University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: **2**

Date: **Oct. 19, 2023**

Group Number: **14**

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
David Guo	34332156	z1u5s	david-guo@live.ca
Kevin Poon	42821165	k2i1b	kevinp@live.ca
Kevin Zhou	35883164	d8u8l	zhoukev12@gmail.com

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

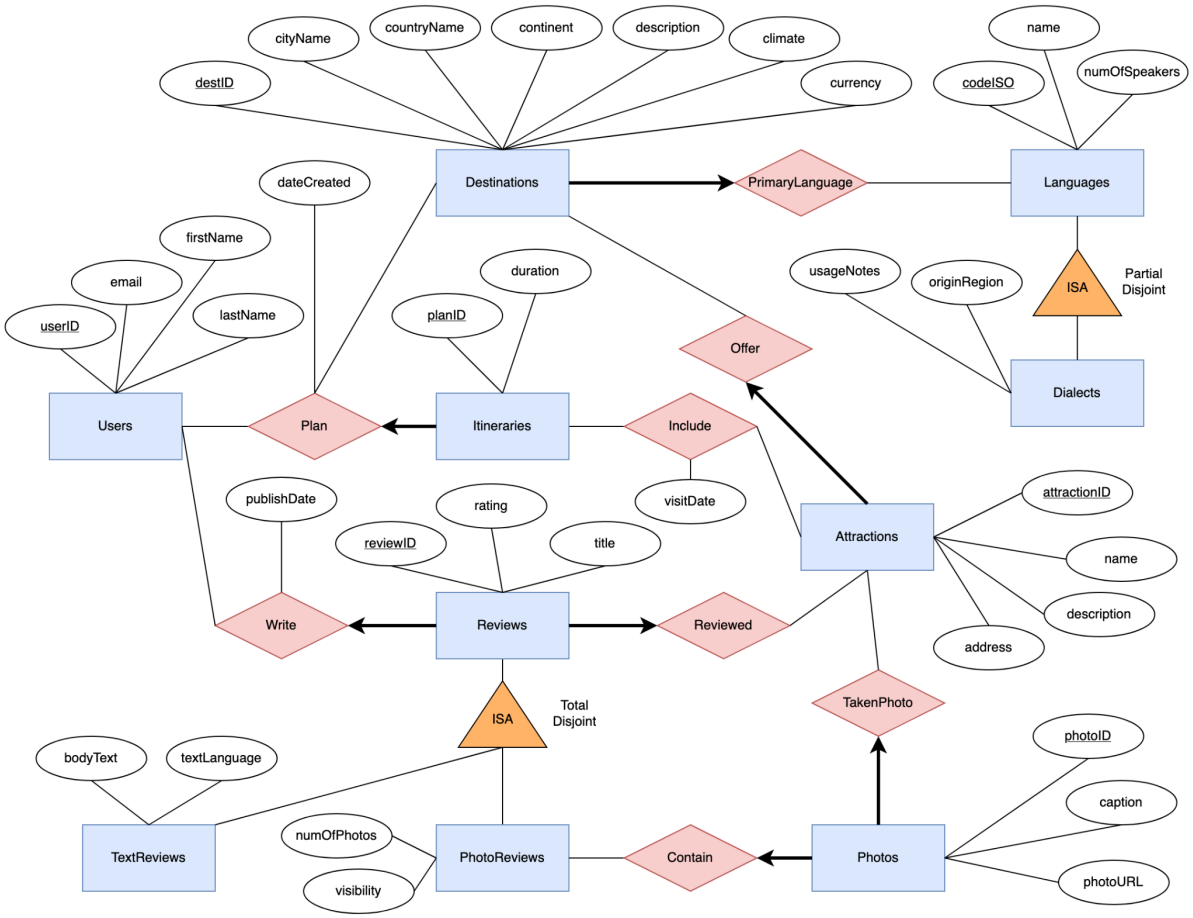
In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

CPSC 304: Milestone 2 (Group 14)
Tourism Planning Application

SUMMARY

Our project centers around a travel and tourism management application, aiming to serve as a cohesive platform that enriches travel planning for its users. Specifically, our application seeks to empower users with information and tools for crafting, sharing, and exploring travel itineraries, attractions, and user-generated content, including reviews and photos. Our database primarily models crucial elements for efficient travel planning and community engagement, such as *Users*, *Itineraries*, *Reviews*, *PhotoReviews*, *Destinations*, *Attractions*, Photos, *Languages*, and *Dialects*.

ER DIAGRAM



- Added appropriate constraints to ISA hierarchies
- Changed total participation many-to-one relationship from Destinations to Plan into many-to-many as each itinerary references one destination and one user
- Changed “Include” relationship to many-to-many since itinerary can include multiple attractions and an attraction can be part of multiple itineraries
- Many relationships have attributes now **(suggestion implemented)**
- Reviews has two subsets now: TextReviews and PhotoReviews **(suggestion implemented)**

RELATIONAL SCHEMA

Notes:

- Used method 1 to convert Languages ISA hierarchy into relational schema.
- Used method 3 to convert Reviews ISA hierarchy into relational schema.
- If a key or attribute is not mentioned in bullet points, there are no constraints on them.
- We decided not to merge the total disjoint ISA, because it will overcomplicate the table.

Users(*userID*: varchar(50), *email*: varchar(50), *firstName*: varchar(50), *lastName*: varchar(50))

- PK: userID
- CK: userID, email
- userID: unique, not null
- email: unique

Destinations(*destID*: integer, *cityName*: varchar(50), *countryName*: varchar(50), *continent*: varchar(50), *description*: varchar(300), *climate*: varchar(500), *currency*: char(3), *codeISO*: char(3))

- PK: destID
- CK: destID
- FK: codeISO
- destID: unique, not null
- codeISO: not null

Itineraries(*planID*: integer, *duration*: integer, *dateCreated*: date, *destID*: integer, *userID*: varchar(50))

- PK: planID
- CK: planID
- FK: userID, destID
- planID: unique, not null
- userID: not null
- destID: not null

Attractions(*attractionID*: integer, *name*: varchar(50), *description*: varchar(500), *address*: varchar(300), *destID*: integer)

- PK: attractionID
- CK: attractionID
- FK: destID
- attractionID: unique, not null
- destID: not null

Photos(*photoID*: integer, *caption*: varchar(300), *photoURL*: varchar(100), *attractionID*: integer, *reviewID*: integer)

- PK: photoID
- CK: photoID, photoURL
- FK: attractionID, reviewID
- photoID: unique, not null
- photoURL: unique
- reviewID: not null
- attractionID: not null

Languages(*codeISO*: char(3), *name*: varchar(50), *numOfSpeakers*: integer, *originRegion*: varchar(50), *usageNotes*: varchar(300))

- PK: codeISO
- CK: codeISO
- codeISO: unique, not null

TextReviews(*reviewID*: integer, *rating*: integer, *title*: varchar(50), *publishDate*: date *bodyText*: varchar(500), *textLanguage*: varchar(50), *userID*: varchar(50), *attractionID*: integer)

- PK: reviewID
- CK: reviewID
- FK:userID, attractionID
- reviewID: unique, not null
- userID: not null
- attractionID: not null

PhotoReviews(*reviewID*: integer, *rating:* integer*, title*: varchar(50), *publishDate*: date, *numOfPhotos*: integer, *visibility*: integer, *userID*: varchar(50), *attractionID*: integer)

- PK: reviewID
- CK: reviewID
- FK: userID, attractionID
- reviewID: unique, not null
- userID: not null
- attractionID: not null

Include(*attractionID*: integer, *planId*: integer, *visitDate*: date)

- PK: (attractionID, planID)
- CK: (attractionID, planID)
- FK: attractionID, planID
- attractionID: unique, not null
- planID: unique, not null

FUNCTIONAL DEPENDENCIES

Users(<u>userID</u>, email, firstName, lastName)
- PK: userID → email, firstName, lastName
- CK: email → userID, firstName, lastName

Destinations(<u>destID</u>, cityName, countryName, continent, description, climate, currency, codeISO)
- PK: destID → cityName, countryName, continent, description, climate, currency, codeISO
- countryName → continent, currency
- cityName, countryName → climate, codeISO

Itineraries(<u>planID</u>, duration, dateCreated, destID, userID)
- PK: planID → duration, dateCreated, destID, userID

Attractions(<u>attractionID</u>, name, description, address, destID)
- PK: attractionID → name, description, address, destID

TextReviews(<u>reviewID</u>, rating, title, publishDate, bodyText, textLanguage, attractionID, userID)
- PK: reviewID → rating, title, publishDate, bodyText, textLanguage, attractionID, userID

PhotoReviews(<u>reviewID</u>, rating, title, publishDate, numOfPhotos, visibility, attractionID, userID)
- PK: reviewID → rating, title, publishDate, numOfPhotos, visibility, attractionID, userID

Languages(<u>codeISO</u>, name, numOfSpeakers, usageNotes, originRegion)
- PK: codeISO → name, numOfSpeakers, usageNotes, originRegion

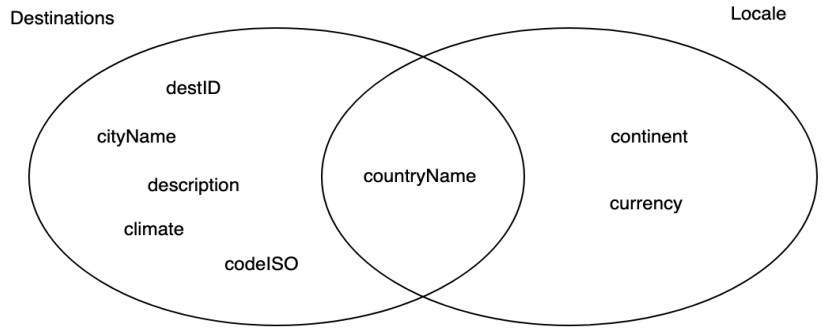
Photos(<u>photoID</u>, caption, photoURL, attractionID, reviewID)
- PK: photoID → caption, photoURL, attractionID, reviewID
- CK: photoURL → caption, photoURL, attractionID, reviewID

Include(<u>attractionID</u>, <u>planID</u>, visitDate)
- PK: attractionID, planID → visitDate

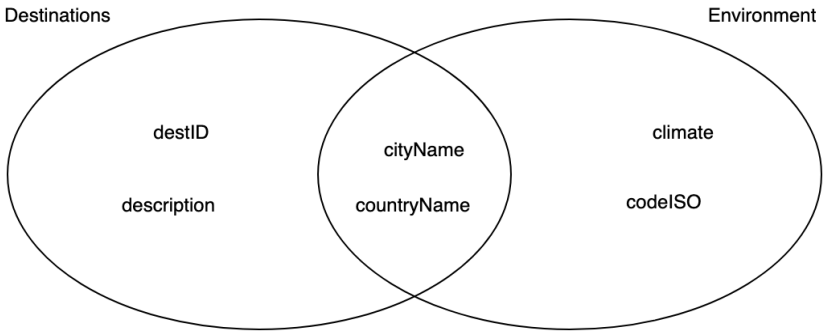
NORMALIZATION (BCNF)

Destinations(<u>destID</u>, cityName, countryName, continent, description, climate, currency, codeISO)
- PK: destID → cityName, countryName, continent, description, climate, currency, codeISO
- countryName → continent, currency
- cityName, countryName → climate, codeISO

This is not in BCNF because the second and third FD are non-trivial but are not superkeys for Destinations. Decomposing Destinations on FD: countryName → continent, currency



Now we decompose Destinations on FD: cityName, countryName → climate, codeISO



Destinations(<u>destID</u>, cityName, countryName, description)

- PK: destID → cityName, countryName, description
- FK: cityName, (cityName, countryName)

Locale(<u>countryName</u>, continent, currency)

- PK: countryName → continent, currency

Environment(<u>cityName</u>, <u>countryName</u>, climate, codeISO)

- PK: cityName, countryName → climate, codeISO
- FK: codeISO

Users(<u>userID</u>, email, firstName, lastName)

- PK: userID → email, firstName, lastName
- CK: email → userID, firstName, lastName

Itineraries(<u>planID</u>, duration, attractionID, dateCreated, destID, userID)

- PK: planID → duration, attractionID, dateCreated, destID, userID

Attractions(<u>attractionID</u>, name, description, address, destID)

- PK: attractionID → name, description, address, destID

TextReviews(reviewID, rating, title, publishDate, bodyText, textLanguage, attractionID, userID)

- PK: reviewID → rating, title, publishDate, bodyText, textLanguage, attractionID, userID

PhotoReviews(reviewID, rating, title, publishDate, numOfPhotos, visibility, attractionID, userID)

- PK: reviewID → rating, title, publishDate, numOfPhotos, visibility, attractionID, userID

Languages(codeISO, name, numOfSpeakers, usageNotes, originRegion)

- PK: codeISO → name, numOfSpeakers, usageNotes, originRegion

Photos(photoID, caption, photoURL, attractionID, reviewID)

- PK: photoID → caption, photoURL, attractionID, reviewID
- CK: photoURL → caption, photoURL, attractionID, reviewID

Include(attractionID, planID, visitDate)

- PK: attractionID, planID → visitDate

SQL DDL STATEMENTS

```
CREATE TABLE Users(
      userID      VARCHAR(50) PRIMARY KEY,
      email       VARCHAR(50) UNIQUE,
      firstName   VARCHAR(50),
      lastName    VARCHAR(50))

CREATE TABLE Locale(
      countryName       VARCHAR(50) PRIMARY KEY,
      continent         VARCHAR(50),
      currency          CHAR(3))

CREATE TABLE Languages(
      codeISO           CHAR(3) PRIMARY KEY,
      name              VARCHAR(50),
      numOfSpeakers     INTEGER,
      originRegion      VARCHAR(50),
      usageNotes        VARCHAR(300))

CREATE TABLE Environment(
      cityName      VARCHAR(50),
      countryName   VARCHAR(50),
      climate       VARCHAR(500),
      codeISO       CHAR(3) NOT NULL,
      PRIMARY KEY (cityName, countryName),
      FOREIGN KEY (codeISO) REFERENCES Languages)
```

```
CREATE TABLE Destinations(
      destID            INTEGER PRIMARY KEY,
      cityName          VARCHAR(50) NOT NULL,
      countryName       VARCHAR(50) NOT NULL,
      description       VARCHAR(300),
      FOREIGN KEY (countryName) REFERENCES Locale
                  ON DELETE CASCADE,
      FOREIGN KEY (cityName, countryName) REFERENCES Environment
                  ON DELETE CASCADE)

CREATE TABLE Attractions(
      attractionID    INTEGER PRIMARY KEY,
      name            VARCHAR(50),
      description     VARCHAR(500),
      address         VARCHAR(300),
      destID          INTEGER NOT NULL,
      FOREIGN KEY (destID) REFERENCES Destinations
            ON DELETE CASCADE)

CREATE TABLE Itineraries(
      planID            INTEGER PRIMARY KEY,
      duration          INTEGER,
      dateCreated       DATE,
      destID            INTEGER NOT NULL,
      userID            VARCHAR(50) NOT NULL,
      FOREIGN KEY (destID) REFERENCES Destinations
            ON DELETE CASCADE,
      FOREIGN KEY (userID) REFERENCES Users
            ON DELETE CASCADE)

CREATE TABLE TextReviews(
      reviewID          INTEGER PRIMARY KEY,
      rating            INTEGER,
      title             VARCHAR(50),
      publishDate       DATE,
      bodyText          VARCHAR(500),
      textLanguage      VARCHAR(50),
      userID            VARCHAR(50) NOT NULL,
      attractionID      INTEGER NOT NULL,
      FOREIGN KEY (userID) REFERENCES Users
            ON DELETE CASCADE,
      FOREIGN KEY (attractionID) REFERENCES Attractions
            ON DELETE CASCADE,
```

```
            CHECK (rating >= 0 AND rating <= 10))

CREATE TABLE PhotoReviews(
      reviewID            INTEGER PRIMARY KEY,
      rating              INTEGER,
      title               VARCHAR(50),
      publishDate         DATE,
      numOfPhotos         INTEGER,
      visibility          INTEGER,
      userID              VARCHAR(50) NOT NULL,
      attractionID        INTEGER NOT NULL,
      FOREIGN KEY (userID) REFERENCES Users
            ON DELETE CASCADE,
      FOREIGN KEY (attractionID) REFERENCES Attractions
            ON DELETE CASCADE,
      CHECK (rating >= 0 AND rating <= 10),
      CHECK (visibility IN (0, 1)))

CREATE TABLE Photos(
      photoID           INTEGER PRIMARY KEY,
      caption           VARCHAR(300),
      photoURL          VARCHAR(100) UNIQUE,
      attractionID      INTEGER NOT NULL,
      reviewID          INTEGER NOT NULL,
      FOREIGN KEY (attractionID) REFERENCES Attractions
            ON DELETE CASCADE,
      FOREIGN KEY (reviewID) REFERENCES PhotoReviews
            ON DELETE CASCADE)

CREATE TABLE Include(
      attractionID      INTEGER,
      planID            INTEGER,
      visitDate         DATE,
      PRIMARY KEY (attractionID, planID),
      FOREIGN KEY (attractionID) REFERENCES Attractions
            ON DELETE CASCADE,
      FOREIGN KEY (planID) REFERENCES Itineraries
            ON DELETE CASCADE)
```

Note: 'ON UPDATE CASCADE' is not supported by Oracle so we will be using triggers for later milestones

INSERT STATEMENTS

```
-- Users table

INSERT INTO Users(userID, email, firstName, lastName)
VALUES ('dagu', 'dagu@email.com', 'David', 'Guo');

INSERT INTO Users(userID, email, firstName, lastName)
VALUES ('kev_mochi', 'kevz@email.com', 'Kevin', 'Zhou');

INSERT INTO Users(userID, email, firstName, lastName)
VALUES ('kev_megu', 'kp@email.com', 'Kevin', 'Poon');

INSERT INTO Users(userID, email, firstName, lastName)
VALUES ('squishyPancake', 'pancake@otheremail.com', 'Sophia', 'Martin');

INSERT INTO Users(userID, email, firstName, lastName)
VALUES ('trustme', 'trusttrust123@random.com', 'Olivia', 'Martin');

-- Destinations table

INSERT INTO Destinations(destID, cityName, countryName, description)
VALUES(1, 'Kyoto', 'Japan', 'It is a historic city with lots of traditional
architecture and temples.');

INSERT INTO Destinations(destID, cityName, countryName, description)
VALUES(2, 'Tokyo', 'Japan', 'Capital of Japan. It is a vibrant metropolis
with busy city life.');

INSERT INTO Destinations(destID, cityName, countryName, description)
VALUES(3, 'Osaka', 'Japan', 'It is a big city in western part of Japan.');

INSERT INTO Destinations(destID, cityName, countryName, description)
VALUES(4, 'Vancouver', 'Canada', 'A city in the province British Columbia.
Known for its integration of nature and city life.');

INSERT INTO Destinations(destID, cityName, countryName, description)
VALUES(5, 'Orlando', 'United States of America', 'Known for Disney World,
Universal Studios and other attractions.');
```

```
-- Locale table

INSERT INTO Locale(countryName, continent, currency)
VALUES('Canada', 'North America', 'CAD');

INSERT INTO Locale(countryName, continent, currency)
VALUES('Japan', 'Asia', 'JPY');

INSERT INTO Locale(countryName, continent, currency)
VALUES('China', 'Asia', 'CNY');

INSERT INTO Locale(countryName, continent, currency)
VALUES('United States of America', 'North America', 'USD');

INSERT INTO Locale(countryName, continent, currency)
VALUES('United Kingdom', 'Europe', 'GBP');


-- Environment table

INSERT INTO Environment(cityName, countryName, climate, codeISO)
VALUES('Vancouver', 'Canada', 'Sunny in the summer with temperature in the
range of 20 to 30 degrees Celsius. Rainy during the rest of the year.',
'eng');

INSERT INTO Environment(cityName, countryName, climate, codeISO)
VALUES('Tokyo', 'Japan', 'Very hot and humid during the summer.', 'jpn');

INSERT INTO Environment(cityName, countryName, climate, codeISO)
VALUES('Orlando', 'United States of America', 'Tropical weather.', 'eng');

INSERT INTO Environment(cityName, countryName, climate, codeISO)
VALUES('Osaka', 'Japan', 'Hot and humid in the summer. Mild weather for the
rest of the year.', 'jpk');

INSERT INTO Environment(cityName, countryName, climate, codeISO)
VALUES('Kyoto', 'Japan', 'Similar to Osaka. Hot and humid in the summer and
mild weather for the rest of the year.', 'jpk');
```

```
-- Itineraries table

INSERT INTO Itineraries(planID, duration, dateCreated, destID, userID)
VALUES(1, 4, '2022-12-01', 3, 'kev_mochi');

INSERT INTO Itineraries(planID, duration, dateCreated, destID, userID)
VALUES(2, 3, '2023-01-25', 2, 'kev_mochi');

INSERT INTO Itineraries(planID, duration, dateCreated, destID, userID)
VALUES(3, 7, '2023-10-10', 5, 'kev_megu');

INSERT INTO Itineraries(planID, duration, dateCreated, destID, userID)
VALUES(4, 3, '2023-03-07', 3, 'kev_megu');

INSERT INTO Itineraries(planID, duration, dateCreated, destID, userID)
VALUES(5, 7, '2022-09-01', 2, 'dagu');

-- Attractions table

INSERT INTO Attractions(attractionID, name, description, address, destID)
VALUES(1, 'Kiyomizu-dera', 'A famous temple in Kyoto.', '1 Chome-294
Kiyomizu, Higashiyama Ward, Kyoto, 605-0862, Japan', 1);

INSERT INTO Attractions(attractionID, name, description, address, destID)
VALUES(2, 'Tokyo Skytree', 'A famous tower in Tokyo', '1-chome-1-2 Oshiage,
Sumida City, Tokyo 131-0045, Japan', 2);

INSERT INTO Attractions(attractionID, name, description, address, destID)
VALUES(3, 'Stanley Park', 'A big park in the downtown Vancouver area.', '1042
Stanley Park Dr, Vancouver, BC V6G 3E2', 4);

INSERT INTO Attractions(attractionID, name, description, address, destID)
VALUES(4, 'Universal Studios Osaka', 'A great attraction in Osaka', '2
Chome-1-33 Sakurajima, Konohana Ward, Osaka, 554-0031, Japan', 3);

INSERT INTO Attractions(attractionID, name, description, address, destID)
VALUES(5, 'Imperial Palace', 'This is where the Japanese Royal family
lives.', '1-1 Chiyoda, Chiyoda City, Tokyo 100-8111, Japan', 2);
```

```
-- Photos table

INSERT INTO Photos(photoID, caption, photoURL, attractionID, reviewID)
VALUES(1, 'Beautiful Day at Stanley Park with Emily and Johnny',
'https://photowebsite.com/library/id=94af34tfaiunwrrj', 3, 5);

INSERT INTO Photos(photoID, caption, photoURL, attractionID, reviewID)
VALUES(2, 'At the entrance of Tokyo Skytree',
'https://photowebsite.com/library/id=gaarewuauhr847abfrgrgr', 2, 2);

INSERT INTO Photos(photoID, caption, photoURL, attractionID, reviewID)
VALUES(3, 'Inside Universal Studios Osaka',
'https://photowebsite.com/library/id=4u587hfy8r3ur', 4, 1);

INSERT INTO Photos(photoID, caption, photoURL, attractionID, reviewID)
VALUES(4, 'At Kiyomizu-dera',
'https://photowebsite.com/library/id=nfaih30434', 1, 3);

INSERT INTO Photos(photoID, caption, photoURL, attractionID, reviewID)
VALUES(5, 'Rainy Stanley Park',
'https://photowebsite.com/library/id=sdfaw97hruh3', 3, 4);

-- Languages table

INSERT INTO Languages(codeISO, name, numOfSpeakers, originRegion, usageNotes)
VALUES('eng', 'English', 604300000, NULL, NULL);

INSERT INTO Languages(codeISO, name, numOfSpeakers, originRegion, usageNotes)
VALUES('jpn', 'Japanese', 125100000, NULL, NULL);

INSERT INTO Languages(codeISO, name, numOfSpeakers, originRegion, usageNotes)
VALUES('kor', 'Korean', 75000000, NULL, NULL);

INSERT INTO Languages(codeISO, name, numOfSpeakers, originRegion, usageNotes)
VALUES('chi', 'Chinese', 1300000000, NULL, NULL);

INSERT INTO Languages(codeISO, name, numOfSpeakers, originRegion, usageNotes)
VALUES('jpk', 'Kansai Japanese', 125100000, 'Kansai region of Japan',
'Kansai-ben is primarily used in western part of Japan such as Osaka and
Kyoto.');
```

```
-- TextReviews table

INSERT INTO TextReviews(reviewID, rating, title, publishDate, bodyText,
textLanguage, userID, attractionID)
VALUES(1, 8, 'Visiting Tokyo Skytree', '2023-09-01', 'とっても素敵でした！',
'Japanese', 'kev_mochi', 2);

INSERT INTO TextReviews(reviewID, rating, title, publishDate, bodyText,
textLanguage, userID, attractionID)
VALUES(2, 10, 'A Day at Universal Studios Osaka', '2023-09-30', 'Many
different things you can do there! Must visit!', 'English', 'kev_megu', 4);

INSERT INTO TextReviews(reviewID, rating, title, publishDate, bodyText,
textLanguage, userID, attractionID)
VALUES(3, 9, 'Stanley Park on Canada Day', '2023-07-01', 'There were a lot of
people! Go early if you can!', 'English', 'trustme', 3);

INSERT INTO TextReviews(reviewID, rating, title, publishDate, bodyText,
textLanguage, userID, attractionID)
VALUES(4, 3, 'Stanley Park Day Trip', '2019-10-04', 'The weather was bad. Not
a fun day.', 'English', 'squishyPancake', 3);

INSERT INTO TextReviews(reviewID, rating, title, publishDate, bodyText,
textLanguage, userID, attractionID)
VALUES(5, 7, 'Kiyomizu-dera trip', '2023-05-31', 'This is a famous attraction
and so expect there to be a lot of people.', 'English', 'dagu', 1);

-- PhotoReviews table
-- visibility == 1 is public, visibility == 0 is private

INSERT INTO PhotoReviews(reviewID, rating, title, publishDate, numOfPhotos,
visibility, userID, attractionID)
VALUES(1, 8, 'Day Trip to Universal Studios Osaka', '2023-09-01', 1, 1,
'kev_megu', 4);

INSERT INTO PhotoReviews(reviewID, rating, title, publishDate, numOfPhotos,
visibility, userID, attractionID)
VALUES(2, 10, 'Visiting Tokyo Skytree', '2023-08-31', 1, 1, 'kev_mochi', 2);

INSERT INTO PhotoReviews(reviewID, rating, title, publishDate, numOfPhotos,
visibility, userID, attractionID)
VALUES(3, 10, 'A day at Kiyomizu-dera', '2023-05-10', 1, 1, 'dagu', 1);
```

```
INSERT INTO PhotoReviews(reviewID, rating, title, publishDate, numOfPhotos,
visibility, userID, attractionID)
VALUES(4, 6, 'Stanley Park Day Trip', '2019-10-03', 1, 0, 'squishyPancake',
3);

INSERT INTO PhotoReviews(reviewID, rating, title, publishDate, numOfPhotos,
visibility, userID, attractionID)
VALUES(5, 9, 'Stanley Park with Friends', '2018-07-15', 1, 1, 'trustme', 3);


-- Include table

INSERT INTO Include(attractionID, planID, visitDate)
VALUES(2, 2, '2023-08-27');

INSERT INTO Include(attractionID, planID, visitDate)
VALUES(4, 4, '2023-07-01');

INSERT INTO Include(attractionID, planID, visitDate)
VALUES(1, 5, '2023-05-03');

INSERT INTO Include(attractionID, planID, visitDate)
VALUES(4, 1, '2023-08-25');

INSERT INTO Include(attractionID, planID, visitDate)
VALUES(5, 2, '2023-08-28');
```