Department of Computer Science

CPSC 304 Project Cover Page

| Milestone #:4 |
|--------------------|
| Date:April 5, 2024 |
| Group Number: 24 |

| Name | Student Number | CS Alias (Userid) | Preferred E-mail Address |
|----------------|-------------------|----------------------|------------------------------|
| Katherine Poon | 90607425 | o1w1j | katherinepoon12345@gmail.com |
| Maya Arafa | 75726604 | k9i0h | arafa.maya@gmail.com |
| Inaki Blasco | 93863819 | g5m5d | inaki.blasco1@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

Department of Computer Science

Short Description of Final Project (and what it accomplished)

Our final project provides a platform for the culinary industry, particularly gourmet dining and restaurant management in Vancouver. Our projection allows users to write, edit and delete reviews, view others' reviews, and find restaurants based on various criteria (group by, aggregation, etc.) through our user-friendly GUI.

Description of How Final Schema Differed From Original Schema

The final schema differs slightly from our original one, as we added an additional many-to-one relationship between SignatureDish and Cuisine, to ensure we meet the minimum requirement of having 7 relationships. Furthermore, in our DDL statements, we have combined some to reduce the redundancy - we got rid of ReviewerHasReview and RestaurantHasReview tables and combined those relationships into the Review table, as well as getting rid of the RestaurantServes table and combining that relationship with the Restaurant table.

Schema

Note: Primary keys are underlined, foreign keys are bolded, and candidate keys are italicized.

Cuisine(CuisineID, Name, Description)

Location(Address, PostalCode, City, Province)

Reviewer(ReviewerID, Name)

LeadChef(ChefID, Name, Biography)

Restaurant(<u>RestaurantID</u>, Name, Website, PriceRange, **Address**, **PostalCode**, AverageScore, **CuisineID**)

WorksAtRestaurant(*ChefID*, *RestaurantID*)

SignatureDish(<u>DishName</u>, <u>RestaurantID</u>, Description, Course, <u>CuisineID</u>)

Award(AwardID, Name, MichelinRating, Description, RestaurantID)

ProfessionalCritic(ReviewerID, Title)

FoodBlogger(**ReviewerID**, Website)

Review(ReviewID, RestaurantID, ReviewerID, Date, Score, Comment)

Department of Computer Science

Screenshots of Data in Each Relation After SQL Script is Run

Cuisine:

```
SQL> select * from cuisine;

CUISINEID NAME DESCRIPTION

1 Chinese Cantonese
2 Contemporary French
3 Japanese Sushi

3 rows selected.
```

Location:

| SQL> select * from location; | | | |
|--|--|---|----------------------------------|
| ADDRESS | POSTALCODE | CITY | PROVINCE |
| 3330 Main St. 1038 Mainland St. 750 Nelson St. 898 Homer St. 845 Hornby St. 108-777 W. Broadway | V5V 3M7 V6B 2T4 V6Z 1A8 V6B 2W5 V6Z 1V1 V5Z 4J7 | Vancouver Vancouver Vancouver Vancouver Vancouver | BC BC BC BC BC BC |

Reviewer

```
SQL> select * from reviewer;
no rows selected
```

LeadChef

```
SQL> select * from leadchef;

CHEFID NAME

1 Hilary Nguy
2 Takuya Matsuda
3 Chef Jin
4 Bobby Milheron
5 Montgomery Lau
6 Sam Leung
CHEFID NAME

BIOGRAPHY

Vietnamese Chef Hilary Nguy grew up in East Van, opening Sushi Hil in 2022.
From Mie Prefecture, Japan, Chef Matsuda started his career at the age of 15, accumulating more than 25 years of experience.
Chef Jin has over 30 years of experience serving extraordinary food to extraordinary guests.
Chef Bobby Milheron is the Executive Chef of Wentworth Hospitality Group and competed at the Canadian Culinary Championship in 2023.
Executive Chef Sam Leung began his apprenticeship in Guangzhou and immigrated to Canada in 1985.
```

Department of Computer Science

Restaurant

| SQL> select * | * from restaurant | t; | | | | | |
|------------------|---|--|------------|---------|--|---------------------------------|-----------|
| RESTAURANTID | NAME | WEBSITE | PRICERANGE | ADDRESS | POSTALCODE | AVERAGESCORE | CUISINEID |
| 2 3 4 5 | Okeya Kyujiro Sushi Jin Homer St. Cafe Bacchus | https://www.sushihil.com/ https://okeya.ca/ https://sushijinvancouver.ca/ https://www.homerstreetcafebar.com/ https://wedgwoodhotel.com/dining/bacchus-restaurant/ https://dynasty-restaurant.ca/ | | | V5V 3M7 V6B 2T4 V6Z 1A8 V6B 2W5 V6Z 1V1 V5Z 4J7 | 4.7 4.8 4.6 4.4 4.5 | |
| 6 rows select | ted. | | | | | | |

WorksAtRestaurant

| SQL> select * from V | WorksAtRestaurant; |
|----------------------|--------------------|
| CHEFID RESTAURA | NTID |
| | |
| 1 | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 6 | 6 |
| 6 rows selected. | |

SignatureDish

| SQL> select * from s | signaturedish; | | |
|--|--|-------------|----------------------------|
| DISHNAME | RESTAURANTID DESCRIPTION | COURSE | CUISINEID |
| Aka Isaki Omakase Course Omakase Rotisserie Chicken Tomato Soup Dim Sum | 1 Aka Isaki, or red grunt, is served with orange to underscore the 2 Highlights include chawanmushi with cherry blossom, shatteringly crispy 3 Begin with otsumami such as live lobster sashimi, followed by delicious courses 4 Brined in Homer St. Cafes signature sauce overnight to render tender, flavourful meat 5 Humble tomato soup is elevated. 6 A wide selection from shrimp dumplings to congee and soups. | Main Course | 3 3 3 2 2 1 |
| 6 rows selected. | | | |

Award

```
SQL> select * from award;

AWARDID NAME MICHELINRATING DESCRIPTION RESTAURANTID

1 Bib Gourmand good quality, good value cooking
1 High quality cooking
2 Good cooking
3 Selected Good cooking
4 Selected Good cooking
5 Selected Good cooking
6 Selected Good cooking
6 Selected
6 Fows selected.
```

Foodblogger, ProfessionalCritic, Review

```
SQL> select * from foodBlogger;

no rows selected

SQL> select * from professionalCritic;

no rows selected

SQL> select * from Review;

no rows selected
```

Department of Computer Science

A list of all SQL queries used and where it can be found in the code (i.e., file name and line number(s))

Insert Operation:

```
INSERT INTO Review
VALUES
  (
    NULL,
    : bind1,
    : bind2,
    TO_DATE(: bind3, 'YYYY-MM-DD'),
    : bind4,
    : bind5
)
```

File: main.php Line: 458

Delete Operation:

DELETE FROM ProfessionalCritic
WHERE ReviewerID = :reviewerID

DELETE FROM FoodBlogger
WHERE ReviewerID = :reviewerID

DELETE FROM Review
WHERE ReviewerID = :reviewerID

DELETE FROM Reviewer
WHERE ReviewerID = :reviewerID

File: main.php

Line: 617, 618, 619, 620

Update Operation:

File: main.php Line: 416

Department of Computer Science

Selection:

```
SELECT *
FROM Review
WHERE ReviewerID = :reviewerID
ORDER BY \"Date\"
```

File: main.php Line: 573

Projection:

SELECT \$attributesListString FROM Restaurant

File: main.php Line: 597

Join:

```
SELECT DISTINCT Name, DishName, PriceRange
FROM Restaurant r
INNER JOIN
SignatureDish s
ON r.RestaurantID = s.RestaurantID
WHERE PriceRange = :price
```

File: restaurant.php

Line: 390

Aggregation with Group By:

SELECT
CuisineID,
MAX(AverageScore)
FROM
Restaurant
GROUP BY
CuisineID

File: restaurant.php

Line: 311

Department of Computer Science

Aggregation with Having:

```
SELECT A.Name, A.MichelinRating, COUNT(R.RestaurantID)
FROM Award A
INNER JOIN
Restaurant R
ON A.RestaurantID = R.RestaurantID
GROUP BY A.Name, A.MichelinRating
HAVING COUNT(R.RestaurantID) >= :minimum"
```

File: restaurant.php

Line: 342

Nested Aggregation with Group By:

File: restaurant.php

Line: 361

Division:

```
SELECT Reviewer.ReviewerID, Reviewer.Name
FROM Reviewer
JOIN Review ON Reviewer.ReviewerID = Review.ReviewerID
GROUP BY Reviewer.ReviewerID, Reviewer.Name
HAVING COUNT(DISTINCT Review.RestaurantID) = :totalRestaurants
```

File: main.php

Line: 656,657,658,659,660

Department of Computer Science

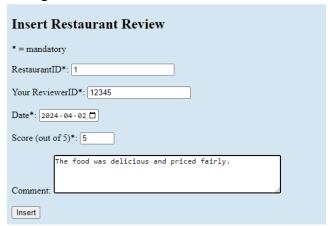
Screenshots demonstrating the functionality of each query using the GUI (before/during/after progression of events)

Insert Operation:

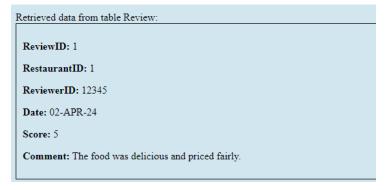
Before:

| urant Review |
|--|
| |
| |
| |
| |
| |
| |
| |
| |
| ata from table Review: * (no reviews present |
| ata from table Review: * (no reviews preser |

During:



After:



Department of Computer Science

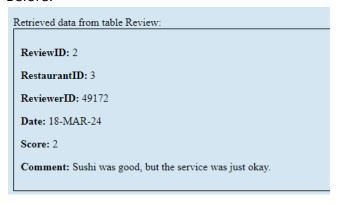
Delete Operation:

Before:

| Delete a Reviewer Account |
|---|
| All reviews by this account will also be deleted |
| Reviewer ID: 2 |
| Delete |
| During: |
| www.students.cs.ubc.ca says |
| Are you sure you want to delete this account? |
| Cancel OK |
| After: |
| Reviewer and all related data successfully deleted. |

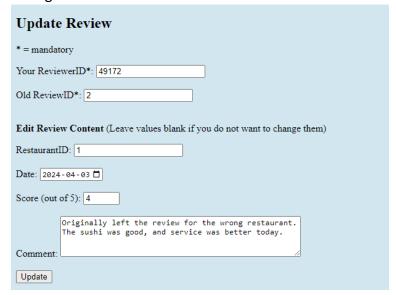
Update Operation:

Before:

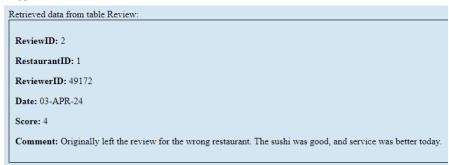


Department of Computer Science

During:



After:



Selection:

During:



Department of Computer Science

| After: | |
|---|--|
| Retrieved data from table Review: | |
| ReviewID: 1 RestaurantID: 1 | |
| ReviewerID: 1 | |
| | |
| Date: 05-APR-24 | |
| Score: 5 | |
| Comment: Amazing! | |
| | |
| Projection: | |
| Before: | |
| Display Restaurants with Selected Attributes | |
| Select attributes to display: | |
| Name (always displayed) Website Price Range Address Average Score Display | |
| During: | |
| Display Restaurants with Selected Attributes | |
| Select attributes to display: | |
| Name (always displayed) Website Price Range Address Average Score Display | |
| After: | |
| Retrieved Restaurant Data: | |
| Name | |
| Sushi Hil | |
| Okeya Kyujiro | |
| Sushi Jin Homer St. Cafe | |
| Homer St. Cafe | |

Bacchus

Dynasty Seafood

Department of Computer Science

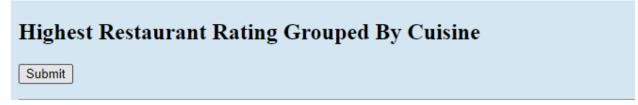
<u>Join:</u>

Before:

| Display Sign | nature Dishes | s by Price | Range |
|--|---|---------------|----------|
| Select Price Range | : \$ V Submit | | |
| Resetting Database | ; | | |
| During: | | | |
| Display Sig | \$\$ | es by Pri | ce Range |
| Select File Rang | \$\$\$\$ Submit | J | |
| Resetting Databas | e | | |
| | | | |
| After: | | | |
| | nature Dishe | s by Pric | e Range |
| | | s by Pric | e Range |
| Display Sig | | | |
| Display Sig | e: \$ V Submit | | |
| Display Sig Select Price Range Retrieved data fro | e: \$ Submit | gnatureDish T | |
| Display Sig Select Price Range Retrieved data fro Restaurant | m Restaurant and Si Signature Dish Rotisserie Chicken | gnatureDish T | |

Aggregation with Group By:

Before:



During:



Department of Computer Science

After:

| Retrieved data from Restaurant table: CuisineID Max Rating |
|--|
| |
| 1 10 |
| 1 4.2 |
| 2 4.5 |
| 3 4.8 |
| |
| Cuisine Table for Reference: |
| CuisineID Name Description |
| 1 Chinese Cantonese |
| 2 Contemporary French |
| 3 Japanese Sushi |

Aggregation with Having:

Before:

| Award Categories with Minimum Number of Restaurants |
|---|
| Enter minimum number of restaurants: |
| Submit |

During:

| Award Categories with Minimum Number of Restaurants | | | | | |
|--|--|--|--|--|--|
| Enter minimum number of restaurants: 1 | | | | | |
| Submit | | | | | |

After:

Retrieved data from Restaurant and Award tables:

| Award Name | Michelin Stars | # of Restaurants |
|---------------|----------------|------------------|
| Selected | | 4 |
| Bib Gourmand | | 1 |
| Michelin Star | 1 | 1 |

Department of Computer Science

Nested Aggregation with Group By:

Before:

Restaurant with Highest Rating By Price Range

Submit

After:

Retrieved data from Restaurant table:

| Restaurant | Price Range | Rating |
|---------------|--------------------|--------|
| Sushi Hil | \$\$ | 4.7 |
| Bacchus | \$\$\$ | 4.5 |
| Okeya Kyujiro | \$\$\$\$ | 4.8 |

Division:

Before:

 $\ \, \textbf{Display All Reviewers that have reviewed all restaurants} \\$

Submit

After:

Reviewers who have reviewed all restaurants:

ReviewerID: 1

Name: abc