RecipeDelight Database Project

ITCS 3160 - 001

Fall 2019

James Cowell
Chandler Godfrey
William Misenheimer
Austin Moody

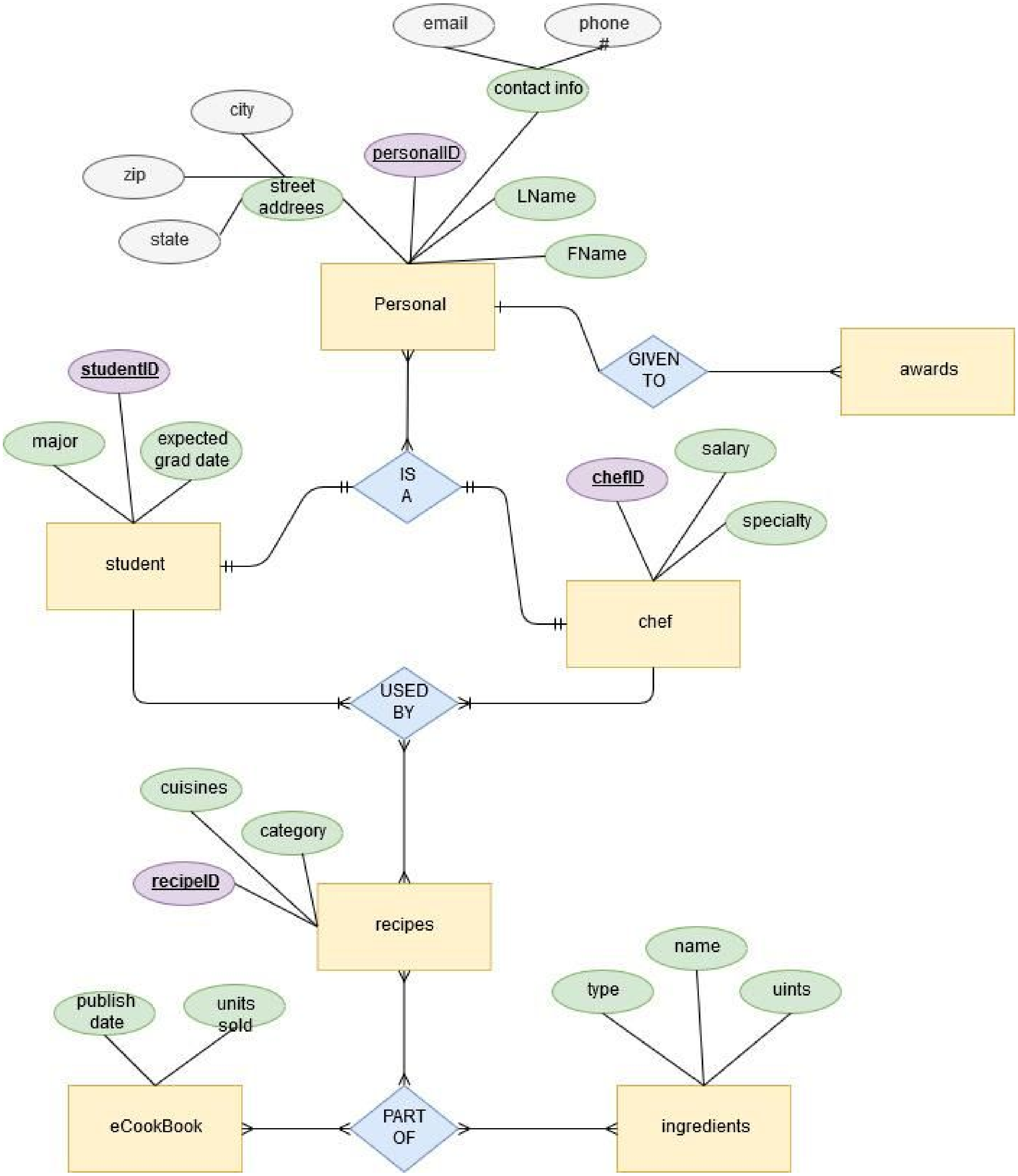
Commander in Chief Keef

October 14, 2019

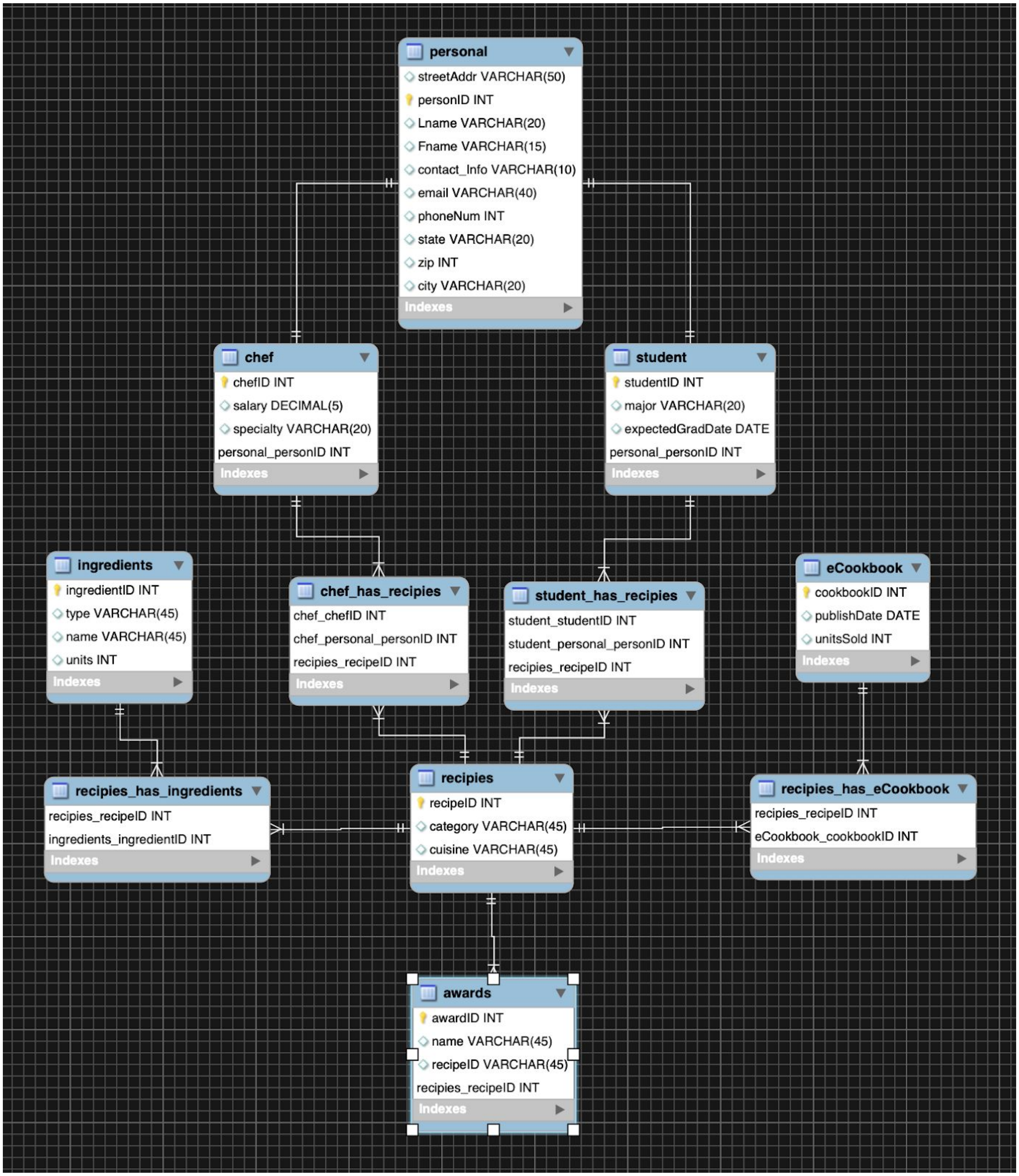
Table of Contents

1.1 Diagrams

1.1.1 EERD



1.1.2 Relational Model



1.2 Business Rules and Assumptions.

1.2.1 Personal

1. Person – Chef/Student is a generalization-specialization relationship (superclass/subclass)
2. A chef may be simultaneously assigned to multiple recipe projects. A project has a chef that is designated the Lead Chef. Egos being what they are, there may be multiple Lead Chefs
3. A student may be simultaneously assigned to multiple recipe projects.
4. A recipe is designated as available for download or for inclusion in an eCookBook when it is published as indicated by its publication date.
5. A chef or a student may be associated with multiple recipe projects.
6. A chef may be the only person assigned to a recipe project. (There cannot be a student assigned to a recipe project if there is no chef assigned.)
7. A student may not be the only person assigned to a recipe project.

1.2.3 Recipes, Ingredients and Quantities

1. Many recipes contain some of the same ingredient(s), and it is advisable for standardization that ensures references to an ingredient name are the same across all recipes.
2. An ingredient may be listed only once in a given recipe.
3. You may choose to express ingredient quantities in metric or English units.
4. Quantities should be expressed precisely. For instance, ½ stored as a float field might be rendered as 0.4999999 instead of 0.5. Therefore, quantity fields should probably not be the float data type.
5. Quantities should be usable in arithmetic computations. For instance, if the Number Served field of a recipe needs to be doubled, you may assume that all ingredient Quantities may be multiplied by two.
6. Awards are associated with recipes.
7. A recipe can win an award only once. Awards may be differentiated by time. A recipe could win both the Best Entre 2019 and Best Entre 2020 awards but could not win the Best Entre 2019 more than once.

8. A recipe can be included in a given eCookBook only once.
9. A recipe can be included in multiple eCookBooks.

1.2.4 Reports

1. Chef reports
   1.1. Listing unpublished recipe projects by Lead Chief.
   1.2. Listing published recipe projects by Lead Chief.
   1.3. Listing all recipe projects and include all assigned chefs.
2. Student reports
   2.1. Listing all students and their published recipe projects with assigned chefs.
3. Management report
   3.1. Listing all chefs including names and salaries.
   3.2. Listing of all students with names, a count of assigned, unpublished recipe projects and expected graduation date.

1.2.5. Recipes and eCookBooks

1. List each recipe with Recipe Name, Lead Chef, and a number of times it has been downloaded.
2. List each Recipe Name with its Ingredients. Include the Quantity and Units.
3. List each Recipe Name with its Instructions.
4. List each eCookBook with eCookBook Name and number of recipes in it.

1.2.6 Awards

1. List all recipes that have received awards including the name of the award.
2. List the person the award was given too.
   2.1.1. Chef
   2.1.2. Student

1.3 Team Member Project Assignment Roles.

James Cowell - Developed the database with help from Will. Austin assisted with debugging the database. Created the EERD model based off the ERD model provided by Austin, with help from Chandler and Will. Aided in formulating and completing the final business rules. Currently working on future aspects of the project.

Chandler Godfrey - Worked Austin to create the Relational Model. Helped develop the EERD made sure everything was in its right place. Aided in formulating and completing the final business rules. Currently working on future aspects of the project.

William Misenheimer - Assisted James with getting the database up and running. Aided in formulating and completing the final business rules. Currently working on future aspects of the project.

Austin Moody - Developed the Relational Model with help from Chandler. Also helped James with debugging the database. Created the draft ERD and prepared it for conversion into an EERD. Currently working on future aspects of the project.

1.4 Sample Database Code

Below is a sample of the SQL code used to CREATE the tables and columns for the personal, chef, and student portion of this project. As not to take up too much space we have limited the code from those three tables to just a few entries. Also included is a sample size of three entries from the INSERT code used to populate the aforementioned tables. A picture of each one of those tables populated is below the code snippets.

1.4.1 CREATE Code

```
drop database if exists `recipedelight`;
create database if not exists `recipedelight`;
use `recipedelight`;
drop table if exists `personal`;
CREATE TABLE `personal` (
   `personalID` VARCHAR(13) NOT NULL,
   `Lname` VARCHAR(255) DEFAULT NULL,
   `Fname` VARCHAR(255) DEFAULT NULL,
   `streetAddr` VARCHAR(255) DEFAULT NULL,
   `city` VARCHAR(255) DEFAULT NULL,
   `state` VARCHAR(50) DEFAULT NULL,
   `zip` VARCHAR(10) DEFAULT NULL,
   `email` VARCHAR(255) DEFAULT NULL,
   `phoneNum` VARCHAR(100) DEFAULT NULL,
   `chefID` VARCHAR(13) DEFAULT NULL,
   `studentID` VARCHAR(13) DEFAULT NULL,
   `awardID` VARCHAR(13) DEFAULT NULL,
   PRIMARY KEY (`personalID`)
)  ENGINE=InnoDB;

drop table if exists `chef`;
CREATE TABLE `chef` (
   `chefID` VARCHAR(13) NOT NULL,
   `salary` VARCHAR(100) DEFAULT NULL,
   `specialty` TEXT DEFAULT NULL,
   `personalID` VARCHAR(13) DEFAULT NULL,
   PRIMARY KEY (`chefID`),
   FOREIGN KEY (`personalID`)
      REFERENCES `personal` (`personalID`)
      ON DELETE NO ACTION ON UPDATE NO ACTION
)  ENGINE=InnoDB;
```

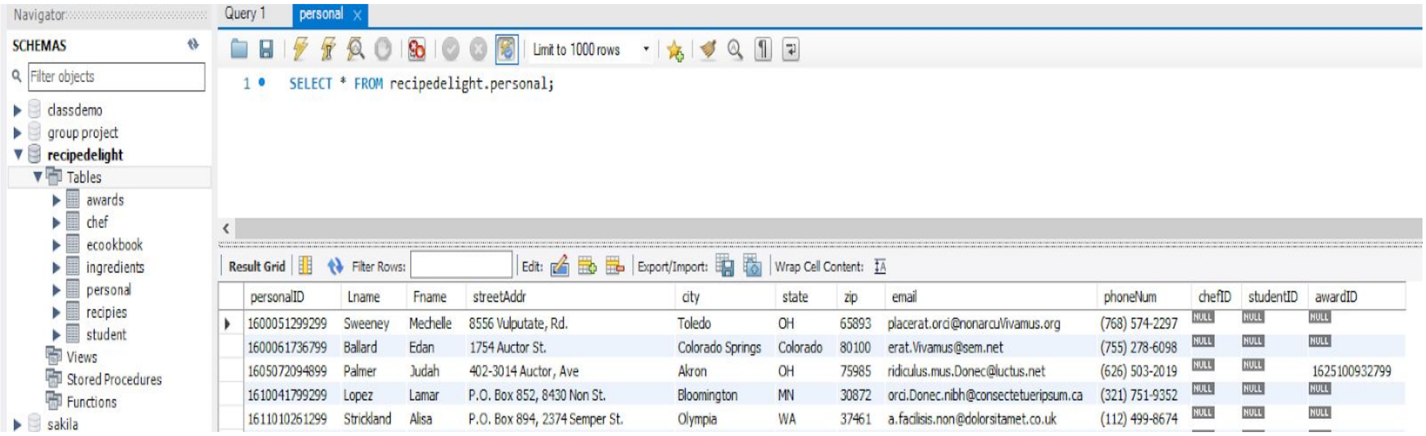
```
ALTER TABLE `personal`
ADD FOREIGN KEY (`chefID`)
                REFERENCES `chef`(`chefID`)
                ON DELETE NO ACTION
                ON UPDATE NO ACTION;

drop table if exists `student`;
CREATE TABLE `student` (
   `studentID` VARCHAR(13) NOT NULL,
   `major` VARCHAR(250) DEFAULT NULL,
   `expectedGradDate` VARCHAR(255) DEFAULT NULL,
   `personalID` VARCHAR(13) DEFAULT NULL,
   PRIMARY KEY (`studentID`),
        FOREIGN KEY (`personalID`)
      REFERENCES `personal` (`personalID`)
      ON DELETE NO ACTION ON UPDATE NO ACTION
)  ENGINE=InnoDB;

ALTER TABLE `personal`
ADD FOREIGN KEY (`studentID`)
                REFERENCES `student`(`studentID`)
                ON DELETE NO ACTION
                ON UPDATE NO ACTION;
```

1.4.2 INSERT Code

1.4.2.1 personal INSERT

```
INSERT INTO `personal`
(`personalID`,`Lname`,`Fname`,`streetAddr`,`city`,`state`,`zip`,`email`,`phoneNum`, `chefID`,
`studentID`, `awardID`)
VALUES ("1669022400199","Holt","Jack","P.O. Box 208, 3838 Fringilla
Ave","Boise","ID","99180","erat.in@scelerisqueduiSuspendisse.co.uk","(661) 343-3163", default,
default, "1683123020499");
INSERT INTO `personal`
(`personalID`,`Lname`,`Fname`,`streetAddr`,`city`,`state`,`zip`,`email`,`phoneNum`, `chefID`,
`studentID`, `awardID`)
VALUES ("1627081333799","Bryan","Cailin","P.O. Box 962, 2689 Pellentesque
Rd.","Pittsburgh","PA","74522","elit.pretium@primisin.edu","(472) 491-6624", default, default,
default);
```

INSERT INTO `personal` (`personalID`,`Lname`,`Fname`,`streetAddr`,`city`,`state`,`zip`,`email`,`phoneNum`, `chefID`, `studentID`, `awardID`)
VALUES ("1664010784399","Reyes","Bruno","1510 At Av.","Glendale","AZ","86695","non.luctus@dolorquamelementum.com","(638) 409-1196", default, default, default);

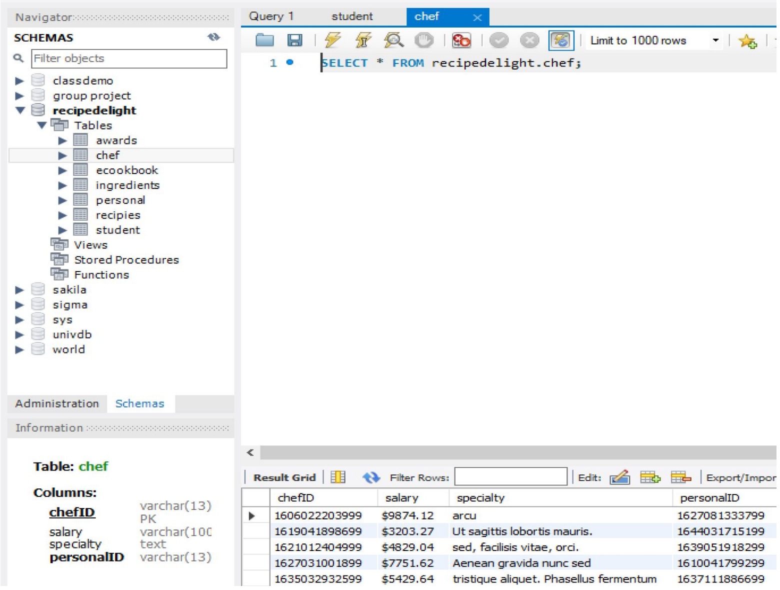


```
1 ● SELECT * FROM recipedelight.personal;
```

personalID	Lname	Fname	streetAddr	city	state	zip	email	phoneNum	chefID	studentID	awardID
1600051299299	Sweeney	Mechelle	8556 Vulputate, Rd.	Toledo	OH	65893	placerat.orci@nonarcuVivamus.org	(768) 574-2297	NULL	NULL	NULL
1600061736799	Ballard	Edan	1754 Auctor St.	Colorado Springs	Colorado	80100	erat.Vivamus@sem.net	(755) 278-6098	NULL	NULL	NULL
1605072094899	Palmer	Judah	402-3014 Auctor, Ave	Akron	OH	75985	ridiculus.mus.Donec@luctus.net	(626) 503-2019	NULL	NULL	1625100932799
1610041799299	Lopez	Lamar	P.O. Box 852, 8430 Non St.	Bloomington	MN	30872	orci.Donec.nibh@consectetueripsum.ca	(321) 751-9352	NULL	NULL	NULL
1611010261299	Strickland	Alisa	P.O. Box 894, 2374 Semper St.	Olympia	WA	37461	a.facilisis.non@dolorsitamet.co.uk	(112) 499-8674	NULL	NULL	NULL

1.4.2.2 chef INSERT

INSERT INTO `Chef` (`chefID`,`salary`,`specialty`, `personalID`)
VALUES ("1637011296799","$9820.77","dui. Fusce diam", "1669022400199");
INSERT INTO `Chef` (`chefID`,`salary`,`specialty`, `personalID`)
VALUES ("1606022203999","$9874.12","arcu", "1627081333799");
INSERT INTO `Chef` (`chefID`,`salary`,`specialty`, `personalID`)
VALUES ("1660062693699","$5462.96","molestie tellus.", "1664010784399");

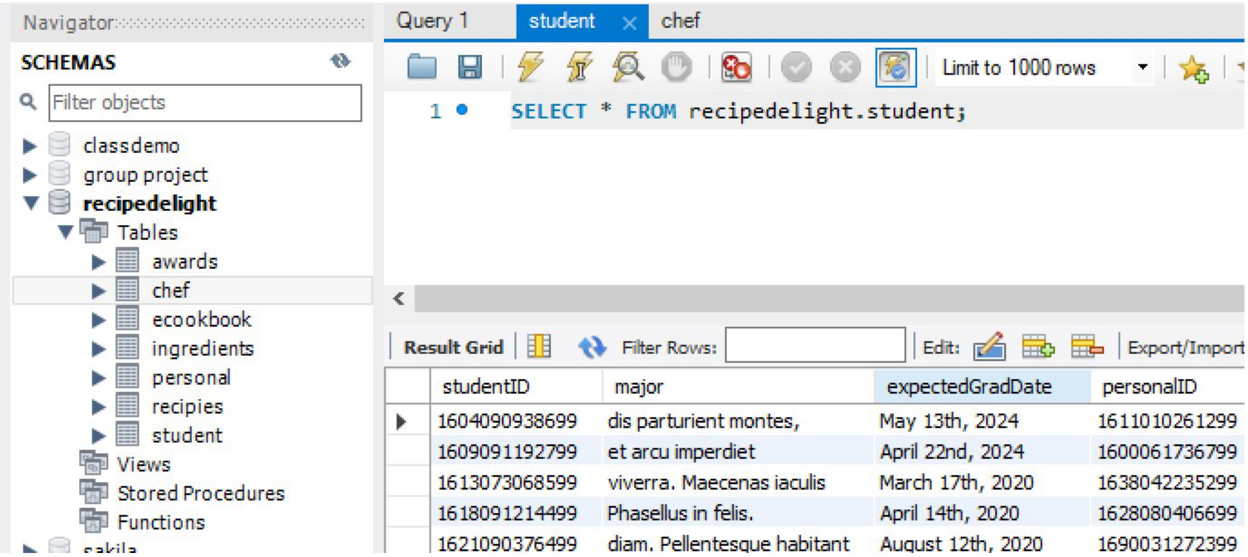


```
1 ● SELECT * FROM recipedelight.chef;
```

chefID	salary	specialty	personalID
1606022203999	$9874.12	arcu	1627081333799
1619041898699	$3203.27	Ut sagittis lobortis mauris.	1644031715199
1621012404999	$4829.04	sed, facilisis vitae, orci.	1639051918299
1627031001899	$7751.62	Aenean gravida nunc sed	1610041799299
1635032932599	$5429.64	tristique aliquet. Phasellus fermentum	1637111886699

1.4.2.3 student INSERT

```
INSERT INTO `student` (`studentID`,`major`,`expectedGradDate`, `personalID`)
VALUES ("1628052884899","convallis in, cursus","February 22nd, 2024", "1662091997599");
INSERT INTO `student` (`studentID`,`major`,`expectedGradDate`, `personalID`)
VALUES ("1621090376499","diam. Pellentesque habitant","August 12th, 2020",
"1690031272399");
INSERT INTO `student` (`studentID`,`major`,`expectedGradDate`, `personalID`)
VALUES ("1674021709799","Morbi neque tellus,","January 24th, 2021", "1638082583899");
```



```
1  SELECT * FROM recipedelight.student;
```

studentID	major	expectedGradDate	personalID
1604090938699	dis parturient montes,	May 13th, 2024	1611010261299
1609091192799	et arcu imperdiet	April 22nd, 2024	1600061736799
1613073068599	viverra. Maecenas iaculis	March 17th, 2020	1638042235299
1618091214499	Phasellus in felis.	April 14th, 2020	1628080406699
1621090376499	diam. Pellentesque habitant	August 12th, 2020	1690031272399

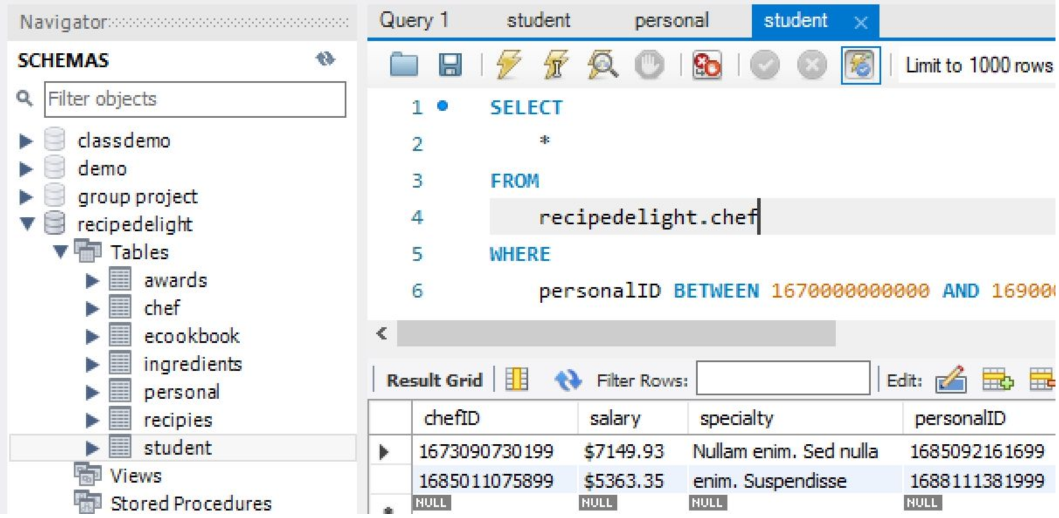
1.4.2.4 Chef ID numeric range QUERY

*SELECT * FROM*

recipedelight.chef

WHERE

personalID BETWEEN 1670000000000 AND 1690000000000;



```
SELECT
    *
FROM
    recipedelight.chef
WHERE
    personalID BETWEEN 1670000000000 AND 16900
```

chefID	salary	specialty	personalID
1673090730199	$7149.93	Nullam enim. Sed nulla	1685092161699
1685011075899	$5363.35	enim. Suspendisse	1688111381999
NULL	NULL	NULL	NULL

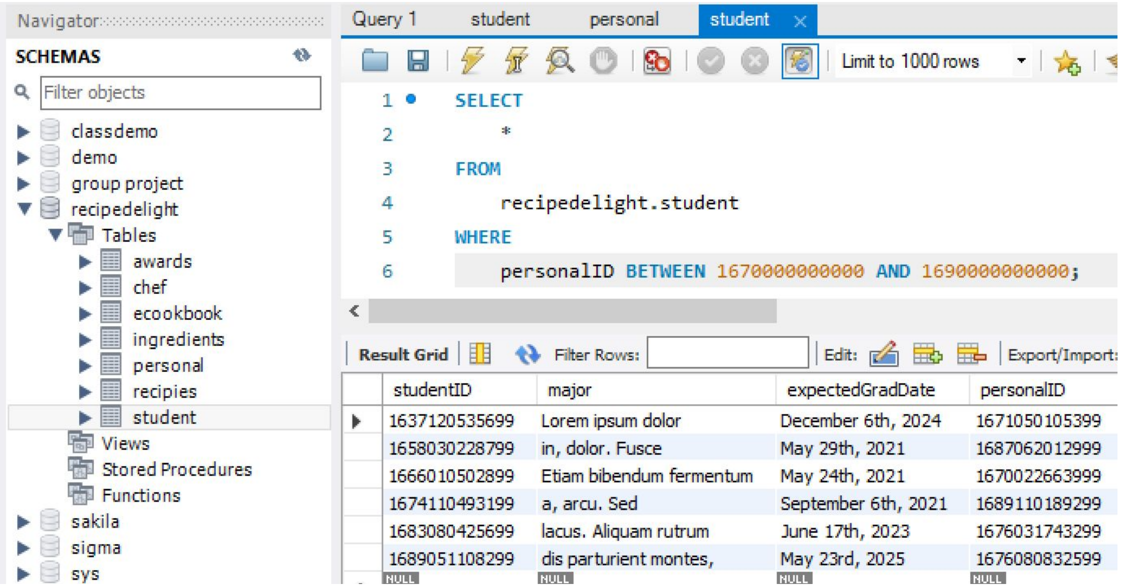
1.4.2.5 Student ID numeric range QUERY

*SELECT * FROM*

recipedelight.student

WHERE

personalID BETWEEN 1670000000000 AND 1690000000000;



```
SELECT
    *
FROM
    recipedelight.student
WHERE
    personalID BETWEEN 1670000000000 AND 1690000000000;
```

studentID	major	expectedGradDate	personalID
1637120535699	Lorem ipsum dolor	December 6th, 2024	1671050105399
1658030228799	in, dolor. Fusce	May 29th, 2021	1687062012999
1666010502899	Etiam bibendum fermentum	May 24th, 2021	1670022663999
1674110493199	a, arcu. Sed	September 6th, 2021	1689110189299
1683080425699	lacus. Aliquam rutrum	June 17th, 2023	1676031743299
1689051108299	dis parturient montes,	May 23rd, 2025	1676080832599
NULL	NULL	NULL	NULL