,
    status      TEXT            DEFAULT 'user' NOT NULL,
    user_id     INTEGER         REFERENCES "user" (id) NOT NULL UNIQUE
);

-- Table: question
CREATE TABLE question (
    id          SERIAL          PRIMARY KEY,
    user_id     INTEGER         REFERENCES "user" (id) NOT NULL,
    title       TEXT            NOT NULL,
    description  TEXT            NOT NULL,
    nr_likes    INTEGER         DEFAULT 0 NOT NULL,
    nr_dislikes INTEGER         DEFAULT 0 NOT NULL,
    question_date DATE          DEFAULT 'Now' NOT NULL,
    CHECK (
        nr_likes >= 0 AND nr_dislikes >= 0
    )
);

-- Table: answer
CREATE TABLE answer (
    id          SERIAL          PRIMARY KEY,
    user_id     INTEGER         REFERENCES "user" (id) NOT NULL,
    question_id INTEGER         REFERENCES "question" (id) NOT NULL,
    answer_date DATE            DEFAULT 'Now' NOT NULL,
    content     TEXT            NOT NULL,
    nr_likes    INTEGER         DEFAULT 0 NOT NULL,
    nr_dislikes INTEGER         DEFAULT 0 NOT NULL,
    marked_answer BOOLEAN       DEFAULT FALSE NOT NULL,

```

```
        CHECK (
            nr_likes >= 0 AND nr_dislikes >= 0
        )
    );

-- Table: comment
CREATE TABLE comment (
    id                SERIAL                PRIMARY KEY,
    user_id           INTEGER               REFERENCES "user" (id) NOT NULL,
    question_id       INTEGER               REFERENCES "question" (id),
    answer_id         INTEGER               REFERENCES "answer" (id),
    comment_date      DATE                  DEFAULT 'Now' NOT NULL,
    content            TEXT                  NOT NULL,
    CHECK (
        (question_id IS NOT NULL AND answer_id IS NULL) OR
        (question_id IS NULL AND answer_id IS NOT NULL)
    )
);

-- Table: notification
CREATE TABLE notification (
    id                SERIAL                PRIMARY KEY,
    content            TEXT                  NOT NULL,
    date              DATE                  DEFAULT 'Now' NOT NULL,
    viewed            BOOLEAN               DEFAULT FALSE NOT NULL,
    user_id           INTEGER               REFERENCES "user" (id) NOT NULL
);

-- Table: vote
CREATE TABLE vote (
    id                SERIAL                PRIMARY KEY,
    "vote"            BOOLEAN               NOT NULL,
    user_id           INTEGER               REFERENCES "user" (id) NOT NULL,
    question_id       INTEGER               REFERENCES "question" (id),
    answer_id         INTEGER               REFERENCES "answer" (id),
    CHECK (
        (question_id IS NOT NULL AND answer_id IS NULL) OR
        (question_id IS NULL AND answer_id IS NOT NULL)
    )
);

-- Table: report
CREATE TABLE report (
    id                SERIAL                PRIMARY KEY,
    user_id           INTEGER               REFERENCES "user" (id),
    question_id       INTEGER               REFERENCES "question" (id),
    answer_id         INTEGER               REFERENCES "answer" (id),
    comment_id        INTEGER               REFERENCES "comment" (id),
    CHECK(
        (question_id IS NOT NULL AND answer_id IS NULL AND comment_id IS NULL) OR
        (question_id IS NULL AND answer_id IS NOT NULL AND comment_id IS NULL) OR
        (question_id IS NULL AND answer_id IS NULL AND comment_id IS NOT NULL)
    )
);
```



```

-- Table: report_status
CREATE TABLE report_status (
    id SERIAL PRIMARY KEY,
    report_id INTEGER REFERENCES "report" (id) ON DELETE CASCADE
NOT NULL,
    state TEXT DEFAULT 'unresolved' NOT NULL,
    comment TEXT,
    responsible_user INTEGER REFERENCES "user" (id) ON DELETE CASCADE NOT
NULL
);

-- Table: following
CREATE TABLE following (
    user_id INTEGER REFERENCES "user" (id) NOT NULL,
    label_id INTEGER REFERENCES "label" (id) NOT NULL
);

-- Table: about
CREATE TABLE about (
    question_id INTEGER REFERENCES "question" (id) NOT NULL,
    label_id INTEGER REFERENCES "label" (id) NOT NULL
);

-----
-- INDEXES
-----

CREATE INDEX question_score ON question USING btree(nr_likes);
CREATE INDEX question_date ON question USING btree(question_date);
CREATE INDEX answer_score ON answer USING btree(question_id, nr_likes);
CREATE INDEX answer_date ON answer USING btree(question_id, answer_date);
CREATE INDEX comment_date ON comment USING btree(question_id, comment_date);
CREATE INDEX label_popularity ON following USING btree(label_id);
CREATE INDEX question_user ON question USING btree(user_id);
CREATE INDEX answer_user ON answer USING btree(user_id);
CREATE INDEX notification_user_date ON notification USING btree(user_id, date);
CREATE INDEX user_username ON "user" USING hash(username);
CREATE INDEX report_user ON report USING btree(user_id);
CREATE INDEX user_score ON "user" USING btree(score);

CREATE INDEX label_name ON label USING gin(to_tsvector('english', name));
CREATE INDEX question_title ON question USING gist(to_tsvector('english', title));

-----
-- TRIGGERS and UDFs
-----

--Trigger 1
CREATE FUNCTION update_score_question() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT vote.id FROM vote WHERE NEW."vote" = FALSE) THEN
        UPDATE question
        SET nr_dislikes = nr_dislikes+1

```

```

        WHERE NEW.question_id = id;
        UPDATE "user"
        SET score = score-1
        FROM question
        WHERE NEW.question_id = question.id AND question.user_id = "user".id;
    ELSE IF EXISTS (SELECT vote.id FROM vote WHERE NEW."vote" = TRUE) THEN
        UPDATE question
        SET nr_likes = nr_likes+1
        WHERE NEW.question_id = id;
        UPDATE "user"
        SET score = score+1
        FROM question
        WHERE NEW.question_id = question.id AND question.user_id = "user".id;
    END IF;
END IF;
RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER update_score_question
    AFTER INSERT OR UPDATE ON vote
    FOR EACH ROW
    EXECUTE PROCEDURE update_score_question();

--Trigger 2
CREATE FUNCTION update_score_answer() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT vote.id FROM vote WHERE NEW."vote" = FALSE) THEN
        UPDATE answer
        SET nr_dislikes = nr_dislikes+1
        WHERE NEW.answer_id = id;
        UPDATE "user"
        SET score = score-1
        FROM answer
        WHERE NEW.answer_id = answer.id AND answer.user_id = "user".id;
    ELSE IF EXISTS (SELECT vote.id FROM vote WHERE NEW."vote" = TRUE) THEN
        UPDATE answer
        SET nr_likes = nr_likes+1
        WHERE NEW.answer_id = id;
        UPDATE "user"
        SET score = score+1
        FROM answer
        WHERE NEW.answer_id = answer.id AND answer.user_id = "user".id;
    END IF;
END IF;
RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER update_score_answer

```

```
AFTER INSERT OR UPDATE ON vote
FOR EACH ROW
EXECUTE PROCEDURE update_score_answer();

--Trigger 3
CREATE FUNCTION vote_own_question() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT question.user_id
               FROM question
               WHERE NEW.user_id = question.user_id AND NEW.question_id =
question.id) THEN
        RAISE EXCEPTION 'A user cannot vote on his own question';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER vote_own_question
BEFORE INSERT OR UPDATE ON vote
FOR EACH ROW
EXECUTE PROCEDURE vote_own_question();

--Trigger 4
CREATE FUNCTION vote_own_answer() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT answer.user_id
               FROM answer
               WHERE NEW.user_id = answer.user_id AND NEW.answer_id = answer.id)
THEN
        RAISE EXCEPTION 'A user cannot vote on his own answer';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER vote_own_answer
BEFORE INSERT OR UPDATE ON vote
FOR EACH ROW
EXECUTE PROCEDURE vote_own_answer();

--Trigger 5
CREATE FUNCTION answer_date() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT question.question_date
               FROM question
               WHERE NEW.question_id = question.id AND NEW.answer_date <
```

```
question.question_date) THEN
    RAISE EXCEPTION 'The date of an answer cannot be earlier than the date of
its question';
END IF;
RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER answer_date
    BEFORE INSERT OR UPDATE ON answer
    FOR EACH ROW
    EXECUTE PROCEDURE answer_date();

--Trigger 6
CREATE FUNCTION comment_date_answer() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT answer.answer_date
                FROM answer
                WHERE NEW.answer_id = answer.id AND NEW.comment_date <
answer.answer_date) THEN
        RAISE EXCEPTION 'The date of a comment cannot be earlier than the date of
its answer';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER comment_date_answer
    BEFORE INSERT OR UPDATE ON comment
    FOR EACH ROW
    EXECUTE PROCEDURE comment_date_answer();

--Trigger 7
CREATE FUNCTION comment_date_question() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT question.question_date
                FROM question
                WHERE NEW.question_id = question.id AND NEW.comment_date <
question.question_date) THEN
        RAISE EXCEPTION 'The date of a comment cannot be earlier than the date of
its question';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER comment_date_question
```

```
    BEFORE INSERT OR UPDATE ON comment
    FOR EACH ROW
    EXECUTE PROCEDURE comment_date_question();

--Trigger 8
CREATE FUNCTION vote_once() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF EXISTS (SELECT * FROM vote
               WHERE ((NEW.user_id = vote.user_id AND NEW.question_id =
vote.question_id) OR
                     (NEW.user_id = vote.user_id AND NEW.answer_id =
vote.answer_id))) THEN
        RAISE EXCEPTION 'An element can only be voted once by the same user';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER vote_once
    BEFORE INSERT ON vote
    FOR EACH ROW
    EXECUTE PROCEDURE vote_once();

--Trigger 9
CREATE FUNCTION report_status_responsible() RETURNS TRIGGER AS
$BODY$
BEGIN
    IF NOT EXISTS (SELECT user_management.user_id
                   FROM user_management
                   WHERE ((user_management.status = 'moderator' OR
user_management.status = 'administrator')
                        AND
                        (user_management.user_id = NEW.responsible_user))) THEN
        RAISE EXCEPTION 'A report can only be handled by an administrator or
moderator';
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER report_status_responsible
    BEFORE INSERT OR UPDATE ON report_status
    FOR EACH ROW
    EXECUTE PROCEDURE report_status_responsible();

--Trigger 10
CREATE FUNCTION marked_answer() RETURNS TRIGGER AS
$BODY$
```

```

BEGIN
    IF EXISTS (SELECT marked_answer FROM answer WHERE answer.id = NEW.id AND
NEW.marked_answer = TRUE) THEN
        UPDATE answer
        SET marked_answer = FALSE
        WHERE (answer.question_id = NEW.question_id AND answer.id != NEW.id);
    END IF;
    RETURN NEW;
END
$BODY$
LANGUAGE plpgsql;

CREATE TRIGGER marked_answer
    AFTER UPDATE OF marked_answer ON answer
    FOR EACH ROW
    EXECUTE PROCEDURE marked_answer();

-----
-- TRANSACTIONS
-----
--Trigger 01
BEGIN TRANSACTION;
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ
    INSERT INTO "user" (first_name, last_name, email, bio, username, password)
        VALUES ('antonio', 'aparicio', 'toutolo@cenass.com', 'nao tenho mts
amigos', 'tonicio', 'sha256woendo+2ph«09328');

    INSERT INTO user_management (user_id)
        VALUES (currval('user_id_seq'));

COMMIT;

--Trigger 02
BEGIN TRANSACTION;
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ
    INSERT INTO report (user_id, question_id, answer_id, comment_id)
        VALUES (1, 2, NULL, NULL);

    INSERT INTO report_status (report_id, comment, responsible_user)
        VALUES (currval('report_id_seq'), 'tou tolo', 2);

COMMIT;

-----
-- end
-----

```

5.2. Database population

```
-- user
```

```
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (1, 'Pedro', 'Dantas', 'pedrodantas@gmail.com', 'Curious I.T
Student', 'pedrodantas', 'password', 30);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (2, 'Nuno', 'Cardoso', 'nunocardas@gmail.com', 'Curious I.T Student
trying to help others', 'nunocardoso', 'password', 77);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (3, 'Eduardo', 'Macedo', 'eduardomacedo@gmail.com', '21 years old,
I.T student @ FEUP', 'edumacedo', 'password', 56);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (4, 'Roberto', 'Mourato', 'robertomourato@gmail.com', 'I.T Student
from Portugal trying to help others', 'robertomourato', 'password', 50);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (5, 'Francisco', 'Noval', 'xicotruques@gmail.com', 'Handball coach,
20y', 'truques', 'password', 12);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (6, 'Gonçalo', 'Nogueira', 'salinhonogueira@gmail.com', 'Try a
little harder to be a little better', 'salinho', 'password', 70);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (7, 'Joao', 'Torgal', 'joaotorgal@gmail.com', '21y, bl4z3 it',
'torgal', 'password', 23);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (8, 'Luis', 'Fernandes', 'luisfernandes@gmail.com', 'Engineering
student, NEEGIUM', 'luisfernandes', 'password', 120);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (9, 'Raquel', 'Fonseca', 'carlafonseca@gmail.com', '21y, FEUP,
Portugal', 'raquelfonseca', 'password', 65);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (10, 'Catarina', 'Silva', 'ccsilva@gmail.com', 'Former I.T Student,
now with a degree in Mathematics', 'ccsilva', 'password', 34);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (11, 'Vitor', 'Goncalves', 'vitorturrinheira@gmail.com', '20y, FEUP,
Pardelhas', 'vitorturrinheira', 'password', 42);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (12, 'Pedro', 'Dantas', 'pedrodantas@gmail.com', 'Curious I.T
Student', 'pedrodantas1', 'password', -7);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (13, 'Nuno', 'Cardoso', 'nunocardas@gmail.com', 'Curious I.T Student
trying to help others', 'nunocardoso1', 'password', 77);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (14, 'Eduardo', 'Macedo', 'eduardomacedo@gmail.com', '21 years old,
I.T student @ FEUP', 'edumacedo1', 'password', 56);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (15, 'Roberto', 'Mourato', 'robertomourato@gmail.com', 'I.T Student
from Portugal trying to help others', 'robertomourato1', 'password', 50);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (16, 'Francisco', 'Noval', 'xicotruques@gmail.com', 'Handball coach,
20y', 'truques1', 'password', 12);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (17, 'Gonçalo', 'Nogueira', 'salinhonogueira@gmail.com', 'Try a
little harder to be a little better', 'salinho1', 'password', 70);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (18, 'Joao', 'Torgal', 'joaotorgal@gmail.com', '21y, bl4z3 it',
'torgal1', 'password', 23);
```

```
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (19, 'Luis', 'Fernandes', 'luisfernandes@gmail.com', 'Engineering
student, NEEGIUM', 'luisfernandes1', 'password', 120);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (20, 'Raquel', 'Fonseca', 'carlafonseca@gmail.com', '21y, FEUP,
Portugal', 'raquelfonseca1', 'password', 65);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (21, 'Vitor', 'Goncalves', 'vitorturpinheira@gmail.com', '20y, FEUP,
Pardelhas', 'vitorturpinheira2', 'password', 42);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (22, 'Pedro', 'Dantas', 'pedrodantas@gmail.com', 'Curious I.T
Student', 'pedrodantas2', 'password', 99);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (23, 'Nuno', 'Cardoso', 'nunocardas@gmail.com', 'Curious I.T Student
trying to help others', 'nunocardoso2', 'password', 77);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (24, 'Eduardo', 'Macedo', 'eduardomacedo@gmail.com', '21 years old,
I.T student @ FEUP', 'edumacedo2', 'password', -56);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (25, 'Roberto', 'Mourato', 'robertomourato@gmail.com', 'I.T Student
from Portugal trying to help others', 'robertomourato2', 'password', 50);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (26, 'Francisco', 'Noval', 'xicotruques@gmail.com', 'Handball coach,
20y', 'truques2', 'password', 12);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (27, 'Gonçalo', 'Nogueira', 'salinhonogueira@gmail.com', 'Try a
little harder to be a little better', 'salinho2', 'password', 70);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (28, 'Joao', 'Torgal', 'joaotorgal@gmail.com', '21y, bl4z3 it',
'torgal2', 'password', 23);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (29, 'Luis', 'Fernandes', 'luisfernandes@gmail.com', 'Engineering
student, NEEGIUM', 'luisfernandes2', 'password', 120);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (30, 'Raquel', 'Fonseca', 'carlafonseca@gmail.com', '21y, FEUP,
Portugal', 'raquelfonseca2', 'password', 65);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (31, 'Vitor', 'Goncalves', 'vitorturpinheira@gmail.com', '20y, FEUP,
Pardelhas', 'vitorturpinheira3', 'password', 42);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (32, 'Pedro', 'Dantas', 'pedrodantas@gmail.com', 'Curious I.T
Student', 'pedrodantas3', 'password', 7);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (33, 'Nuno', 'Cardoso', 'nunocardas@gmail.com', 'Curious I.T Student
trying to help others', 'nunocardoso3', 'password', 77);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (34, 'Eduardo', 'Macedo', 'eduardomacedo@gmail.com', '21 years old,
I.T student @ FEUP', 'edumacedo3', 'password', 56);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (35, 'Roberto', 'Mourato', 'robertomourato@gmail.com', 'I.T Student
from Portugal trying to help others', 'robertomourato3', 'password', 50);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (36, 'Francisco', 'Noval', 'xicotruques@gmail.com', 'Handball coach,
20y', 'truques3', 'password', 12);
```



```
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (37, 'Gonçalo', 'Nogueira', 'salinhonogueira@gmail.com', 'Try a
little harder to be a little better', 'salinho3', 'password', -27);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (38, 'Joao', 'Torgal', 'joaotorgal@gmail.com', '21y, bl4z3 it',
'torgal3', 'password', 23);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (39, 'Luis', 'Fernandes', 'luisfernandes@gmail.com', 'Engineering
student, NEEGIUM', 'luisfernandes3', 'password', 120);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (40, 'Raquel', 'Fonseca', 'carlafonseca@gmail.com', '21y, FEUP,
Portugal', 'raquelfonseca3', 'password', 65);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (41, 'Vitor', 'Goncalves', 'vitorturrinheira@gmail.com', '20y, FEUP,
Pardelhas', 'vitorturrinheira4', 'password', 42);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (42, 'Pedro', 'Dantas', 'pedrodantas@gmail.com', 'Curious I.T
Student', 'pedrodantas4', 'password', 89);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (43, 'Nuno', 'Cardoso', 'nunocardas@gmail.com', 'Curious I.T Student
trying to help others', 'nunocardoso4', 'password', 77);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (44, 'Eduardo', 'Macedo', 'eduardomacedo@gmail.com', '21 years old,
I.T student @ FEUP', 'edumacedo4', 'password', -56);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (45, 'Roberto', 'Mourato', 'robertomourato@gmail.com', 'I.T Student
from Portugal trying to help others', 'robertomourato4', 'password', 50);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (46, 'Francisco', 'Noval', 'xicotruques@gmail.com', 'Handball coach,
20y', 'truques4', 'password', 12);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (47, 'Gonçalo', 'Nogueira', 'salinhonogueira@gmail.com', 'Try a
little harder to be a little better', 'salinho4', 'password', 70);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (48, 'Joao', 'Torgal', 'joaotorgal@gmail.com', '21y, bl4z3 it',
'torgal4', 'password', 23);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (49, 'Luis', 'Fernandes', 'luisfernandes@gmail.com', 'Engineering
student, NEEGIUM', 'luisfernandes4', 'password', 120);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (50, 'Raquel', 'Fonseca', 'carlafonseca@gmail.com', '21y, FEUP,
Portugal', 'raquelfonseca4', 'password', 65);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (51, 'Vitor', 'Goncalves', 'vitorturrinheira@gmail.com', '20y, FEUP,
Pardelhas', 'vitorturrinheira5', 'password', 42);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (52, 'Pedro', 'Dantas', 'pedrodantas@gmail.com', 'Curious I.T
Student', 'pedrodantas5', 'password', 57);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (53, 'Nuno', 'Cardoso', 'nunocardas@gmail.com', 'Curious I.T Student
trying to help others', 'nunocardoso5', 'password', 77);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (54, 'Eduardo', 'Macedo', 'eduardomacedo@gmail.com', '21 years old,
I.T student @ FEUP', 'edumacedo5', 'password', -56);
```

```
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (55, 'Roberto', 'Mourato', 'robertomourato@gmail.com', 'I.T Student
from Portugal trying to help others', 'robertomourato5', 'password', 50);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (56, 'Francisco', 'Noval', 'xicotruques@gmail.com', 'Handball coach,
20y', 'truques5', 'password', 12);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (57, 'Gonçalo', 'Nogueira', 'salinhonogueira@gmail.com', 'Try a
little harder to be a little better', 'salinho5', 'password', 70);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (58, 'Joao', 'Torgal', 'joaotorgal@gmail.com', '21y, bl4z3 it',
'torgal5', 'password', 23);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (59, 'Luis', 'Fernandes', 'luisfernandes@gmail.com', 'Engineering
student, NEEGIUM', 'luisfernandes5', 'password', 120);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (60, 'Raquel', 'Fonseca', 'carlafonseca@gmail.com', '21y, FEUP,
Portugal', 'raquelfonseca5', 'password', 65);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (61, 'Vitor', 'Goncalves', 'vitorturpinheira@gmail.com', '20y, FEUP,
Pardelhas', 'vitorturpinheira6', 'password', 42);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (62, 'Pedro', 'Dantas', 'pedrodantas@gmail.com', 'Curious I.T
Student', 'pedrodantas6', 'password', 33);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (63, 'Nuno', 'Cardoso', 'nunocardas@gmail.com', 'Curious I.T Student
trying to help others', 'nunocardoso6', 'password', 77);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (64, 'Eduardo', 'Macedo', 'eduardomacedo@gmail.com', '21 years old,
I.T student @ FEUP', 'edumacedo6', 'password', -56);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (65, 'Roberto', 'Mourato', 'robertomourato@gmail.com', 'I.T Student
from Portugal trying to help others', 'robertomourato6', 'password', 50);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (66, 'Francisco', 'Noval', 'xicotruques@gmail.com', 'Handball coach,
20y', 'truques6', 'password', 12);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (67, 'Gonçalo', 'Nogueira', 'salinhonogueira@gmail.com', 'Try a
little harder to be a little better', 'salinho6', 'password', 70);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (68, 'Joao', 'Torgal', 'joaotorgal@gmail.com', '21y, bl4z3 it',
'torgal6', 'password', -23);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (69, 'Luis', 'Fernandes', 'luisfernandes@gmail.com', 'Engineering
student, NEEGIUM', 'luisfernandes6', 'password', 120);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (70, 'Raquel', 'Fonseca', 'carlafonseca@gmail.com', '21y, FEUP,
Portugal', 'raquelfonseca6', 'password', 65);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (71, 'Vitor', 'Goncalves', 'vitorturpinheira@gmail.com', '20y, FEUP,
Pardelhas', 'vitorturpinheira7', 'password', 42);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (72, 'Pedro', 'Dantas', 'pedrodantas@gmail.com', 'Curious I.T
Student', 'pedrodantas7', 'password', 12);
```

```
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (73, 'Nuno', 'Cardoso', 'nunocardas@gmail.com', 'Curious I.T Student
trying to help others', 'nunocardoso7', 'password', 77);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (74, 'Eduardo', 'Macedo', 'eduardomacedo@gmail.com', '21 years old,
I.T student @ FEUP', 'edumacedo7', 'password', -56);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (75, 'Roberto', 'Mourato', 'robertomourato@gmail.com', 'I.T Student
from Portugal trying to help others', 'robertomourato7', 'password', 50);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (76, 'Francisco', 'Noval', 'xicotruques@gmail.com', 'Handball coach,
20y', 'truques7', 'password', 12);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (77, 'Gonçalo', 'Nogueira', 'salinhonogueira@gmail.com', 'Try a
little harder to be a little better', 'salinho7', 'password', 70);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (78, 'Joao', 'Torgal', 'joaotorgal@gmail.com', '21y, bl4z3 it',
'torgal7', 'password', 23);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (79, 'Luis', 'Fernandes', 'luisfernandes@gmail.com', 'Engineering
student, NEEGIUM', 'luisfernandes7', 'password', 120);
insert into 'user' (id, first_name, last_name, email, bio, username, password,
score) values (80, 'Raquel', 'Fonseca', 'carlafonseca@gmail.com', '21y, FEUP,
Portugal', 'raquelfonseca7', 'password', 65);
```

```
-- label
```

```
insert into 'label' (id, name) values (1, 'sports');
insert into 'label' (id, name) values (2, 'philosophy');
insert into 'label' (id, name) values (3, 'music');
insert into 'label' (id, name) values (4, 'science');
insert into 'label' (id, name) values (5, 'cinema');
insert into 'label' (id, name) values (6, 'photography');
insert into 'label' (id, name) values (7, 'computers');
insert into 'label' (id, name) values (8, 'programming');
insert into 'label' (id, name) values (9, 'tv');
insert into 'label' (id, name) values (10, 'gaming');
insert into 'label' (id, name) values (11, 'macOS');
insert into 'label' (id, name) values (12, 'windows');
insert into 'label' (id, name) values (13, 'linux');
insert into 'label' (id, name) values (14, 'geography');
insert into 'label' (id, name) values (15, 'coronavirus');
insert into 'label' (id, name) values (16, 'jazz');
insert into 'label' (id, name) values (17, 'politics');
insert into 'label' (id, name) values (18, 'football');
insert into 'label' (id, name) values (19, 'cr7');
insert into 'label' (id, name) values (20, 'health');
insert into 'label' (id, name) values (21, 'technology');
insert into 'label' (id, name) values (22, 'psychology');
insert into 'label' (id, name) values (23, 'java');
insert into 'label' (id, name) values (24, 'c++');
insert into 'label' (id, name) values (25, 'python');
insert into 'label' (id, name) values (26, 'hockey');
insert into 'label' (id, name) values (27, 'basketball');
```

```
insert into 'label' (id, name) values (28, 'rock');
insert into 'label' (id, name) values (29, 'pop');
insert into 'label' (id, name) values (30, 'english');

--user_management
insert into 'user_management' (id, status, user_id) values (1, 'administrator',
1);
insert into 'user_management' (id, status, user_id) values (2, 'administrator',
2);
insert into 'user_management' (id, status, user_id) values (3, 'administrator',
3);
insert into 'user_management' (id, status, user_id) values (4, 'administrator',
4);
insert into 'user_management' (id, status, user_id) values (5, 'moderator', 5);
insert into 'user_management' (id, status, user_id) values (6, 'moderator', 6);
insert into 'user_management' (id, status, user_id) values (7, 'moderator', 7);
insert into 'user_management' (id, status, user_id) values (8, 'moderator', 8);
insert into 'user_management' (id, status, user_id) values (9, 'moderator', 9);
insert into 'user_management' (id, status, user_id) values (10, 'moderator', 10);

insert into 'user_management' (id, status, user_id) values (11, 'user', 11);
insert into 'user_management' (id, status, user_id) values (12, 'user', 12);
insert into 'user_management' (id, status, user_id) values (13, 'user', 13);
insert into 'user_management' (id, status, user_id) values (14, 'user', 14);
insert into 'user_management' (id, status, user_id) values (15, 'user', 15);
insert into 'user_management' (id, status, user_id) values (16, 'user', 16);
insert into 'user_management' (id, status, user_id) values (17, 'user', 17);
insert into 'user_management' (id, status, user_id) values (18, 'user', 18);
insert into 'user_management' (id, status, user_id) values (19, 'user', 19);
insert into 'user_management' (id, status, user_id) values (20, 'user', 20);
insert into 'user_management' (id, status, user_id) values (21, 'user', 21);
insert into 'user_management' (id, status, user_id) values (22, 'user', 22);
insert into 'user_management' (id, status, user_id) values (23, 'user', 23);
insert into 'user_management' (id, status, user_id) values (24, 'user', 24);
insert into 'user_management' (id, status, user_id) values (25, 'user', 25);
insert into 'user_management' (id, status, user_id) values (26, 'user', 26);
insert into 'user_management' (id, status, user_id) values (27, 'user', 27);
insert into 'user_management' (id, status, user_id) values (28, 'user', 28);
insert into 'user_management' (id, status, user_id) values (29, 'user', 29);
insert into 'user_management' (id, status, user_id) values (30, 'user', 30);
insert into 'user_management' (id, status, user_id) values (31, 'user', 31);
insert into 'user_management' (id, status, user_id) values (32, 'user', 32);
insert into 'user_management' (id, status, user_id) values (33, 'user', 33);
insert into 'user_management' (id, status, user_id) values (34, 'user', 34);
insert into 'user_management' (id, status, user_id) values (35, 'user', 35);
insert into 'user_management' (id, status, user_id) values (36, 'user', 36);
insert into 'user_management' (id, status, user_id) values (37, 'user', 37);
insert into 'user_management' (id, status, user_id) values (38, 'user', 38);
insert into 'user_management' (id, status, user_id) values (39, 'user', 39);
insert into 'user_management' (id, status, user_id) values (40, 'user', 40);
insert into 'user_management' (id, status, user_id) values (41, 'user', 41);
insert into 'user_management' (id, status, user_id) values (42, 'user', 42);
insert into 'user_management' (id, status, user_id) values (43, 'user', 43);
```



```
--following
```

37 / 53

[illegible]

[illegible]

[illegible]

41 / 53

```
insert into 'following' (user_id, label_id) values (73, 1);
insert into 'following' (user_id, label_id) values (73, 2);
insert into 'following' (user_id, label_id) values (73, 4);
insert into 'following' (user_id, label_id) values (73, 6);
insert into 'following' (user_id, label_id) values (74, 8);
insert into 'following' (user_id, label_id) values (74, 10);
insert into 'following' (user_id, label_id) values (74, 12);
insert into 'following' (user_id, label_id) values (75, 13);
insert into 'following' (user_id, label_id) values (75, 14);
insert into 'following' (user_id, label_id) values (75, 15);
insert into 'following' (user_id, label_id) values (76, 16);
insert into 'following' (user_id, label_id) values (76, 17);
insert into 'following' (user_id, label_id) values (76, 17);
insert into 'following' (user_id, label_id) values (76, 18);
insert into 'following' (user_id, label_id) values (77, 19);
insert into 'following' (user_id, label_id) values (77, 20);
insert into 'following' (user_id, label_id) values (78, 21);
insert into 'following' (user_id, label_id) values (78, 22);
insert into 'following' (user_id, label_id) values (78, 23);
insert into 'following' (user_id, label_id) values (79, 24);
insert into 'following' (user_id, label_id) values (79, 25);
insert into 'following' (user_id, label_id) values (79, 26);
insert into 'following' (user_id, label_id) values (79, 27);
insert into 'following' (user_id, label_id) values (79, 28);
insert into 'following' (user_id, label_id) values (80, 29);
insert into 'following' (user_id, label_id) values (80, 30);

-- questions
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (1, 4, 'Name some great method actors', 'What are some
fascinating examples of how an actor got into character for a movie role?', 32, 2,
'2019-12-03');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (2, 2, 'Are intelligent and wise people the same?', 'What is
the difference between an intelligent person and a wise person?', 40, 5, '2019-11-
24');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (3, 1, 'Who are the biggest jerks in music?', 'Can you name
some musicians that are particularly obnoxious? Who are the biggest douchebags in
the music world?', 10, 5, '2020-01-2');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (4, 3, 'Is there a difference between every day and
everyday?', 'This an english language question that has always tormented me. I
never know which of the two should I use, or if there is any difference at all.',
100, 21, '2019-04-18');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (5, 6, 'Will science reconcile with religion in the
future?', 'Just wondering if I could ever pair the certainty of science with my
love for the lord.', 22, 33, '2020-02-02');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (6, 7, 'Can someone explain to me the AirPods hype?', 'Why
are Apples AirPods so hyped up when fake Airpods work just the same?', 33, 22,
'2019-09-30');
```

```
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (7, 5, 'Running a program from within java code', 'What is
the simplest way to call a program from with a piece of Java code? (The program I
want to run is aiSee and it can be run from command line or from Windows GUI; and
I am on Vista but the code will also be run on Linux systems).', 67, 5, '2020-11-
04');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (8, 8, 'What is the best TV series and why?', 'I would like
to really get into a tv show, but I only want to watch the best of the best.', 20,
5, '2020-03-20');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (9, 10, 'What country's flag is the most controversial?',
'This may seem odd, but I would like to hear the internets opinion.', 132, 32,
'2019-10-19');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (10, 9, 'Who is handling the coronavirus outbreak better,
the United States or Canada?', 'Im a proud American and Ill be damned if Canada is
handeling it better >:(', 30, 62, '2020-03-03');

insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (11, 14, 'Name some great method actors', 'What are some
fascinating examples of how an actor got into character for a movie role?', 32, 2,
'2019-12-03');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (12, 12, 'Are intelligent and wise people the same?', 'What
is the difference between an intelligent person and a wise person?', 40, 5, '2019-
11-24');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (13, 11, 'Who are the biggest jerks in music?', 'Can you
name some musicians that are particularly obnoxious? Who are the biggest
douchebags in the music world?', 10, 5, '2020-01-2');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (14, 13, 'Is there a difference between every day and
everyday?', 'This an english language question that has always tormented me. I
never know which of the two should I use, or if there is any difference at all.',
100, 21, '2019-04-18');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (15, 16, 'Will science reconcile with religion in the
future?', 'Just wondering if I could ever pair the certainty of science with my
love for the lord.', 22, 33, '2020-02-02');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (16, 17, 'Can someone explain to me the AirPods hype?', 'Why
are Apples AirPods so hyped up when fake Airpods work just the same?', 33, 22,
'2019-09-30');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (17, 15, 'Running a program from within java code', 'What is
the simplest way to call a program from with a piece of Java code? (The program I
want to run is aiSee and it can be run from command line or from Windows GUI; and
I am on Vista but the code will also be run on Linux systems).', 67, 5, '2020-11-
04');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (18, 18, 'What is the best TV series and why?', 'I would
like to really get into a tv show, but I only want to watch the best of the
best.', 20, 5, '2020-03-20');
```

```
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (19, 20, 'What country's flag is the most controversial?',
'This may seem odd, but I would like to hear the internets opinion.', 132, 32,
'2019-10-19');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (20, 19, 'Who is handling the coronavirus outbreak better,
the United States or Canada?', 'Im a proud American and Ill be damned if Canada is
handeling it better >:(', 30, 62, '2020-03-03');

insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (21, 24, 'Name some great method actors', 'What are some
fascinating examples of how an actor got into character for a movie role?', 32, 2,
'2019-12-03');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (22, 22, 'Are intelligent and wise people the same?', 'What
is the difference between an intelligent person and a wise person?', 40, 5, '2019-
11-24');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (23, 21, 'Who are the biggest jerks in music?', 'Can you
name some musicians that are particularly obnoxious? Who are the biggest
douchebags in the music world?', 10, 5, '2020-01-2');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (24, 23, 'Is there a difference between every day and
everyday?', 'This an english language question that has always tormented me. I
never know which of the two should I use, or if there is any difference at all.',
100, 21, '2019-04-18');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (25, 26, 'Will science reconcile with religion in the
future?', 'Just wondering if I could ever pair the certainty of science with my
love for the lord.', 22, 33, '2020-02-02');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (26, 27, 'Can someone explain to me the AirPods hype?', 'Why
are Apples AirPods so hyped up when fake Airpods work just the same?', 33, 22,
'2019-09-30');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (27, 25, 'Running a program from within java code', 'What is
the simplest way to call a program from with a piece of Java code? (The program I
want to run is aiSee and it can be run from command line or from Windows GUI; and
I am on Vista but the code will also be run on Linux systems).', 67, 5, '2020-11-
04');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (28, 28, 'What is the best TV series and why?', 'I would
like to really get into a tv show, but I only want to watch the best of the
best.', 20, 5, '2020-03-20');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (29, 30, 'What country's flag is the most controversial?',
'This may seem odd, but I would like to hear the internets opinion.', 132, 32,
'2019-10-19');
insert into 'question' (id, user_id, title, description, nr_likes, nr_dislikes,
question_date) values (30, 29, 'Who is handling the coronavirus outbreak better,
the United States or Canada?', 'Im a proud American and Ill be damned if Canada is
handeling it better >:(', 30, 62, '2020-03-03');
```

```
--answers
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (1, 1, 1, '2019-12-13', 'Undoubtedly the worst form of over-the-top method acting is the kind that a) makes you look like an ass, and b) was for a terrible film. We've already shared some of the stories of Jared Leto's Joker-prep for Suicide Squad, in which the actor famously tormented his co-workers for the sake of a half-written and pandering story. But there's so much more.', 30, 32, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (2, 3, 2, '2019-12-01', 'An intelligent person knows so much that it knows nothing, leaves you hanging upside down, mouthing knowledge as your heart falls out of your mouth - Anne.
```

But,

A wise man knows when to use his intelligence. He knows when to be silent.', 67, 4, FALSE);

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (3, 4, 3, '2020-01-12', 'There are lots of jerks in music. People like Gene Simmons and Morrissey are well-known jerks.', 10, 0, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (4, 2, 4, '2019-05-01', 'I'm glad you realize there is a difference, because many people don't.
```

When you do something daily, let's say, then you do it every day. Every. Day. Two words. Two separate words. "Every day" means, refers to, and includes all days, just like "every cantaloupe" means, refers to, and includes all cantaloupes. You get the idea.

When something is commonplace, ordinary, mundane, routine, average, run-of-the-mill, plain, or typical, then it can be accurately described as everyday. One word. One. "Everyday" is an adjective, a descriptor.', 100, 4, TRUE);

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (5, 5, 5, '2020-02-05', 'There never was a breach. BOTH two sides of the SAME coin. Science asks "how", religion asks "why".', 43, 15, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (6, 6, 6, '2019-10-01', 'First of all, Apple AirPods are not hype, they are the real deal with near unparalleled Bluetooth pairing and connectivity, excellent sound, good fit and reasonable cost. AirPods 2 are quite sufficient and nearly as good all around as the Pro version. You can spend MORE for better sound, not necessarily better Bluetooth. The only drawback to iPhone/AirPods connection is that Apple only offers 1 Bluetooth codec, and not AptX or AptX HD. Knockoffs of nearly anything are not as satisfying, or as quality as the Original!
```

There are other earbuds out there, but you will risk the one element that is essential: Bluetooth connectivity, and pairing, regardless of upgrades.', 54, 13, TRUE);

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (7, 7, 7, '2020-11-05', 'Take a look at Process and Runtime classes. Keep in mind that what you are trying to accomplish is probably not platform independent.', 43, 12, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (8, 9, 8, '2020-03-26', 'BREAKING BAD! This is the best TV show on earth. It builds up slowly but methodically to expand over several seasons. I was amazed by the storytelling. Every piece slowly falls into place to create a grand finale that has been mastered to blow your mind away.',
```



```
31, 1, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (9, 8, 9, '2019-11-06', 'Japans rising sun
flag. The Rising Sun Flag is the military flag of Japan. It was used as the ensign
of the Imperial Japanese Navy and the war flag of the Imperial Japanese Army until
the end of World War II. It is also presently the ensign of the Japan Maritime
Self-Defense Force and the war flag of the Japan Ground Self-Defense Force. It is
also waved during the Japanese New Year and in sporting events. The design is
similar to the flag of Japan in that it has a red circle close to the middle
signifying the sun, the difference being the addition of extra sun rays (16 for
the ensign) exemplifying the name of Japan as "The Land of the Rising Sun". The
flag was used in overseas actions from the Meiji period to World War II. When
Japan was defeated in August 1945, the flag was banned by Allied Occupation
authorities. However, with the re-establishment of a Self-Defense Force, the flag
was re-adopted in 1954. The flag with 16 rays is today the ensign of the Maritime
Self-Defense Force while the Ground Self-Defense Force uses an 8 ray version. This
flag is often considered offensive in countries which were victims of Japanese
aggression, particularly China and Korea, where it is considered as a symbol of
Japanese imperialism.', 55, 12, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (10, 10, 10, '2020-03-30', 'it would depend
upon your definition of better, is it better to have a low figure because you have
not tested people at the appropriate time or have a higher figure because you have
accurately tested many more people who may have the infection so naturally you
have a more accurate understanding of the scale of the issue?
perhaps you consider it better to look at the number of confirmed cases that
subsequently died as a result of the infection either directly or indirectly?
Canada has done the tests and can track those patients being treated for serious
reaction to the infection. The USA has not done the testing to the same level as
Canada so their results are patchy at best.
Canada has a health service that everyone can access without fear of bankruptcy,
lifelong debit or needing to sell there homes, so more people are willing to find
out whether they are infected, in the USA due to their healthcare system many
poorer people try not to go to the Doctor as they cannot afford it, so their
deaths could well be misclassified as there is no test data or possibly post
mortem to accurately identify the cause of death.
In either case I would say that Canada is beating the USA hands down.', 21, 3,
FALSE);

insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (11, 11, 11, '2019-12-13', 'Undoubtedly the
worst form of over-the-top method acting is the kind that a) makes you look like
an ass, and b) was for a terrible film. We've already shared some of the stories
of Jared Leto's Joker-prep for Suicide Squad, in which the actor famously
tormented his co-workers for the sake of a half-written and pandering story. But
there's so much more.', 30, 32, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (12, 13, 12, '2019-12-01', 'An intelligent
person knows so much that it knows nothing, leaves you hanging upside down,
mouthing knowledge as your heart falls out of your mouth - Anne.
But,
A wise man knows when to use his intelligence. He knows when to be silent.', 67,
4, FALSE);
```

```

insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (13, 14, 13, '2020-01-12', 'There are lots of
jerks in music. People like Gene Simmons and Morrissey are well-known jerks.', 10,
0, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (14, 12, 14, '2019-05-01', 'I'm glad you
realize there is a difference, because many people don't.
When you do something daily, let's say, then you do it every day. Every. Day. Two
words. Two separate words. "Every day" means, refers to, and includes all days,
just like "every cantaloupe" means, refers to, and includes all cantaloupes. You
get the idea.
When something is commonplace, ordinary, mundane, routine, average, run-of-the-
mill, plain, or typical, then it can be accurately described as everyday. One
word. One. "Everyday" is an adjective, a descriptor.', 100, 4, TRUE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (15, 15, 15, '2020-02-05', 'There never was a
breach. BOTH two sides of the SAME coin. Science asks "how", religion asks
"why".', 43, 15, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (16, 16, 16, '2019-10-01', 'First of all, Apple
AirPods are not hype, they are the real deal with near unparalleled Bluetooth
pairing and connectivity, excellent sound, good fit and reasonable cost. AirPods 2
are quite sufficient and nearly as good all around as the Pro version. You can
spend MORE for better sound, not necessarily better Bluetooth. The only drawback
to iPhone/AirPods connection is that Apple only offers 1 Bluetooth codec, and not
AptX or AptX HD. Knockoffs of nearly anything are not as satisfying, or as quality
as the Original!
There are other earbuds out there, but you will risk the one element that is
essential: Bluetooth connectivity, and pairing, regardless of upgrades.', 54, 13,
TRUE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (17, 17, 17, '2020-11-05', 'Take a look at
Process and Runtime classes. Keep in mind that what you are trying to accomplish
is probably not platform independent.', 43, 12, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (18, 19, 18, '2020-03-26', 'BREAKING BAD! This
is the best TV show on earth. It builds up slowly but methodically to expand over
several seasons. I was amazed by the storytelling. Every piece slowly falls into
place to create a grand finale that has been mastered to blow your mind away.',
31, 1, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (19, 18, 19, '2019-11-06', 'Japans rising sun
flag. The Rising Sun Flag is the military flag of Japan. It was used as the ensign
of the Imperial Japanese Navy and the war flag of the Imperial Japanese Army until
the end of World War II. It is also presently the ensign of the Japan Maritime
Self-Defense Force and the war flag of the Japan Ground Self-Defense Force. It is
also waved during the Japanese New Year and in sporting events. The design is
similar to the flag of Japan in that it has a red circle close to the middle
signifying the sun, the difference being the addition of extra sun rays (16 for
the ensign) exemplifying the name of Japan as "The Land of the Rising Sun". The
flag was used in overseas actions from the Meiji period to World War II. When
Japan was defeated in August 1945, the flag was banned by Allied Occupation
authorities. However, with the re-establishment of a Self-Defense Force, the flag
was re-adopted in 1954. The flag with 16 rays is today the ensign of the Maritime

```

Self-Defense Force while the Ground Self-Defense Force uses an 8 ray version. This flag is often considered offensive in countries which were victims of Japanese aggression, particularly China and Korea, where it is considered as a symbol of Japanese imperialism.', 55, 12, FALSE);

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (20, 20, 20, '2020-03-30', 'it would depend upon your definition of better, is it better to have a low figure because you have not tested people at the appropriate time or have a higher figure because you have accurately tested many more people who may have the infection so naturally you have a more accurate understanding of the scale of the issue?
```

```
perhaps you consider it better to look at the number of confirmed cases that subsequently died as a result of the infection either directly or indirectly?
```

Canada has done the tests and can track those patients being treated for serious reaction to the infection. The USA has not done the testing to the same level as Canada so their results are patchy at best.

Canada has a health service that everyone can access without fear of bankruptcy, lifelong debt or needing to sell their homes, so more people are willing to find out whether they are infected, in the USA due to their healthcare system many poorer people try not to go to the Doctor as they cannot afford it, so their deaths could well be misclassified as there is no test data or possibly post mortem to accurately identify the cause of death.

```
In either case I would say that Canada is beating the USA hands down.', 21, 3, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (21, 21, 21, '2019-12-13', 'Undoubtedly the worst form of over-the-top method acting is the kind that a) makes you look like an ass, and b) was for a terrible film. We've already shared some of the stories of Jared Leto's Joker-prep for Suicide Squad, in which the actor famously tormented his co-workers for the sake of a half-written and pandering story. But there's so much more.', 30, 32, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (22, 23, 22, '2019-12-01', 'An intelligent person knows so much that it knows nothing, leaves you hanging upside down, mouthing knowledge as your heart falls out of your mouth - Anne.
```

But,

```
A wise man knows when to use his intelligence. He knows when to be silent.', 67, 4, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (23, 24, 23, '2020-01-12', 'There are lots of jerks in music. People like Gene Simmons and Morrissey are well-known jerks.', 10, 0, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (24, 22, 24, '2019-05-01', 'I'm glad you realize there is a difference, because many people don't.
```

When you do something daily, let's say, then you do it every day. Every. Day. Two words. Two separate words. "Every day" means, refers to, and includes all days, just like "every cantaloupe" means, refers to, and includes all cantaloupes. You get the idea.

When something is commonplace, ordinary, mundane, routine, average, run-of-the-mill, plain, or typical, then it can be accurately described as everyday. One word. One. "Everyday" is an adjective, a descriptor.', 100, 4, TRUE);

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes, nr_dislikes, marked_answer) values (25, 25, 25, '2020-02-05', 'There never was a
```



```

breach. BOTH two sides of the SAME coin. Science asks "how", religion asks
"why".', 43, 15, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (26, 26, 26, '2019-10-01', 'First of all, Apple
AirPods are not hype, they are the real deal with near unparalleled Bluetooth
pairing and connectivity, excellent sound, good fit and reasonable cost. AirPods 2
are quite sufficient and nearly as good all around as the Pro version. You can
spend MORE for better sound, not necessarily better Bluetooth. The only drawback
to iPhone/AirPods connection is that Apple only offers 1 Bluetooth codec, and not
AptX or AptX HD. Knockoffs of nearly anything are not as satisfying, or as quality
as the Original!
There are other earbuds out there, but you will risk the one element that is
essential: Bluetooth connectivity, and pairing, regardless of upgrades.', 54, 13,
TRUE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (27, 27, 27, '2020-11-05', 'Take a look at
Process and Runtime classes. Keep in mind that what you are trying to accomplish
is probably not platform independent.', 43, 12, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (28, 29, 28, '2020-03-26', 'BREAKING BAD! This
is the best TV show on earth. It builds up slowly but methodically to expand over
several seasons. I was amazed by the storytelling. Every piece slowly falls into
place to create a grand finale that has been mastered to blow your mind away.',
31, 1, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (29, 28, 29, '2019-11-06', 'Japans rising sun
flag. The Rising Sun Flag is the military flag of Japan. It was used as the ensign
of the Imperial Japanese Navy and the war flag of the Imperial Japanese Army until
the end of World War II. It is also presently the ensign of the Japan Maritime
Self-Defense Force and the war flag of the Japan Ground Self-Defense Force. It is
also waved during the Japanese New Year and in sporting events. The design is
similar to the flag of Japan in that it has a red circle close to the middle
signifying the sun, the difference being the addition of extra sun rays (16 for
the ensign) exemplifying the name of Japan as "The Land of the Rising Sun". The
flag was used in overseas actions from the Meiji period to World War II. When
Japan was defeated in August 1945, the flag was banned by Allied Occupation
authorities. However, with the re-establishment of a Self-Defense Force, the flag
was re-adopted in 1954. The flag with 16 rays is today the ensign of the Maritime
Self-Defense Force while the Ground Self-Defense Force uses an 8 ray version. This
flag is often considered offensive in countries which were victims of Japanese
aggression, particularly China and Korea, where it is considered as a symbol of
Japanese imperialism.', 55, 12, FALSE);
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (30, 30, 30, '2020-03-30', 'it would depend
upon your definition of better, is it better to have a low figure because you have
not tested people at the appropriate time or have a higher figure because you have
accurately tested many more people who may have the infection so naturally you
have a more accurate understanding of the scale of the issue?
perhaps you consider it better to look at the number of confirmed cases that
subsequently died as a result of the infection either directly or indirectly?
Canada has done the tests and can track those patients being treated for serious
reaction to the infection. The USA has not done the testing to the same level as
Canada so their results are patchy at best.
Canada has a health service that everyone can access without fear of bankruptcy,

```

lifelong debit or needing to sell their homes, so more people are willing to find out whether they are infected, in the USA due to their healthcare system many poorer people try not to go to the Doctor as they cannot afford it, so their deaths could well be misclassified as there is no test data or possibly post mortem to accurately identify the cause of death.

In either case I would say that Canada is beating the USA hands down.', 21, 3, FALSE);

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (31, 5, 1, '2019-12-20', 'Angelina Jolie.
The real life Gia Carangi struggled with drug addiction and isolation, making it
less-than-ideal to completely method your way through her portrayal. Angelina
Jolie settled on meeting the goal halfway, hoping that the role (which she felt
personally connected to) would "purge her demons" in the process.
To fully realize her character's loneliness, the actress cut off ties with her
friends and even refrained from speaking to her then-husband Jonny Lee Miller,
telling him "I'm alone; I'm dying; I'm gay; I'm not going to see you for weeks."
Pretty hardcore, but also suspicious since they divorced soon after this movie.',
54, 21, FALSE);
```

```
insert into 'answer' (id, user_id, question_id, answer_date, content, nr_likes,
nr_dislikes, marked_answer) values (32, 4, 8, '2020-03-27', 'Chernobyl
(miniseries).
```

After hearing a lot of rave reviews, I had high expectations. I just finished watching Chernobyl. Forget expectations. It blew my mind away. Everything about this series is superlative. It has brilliantly portrayed the meltdown of the Chernobyl nuclear power plant with deeply unsettling representation of the before and after events. The tragedy is woven in magical storytelling that makes this series utterly-compulsive TV drama.', 37, 6, FALSE);

--comment

```
insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (1, 3, 1, NULL, '2019-12-07', 'I believe it would also be interesting if
people commented on the worse method actors as well.');
```

```
insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (2, 4, 10, NULL, '2020-03-21', 'You should not compare the health strength
of two countries during these hard times! Dislike!');
```

```
insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (3, 1, NULL, 32, '2020-03-27', 'I agree! Awesome dialogue, great choice of
actors and awesome camera shots. The perfect drama for sure.');
```

```
insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (4, 6, 9, NULL, '2020-03-28', 'Well, that is a rather weird question, but in
a good way though.');
```

```
insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (5, 9, NULL, 2, '2020-03-28', 'Impeccable! What a good answer. Like!');
```

```
insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (6, 5, NULL, 9, '2019-12-03', 'That is a good answer! But you have made it
way to big. Next time, try to sum up the content more efficiently :));
```

```
insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (7, 13, 11, NULL, '2019-12-07', 'I believe it would also be interesting if
people commented on the worse method actors as well.');
```

```
insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (8, 14, 20, NULL, '2020-03-21', 'You should not compare the health strength
of two countries during these hard times! Dislike!');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (9, 16, 19, NULL, '2020-03-28', 'Well, that is a rather wurd question, but
in a good way though.');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (10, 19, NULL, 12, '2020-03-28', 'Impecable! What a good answer. Like!');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (11, 15, NULL, 19, '2019-12-03', 'That is a good answer! But you have made
it way to big. Next time, try to sum up the content more efficiently :)');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (12, 23, 21, NULL, '2019-12-07', 'I believe it would also be interesting if
people commented on the worse method actors as well.');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (13, 24, 30, NULL, '2020-03-21', 'You should not compare the health
strength of two countries during these hard times! Dislike!');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (14, 26, 29, NULL, '2020-03-28', 'Well, that is a rather wurd question, but
in a good way though.');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (15, 29, NULL, 22, '2020-03-28', 'Impecable! What a good answer. Like!');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (16, 25, NULL, 29, '2019-12-03', 'That is a good answer! But you have made
it way to big. Next time, try to sum up the content more efficiently :)');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (17, 33, 31, NULL, '2019-12-07', 'I believe it would also be interesting if
people commented on the worse method actors as well.');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (18, 34, 40, NULL, '2020-03-21', 'You should not compare the health
strength of two countries during these hard times! Dislike!');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (19, 36, 39, NULL, '2020-03-28', 'Well, that is a rather wurd question, but
in a good way though.');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (20, 39, NULL, 32, '2020-03-28', 'Impecable! What a good answer. Like!');
```

```

insert into 'comment' (id, user_id, question_id, answer_id, comment_date, content)
values (21, 35, NULL, 39, '2019-12-03', 'That is a good answer! But you have made
it way to big. Next time, try to sum up the content more efficiently :)');
```

--about

```

insert into 'about' (question_id, label_id) values (1, 5);
insert into 'about' (question_id, label_id) values (2, 2);
insert into 'about' (question_id, label_id) values (3, 3);
insert into 'about' (question_id, label_id) values (4, 30);
insert into 'about' (question_id, label_id) values (5, 2);
insert into 'about' (question_id, label_id) values (5, 4);
insert into 'about' (question_id, label_id) values (6, 21);
insert into 'about' (question_id, label_id) values (7, 7);
insert into 'about' (question_id, label_id) values (7, 23);
insert into 'about' (question_id, label_id) values (8, 9);
insert into 'about' (question_id, label_id) values (9, 17);
insert into 'about' (question_id, label_id) values (10, 20);
insert into 'about' (question_id, label_id) values (11, 5);
insert into 'about' (question_id, label_id) values (12, 2);
insert into 'about' (question_id, label_id) values (13, 3);
insert into 'about' (question_id, label_id) values (14, 30);
```

```
insert into 'about' (question_id, label_id) values (15, 2);
insert into 'about' (question_id, label_id) values (15, 4);
insert into 'about' (question_id, label_id) values (16, 21);
insert into 'about' (question_id, label_id) values (17, 7);
insert into 'about' (question_id, label_id) values (17, 23);
insert into 'about' (question_id, label_id) values (18, 9);
insert into 'about' (question_id, label_id) values (19, 17);
insert into 'about' (question_id, label_id) values (20, 20);
insert into 'about' (question_id, label_id) values (21, 5);
insert into 'about' (question_id, label_id) values (22, 2);
insert into 'about' (question_id, label_id) values (23, 3);
insert into 'about' (question_id, label_id) values (24, 30);
insert into 'about' (question_id, label_id) values (25, 2);
insert into 'about' (question_id, label_id) values (25, 4);
insert into 'about' (question_id, label_id) values (26, 21);
insert into 'about' (question_id, label_id) values (27, 7);
insert into 'about' (question_id, label_id) values (27, 23);
insert into 'about' (question_id, label_id) values (28, 9);
insert into 'about' (question_id, label_id) values (29, 17);
insert into 'about' (question_id, label_id) values (30, 20);

--notification
insert into 'notification' (id, content, date, viewed, user_id) values (1,
'cenas1', '2019-12-1', FALSE, 20);
insert into 'notification' (id, content, date, viewed, user_id) values (2,
'cenas2', '2019-12-2', FALSE, 19);
insert into 'notification' (id, content, date, viewed, user_id) values (3,
'cenas3', '2019-12-3', FALSE, 18);
insert into 'notification' (id, content, date, viewed, user_id) values (4,
'cenas4', '2019-12-4', FALSE, 17);
insert into 'notification' (id, content, date, viewed, user_id) values (5,
'cenas5', '2019-12-5', TRUE, 16);
insert into 'notification' (id, content, date, viewed, user_id) values (6,
'cenas6', '2019-12-6', TRUE, 15);
insert into 'notification' (id, content, date, viewed, user_id) values (7,
'cenas7', '2019-12-7', FALSE, 14);
insert into 'notification' (id, content, date, viewed, user_id) values (8,
'cenas8', '2019-12-8', TRUE, 13);
insert into 'notification' (id, content, date, viewed, user_id) values (9,
'cenas9', '2019-12-9', TRUE, 12);
insert into 'notification' (id, content, date, viewed, user_id) values (10,
'cenas10', '2019-12-10', FALSE, 11);
insert into 'notification' (id, content, date, viewed, user_id) values (11,
'cenas11', '2019-12-11', TRUE, 10);
insert into 'notification' (id, content, date, viewed, user_id) values (12,
'cenas12', '2019-12-12', TRUE, 9);
insert into 'notification' (id, content, date, viewed, user_id) values (13,
'cenas13', '2019-12-13', FALSE, 8);
insert into 'notification' (id, content, date, viewed, user_id) values (14,
'cenas14', '2019-12-1', TRUE, 7);
insert into 'notification' (id, content, date, viewed, user_id) values (15,
'cenas15', '2019-12-1', TRUE, 6);
insert into 'notification' (id, content, date, viewed, user_id) values (16,
'cenas16', '2019-12-1', FALSE, 5);
```

```
insert into 'notification' (id, content, date, viewed, user_id) values (17,
'cen17', '2019-12-1', TRUE, 4);
insert into 'notification' (id, content, date, viewed, user_id) values (18,
'cen18', '2019-12-1', TRUE, 3);
insert into 'notification' (id, content, date, viewed, user_id) values (19,
'cen19', '2019-12-1', FALSE, 2);
insert into 'notification' (id, content, date, viewed, user_id) values (20,
'cen20', '2019-12-1', TRUE, 1);

--vote
insert into 'vote' (id, value, user_id, question_id, answer_id) values (1, FALSE,
1, NULL, 1);
insert into 'vote' (id, value, user_id, question_id, answer_id) values (2, TRUE,
2, NULL, 1);
insert into 'vote' (id, value, user_id, question_id, answer_id) values (3, TRUE,
3, 1, NULL);
insert into 'vote' (id, value, user_id, question_id, answer_id) values (4, FALSE,
4, 2, NULL);
insert into 'vote' (id, value, user_id, question_id, answer_id) values (5, TRUE,
5, 2, NULL);
```

Revision history

1. First submission (30/03/2020).

GROUP2064, 29/03/2020

- Antonio Pedro Reis Ribeiro Sousa Dantas, up201703878@fe.up.pt
- Eduardo João Santana Macedo, up201703658@fe.up.pt
- Nuno Miguel Teixeira Cardoso, up201706162@fe.up.pt
- Paulo Roberto Dias Mourato, up201705616@fe.up.pt