SE Comp - B	Roll number :
Experiment no. : 8	Date of Implementation :
Aim : To implement simple PLSql using Mysql stored procedures	
Tool Used : Mysql / PostgreSQL	
Related Course outcome : At the end of the course, Students will be able to Use SQL : Standard language of relational database	

Rubrics for assessment of Experiment:

Indicator	Poor	Average	Good
Timeliness • Maintains assignment deadline (3)	Assignment not done (0)	One or More than One week late (1-2)	Maintains deadline (3)
Completeness and neatness • Complete all parts of assignment(3)	N/A	< 80% complete (1-2)	100% complete (3)
Originality • Extent of plagiarism(2)	Copied it from someone else(0)	At least few questions have been done without copying(1)	Assignment has been solved completely without copying (2)
Knowledge • In depth knowledge of the assignment(2)	Unable to answer 2 questions(0)	Unable to answer 1 question (1)	Able to answer 2 questions (2)

Assessment Marks :

Timeliness	
Completeness and neatness	
Originality	
Knowledge	
Total	

Total : (Out of 10)

Teacher's Sign :

EXPERIMENT 8	Procedural sql
Aim	To implement PLSQL stored procedures
Tools	Link for Mysql: https://www.javatpoint.com/mysql-stored-function
Procedure	

	PL/SQL is a combination of SQL along with the procedural features of programming languages. Basic Syntax of PL/SQL which is a block-structured language; this means that the PL/SQL programs are divided and written in logical blocks of code. Each block consists of three sub-parts. Every PL/SQL statement ends with a semicolon (;). Following is the basic structure of a PL/SQL block –
	
	DECLARE
	<Declaration statements>
	BEGIN
	<Executable commands>
	EXCEPTION
	<Exception Handling>
	END;
	Simple example of a PL/SQL block:
	Declare
	Msg varcha2(20) := 'Helo world'
	Begin
	Dbms.output.put_line(msg);
	End;
	
	In Oracle, this simple block without name works. Such blocks are called anonymous blocks, but in Mysql , we cannot have anonymous name. In Mysql this can be implemented as named block or stored procedures.
	Anonymous blocks are PL/SQL blocks which do not have any names assigned to them.
	They need to be created and used in the same session because they will not be stored in the server as a database objects.
	Named blocks are having a specific and unique name for them.
	They are stored as the database objects in the server.
	Since they are available as database objects, they can be referred to or used as long as it is present in the server.

Example : Unnamed blocks

```
declare
num number:=1;
begin
for num in 1..10 loop
    dbms_output.put_line(num);
end loop;
end;
```

Named block/stored procedure:

CREATE PROCEDURE sp_name ([proc_parameter: [IN | OUT | INOUT] param_name data_type])

Begin

< declare variable_name data_type>

<control statements if else/loop>

SQL executable statements; End

End

Where,

procedure_name:

The name to assign to this procedure in MySQL.

Parameter:

Optional. One or more parameters passed into the procedure. When creating a procedure, there are three types of parameters that can be declared:

1. IN - The parameter can be referenced by the procedure. The value of the parameter can not be overwritten by the procedure.
2. OUT - The parameter can not be referenced by the procedure, but the value of the parameter can be overwritten by the procedure.

3. IN OUT - The parameter can be referenced by the procedure and the value of the parameter can be overwritten by the procedure.

declaration_section

The place in the procedure where you declare local variables.

executable_section

The place in the procedure where you enter the code for the procedure.

Example: Create a procedure for adding two numbers

```
DELIMITER //
CREATE procedure sumtwo ( IN a int, IN b int, OUT c INT )
BEGIN
    set c=a+b;
END //
DELIMITER ;
```

Note: here Delimiter could be any character like // or && or $$

Calling the procedure sumtwo:

```
call sumtwo(10,20,@var);
select @var;
```

Example: If..else.. Create a simple procedure that takes age as an input and gives status as 'senior citizen' or 'Not senior citizen' as output

```
DELIMITER //
CREATE PROCEDURE senior_citizen(IN age int, OUT status varcharacter(20))
BEGIN
        IF age >= 60 THEN
                set status = 'Senior citizen';
```

```
        ELSE
                set status = 'Not a senior citizen';
         END IF;
END //
DELIMITER ;

call senior_citizen(65,@status);
select @status;
```

Example: If..else..if ladder

```
DELIMITER //
CREATE PROCEDURE cal_grades(IN marks int, OUT grade varchar(10))
BEGIN
        IF marks >= 75 THEN
                set grade = 'DISTICTION';
        elseif marks>= 60 then
      set grade = 'First class';
   elseif marks >= 50 then
      set grade = 'second class';
        elseif marks>=40 then
      set grade = 'pass class';
   else
      set  grade = 'fail';
end if;
end //

call cal_grades(45,@grade);
select @grade;
```

Example: while Loop

Create a procedure to calculate income

```
DELIMITER //
CREATE procedure CalcIncome ( OUT ending_value INT )
```

	BEGIN
	DECLARE income INT;
	SET income = 50;
	label1: WHILE income <= 3000 DO
	SET income = income * 2;
	END WHILE label1;
	SET ending_value = income;
	END; //
	DELIMITER ;
	You could then reference your new procedure as follows:
	Call CalcIncome(@varname)
	Selecr @varname;
	
	References
	1) https://www.techonthenet.com/mysql/procedures.php
	2) https://www.mysqltutorial.org/mysql-stored-procedure-tutorial.aspx

Procedure

Task 1: Write PL/Sql block for the following

1. Write a stored procedure to accept salary of the employee and display grade of employee accordingly.
   If salary > 50000 then grade is 'A'
   If salary between 30000 to 50000 the grade is 'B' and
   If salary < 30000 then grade is 'c'
2. Write a block to display sum of 1 to 10 numbers
3. Write a block to display Fibonacci series upto 8th term (start with 0,1)

Task 2:

Create following tables:
Borrower(roll_no, name, DateofIssue, book_name, Status)
Fine(roll_no, Date , Amt)

Insert few values into Borrower table

Borrower Table

Roll_no	Name	DateofIssue	NameofBook	Status
1	Amita	2017-06-25	Java	I
2	Sonakshi	2017-07-10	Networking	I
3	Nira	2017-05-22	MySQL	I
4	Jagdish	2017-06-10	DBMS	I
5	Jayashree	2017-07-05	MySQL	I
6	Kiran	2017-06-30	Java	I

Fine Table

Roll_no	Date	Amt

Write a procedure that Accepts roll_no & name of book from user.
Check the number of days (from date of issue)

1) if days are between 15 to 30 then fine amount will be Rs 5 per day.
2) If no. of days > 30, per day fine will be Rs 50 per day &
3) for days less than 30, Rs. 5 per day.

Whenever student returns the book, update both the tables

1) Make status of that book in Borrower table as 'R' ('R'; for return)
2) If there is a fine , then new row should be added to the fine table.

Table after procedure Run

Borrower Table

Rno	Name	DateofIssue	NameofBook	Status
1	Amita	2017-06-25	Java	I
2	Sonakshi	2017-07-10	Networking	I
3	Nira	2017-05-22	MySQL	I
4	Jagdish	2017-06-10	DBMS	R
5	Jayashree	2017-07-05	MySQL	I
6	Kiran	2017-06-30	Java	I

Fine Table

Roll_no	Date	Amt
4	2017-06-30	100

Post Lab Questions:

1. Give advantages of PLSQL vs SQL
2. Explain data types of PLSQL in Mysql

Name: Brendan Lucas, Div: SE COMP B, Roll No: 8953

DBMS Practical Implementation, Lab 8.

Task 1

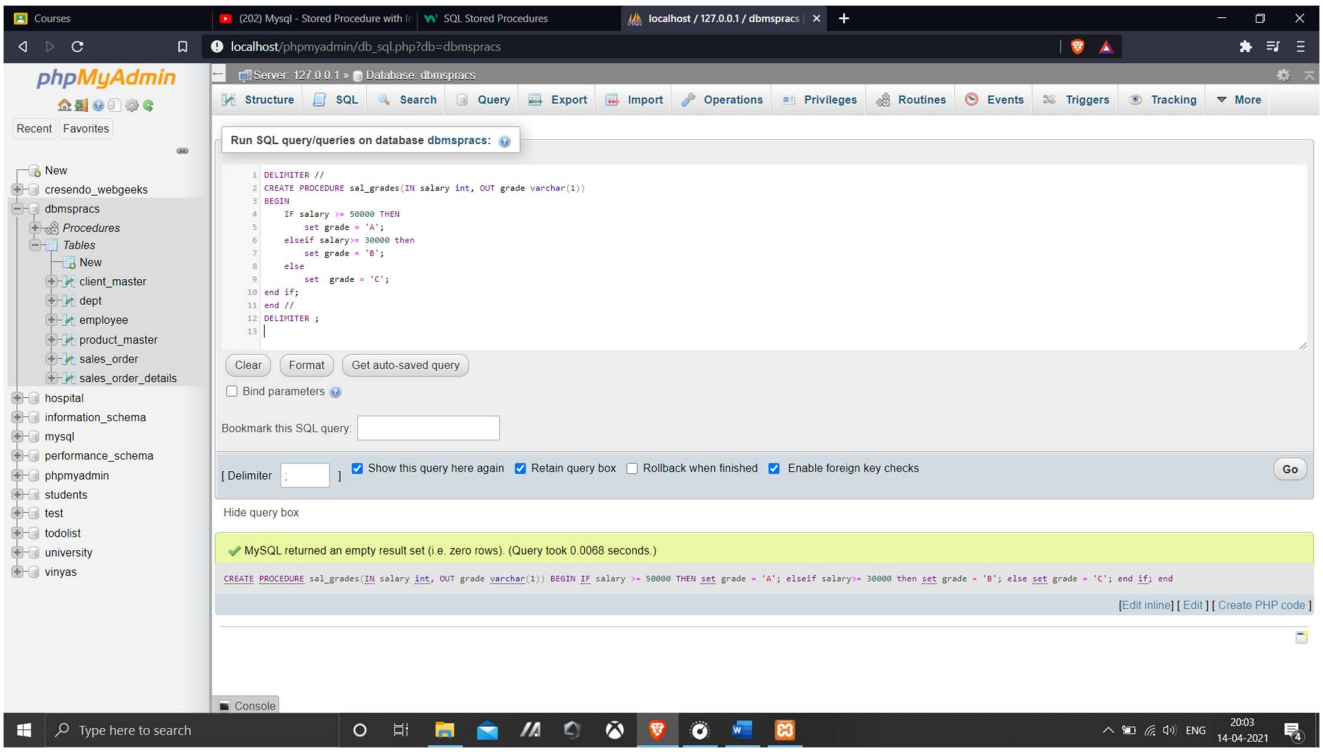
1.Write a stored procedure to accept salary of the employee and display grade of employee accordingly.

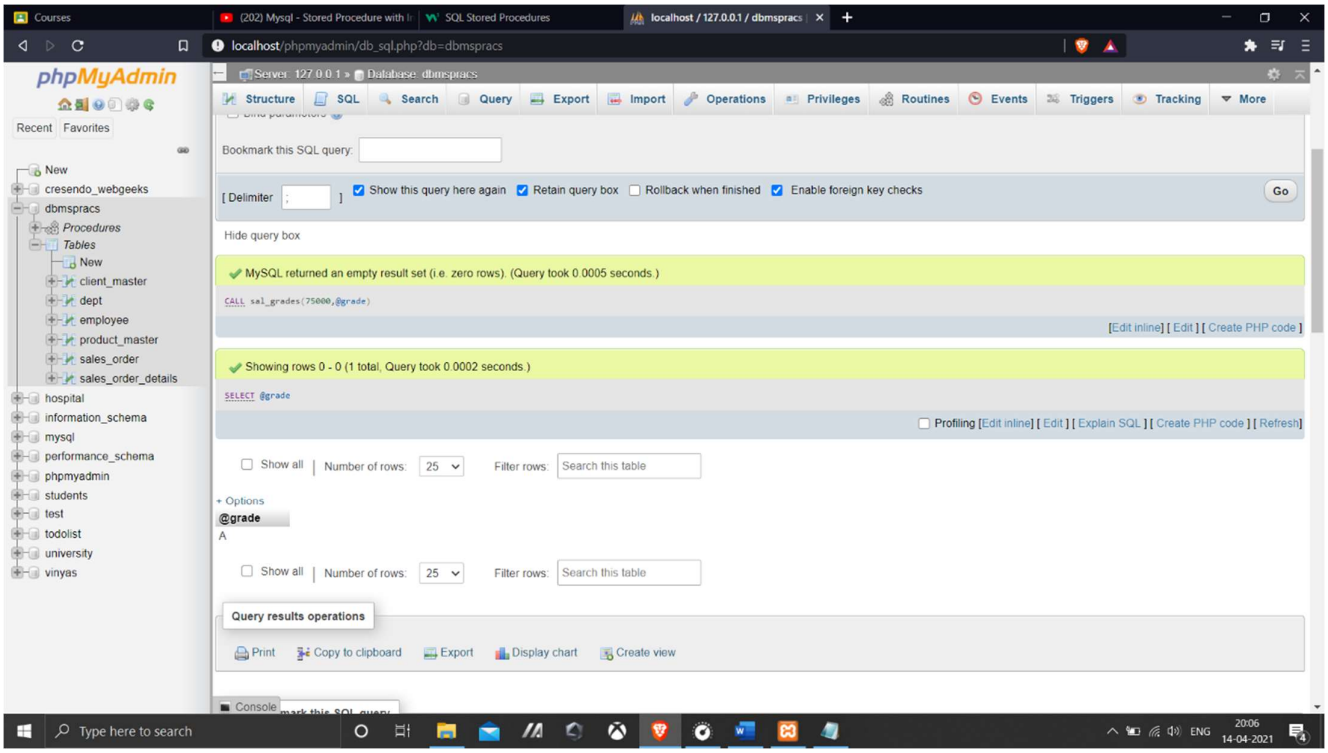
If salary > 50000 then grade is ‘A’

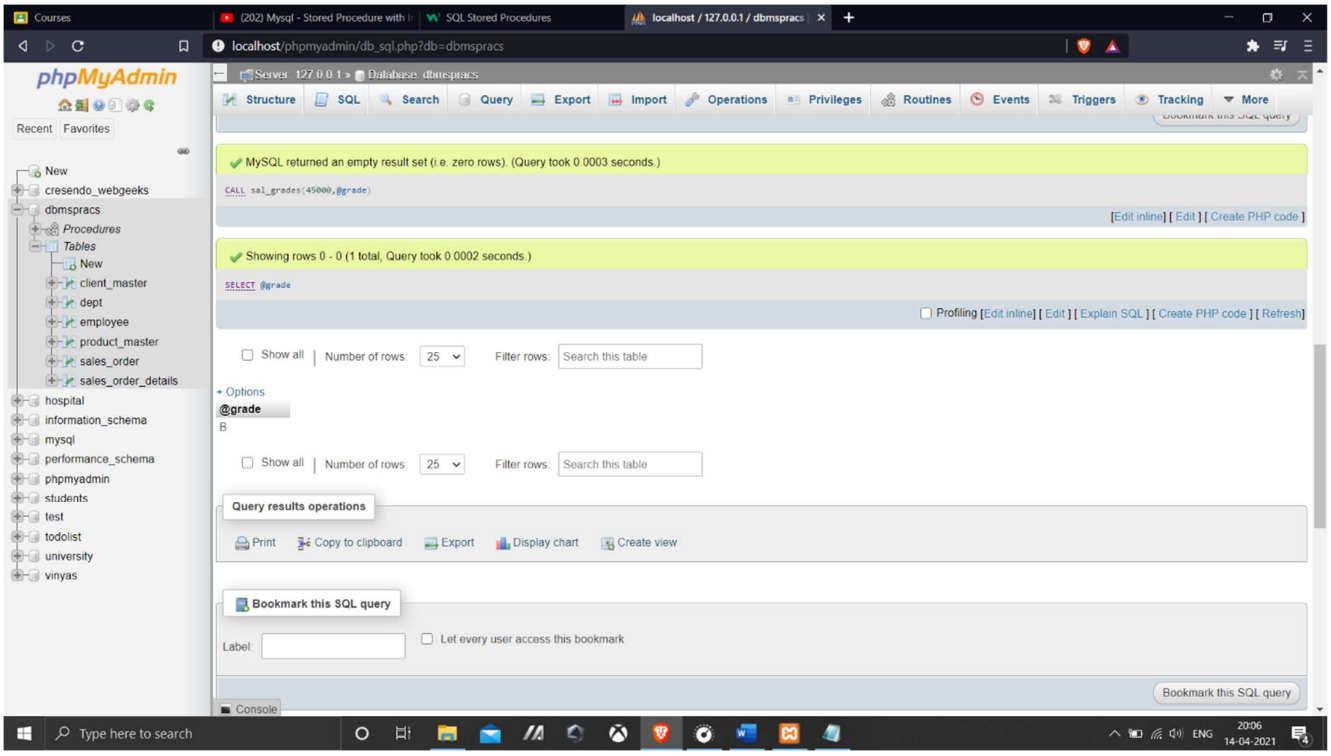
If salary between 30000 to 50000 the grade is ‘B’ and

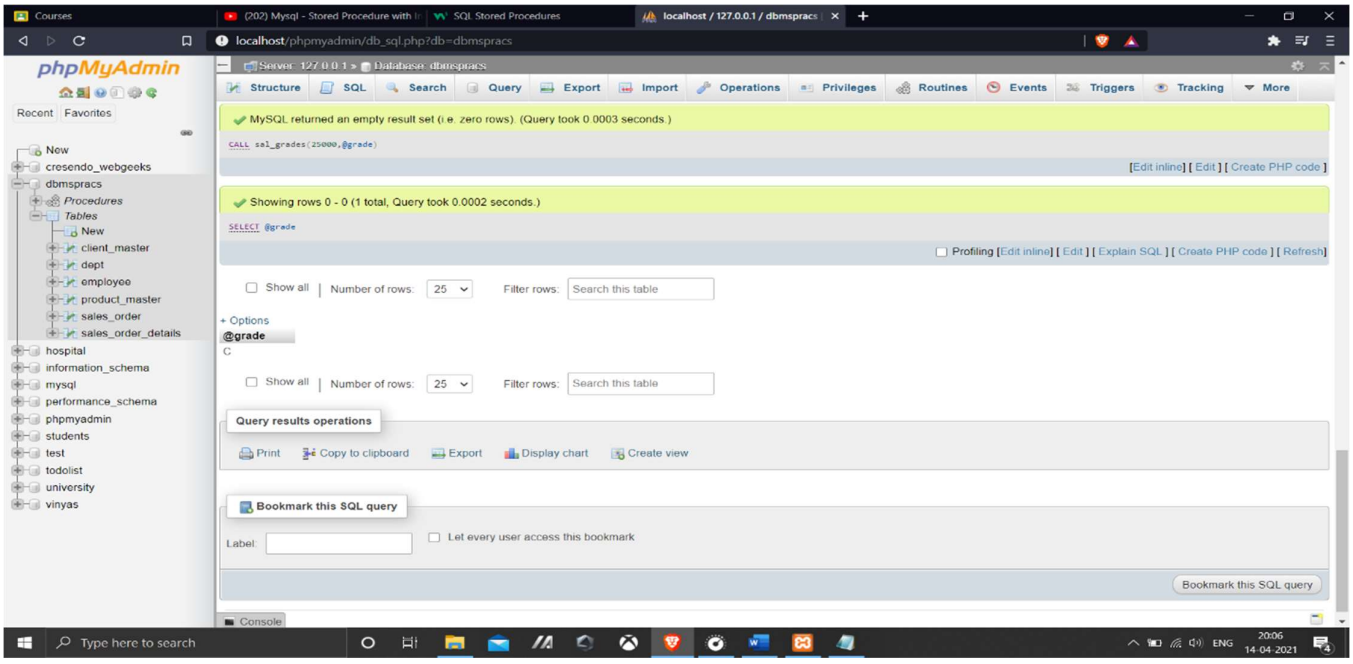
If salary < 30000 then grade is ‘C’:

```
DELIMITER //
CREATE PROCEDURE sal_grades(IN salary int, OUT grade varchar(1))
BEGIN
        IF salary >= 50000 THEN
                set grade = 'A';
        elseif salary>= 30000 then
       set grade = 'B';
   else
       set  grade = 'C';
end if;
end //
DELIMITER ;

CALL sal_grades(75000,@grade);
SELECT @grade;
CALL sal_grades(45000,@grade);
SELECT @grade;
CALL sal_grades(25000,@grade);
SELECT @grade;
```


Courses
(202) Mysql - Stored Procedure with I
SQL Stored Procedures
localhost / 127.0.0.1 / dbmspracs
localhost/phpmyadmin/db_sql.php?db=dbmspracs
phpMyAdmin
Recent
Favorites
New
cresendo_webgeeks
dbmspracs
Procedures
Tables
New
client_master
dept
employee
product_master
sales_order
sales_order_details
hospital
information_schema
mysql
performance_schema
phpmyadmin
students
test
todolist
university
vinyas
Server: 127.0.0.1 » Database: dbmspracs
Structure
SQL
Search
Query
Export
Import
Operations
Privileges
Routines
Events
Triggers
Tracking
More
Run SQL query/queries on database dbmspracs:
DELIMITER //
CREATE PROCEDURE sal_grades(IN salary int, OUT grade varchar(1))
BEGIN
IF salary >= 50000 THEN
set grade = 'A';
elseif salary>= 30000 then
set grade = 'B';
else
set grade = 'C';
end if;
end //
DELIMITER ;
Clear
Format
Get auto-saved query
Bind parameters
Bookmark this SQL query:
[Delimiter ;]
Show this query here again
Retain query box
Rollback when finished
Enable foreign key checks
Go
Hide query box
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0068 seconds.)
CREATE PROCEDURE sal_grades(IN salary int, OUT grade varchar(1)) BEGIN IF salary >= 50000 THEN set grade = 'A'; elseif salary>= 30000 then set grade = 'B'; else set grade = 'C'; end if; end
[Edit inline] [Edit] [Create PHP code]
Console
Type here to search
ENG
20:03
14-04-2021

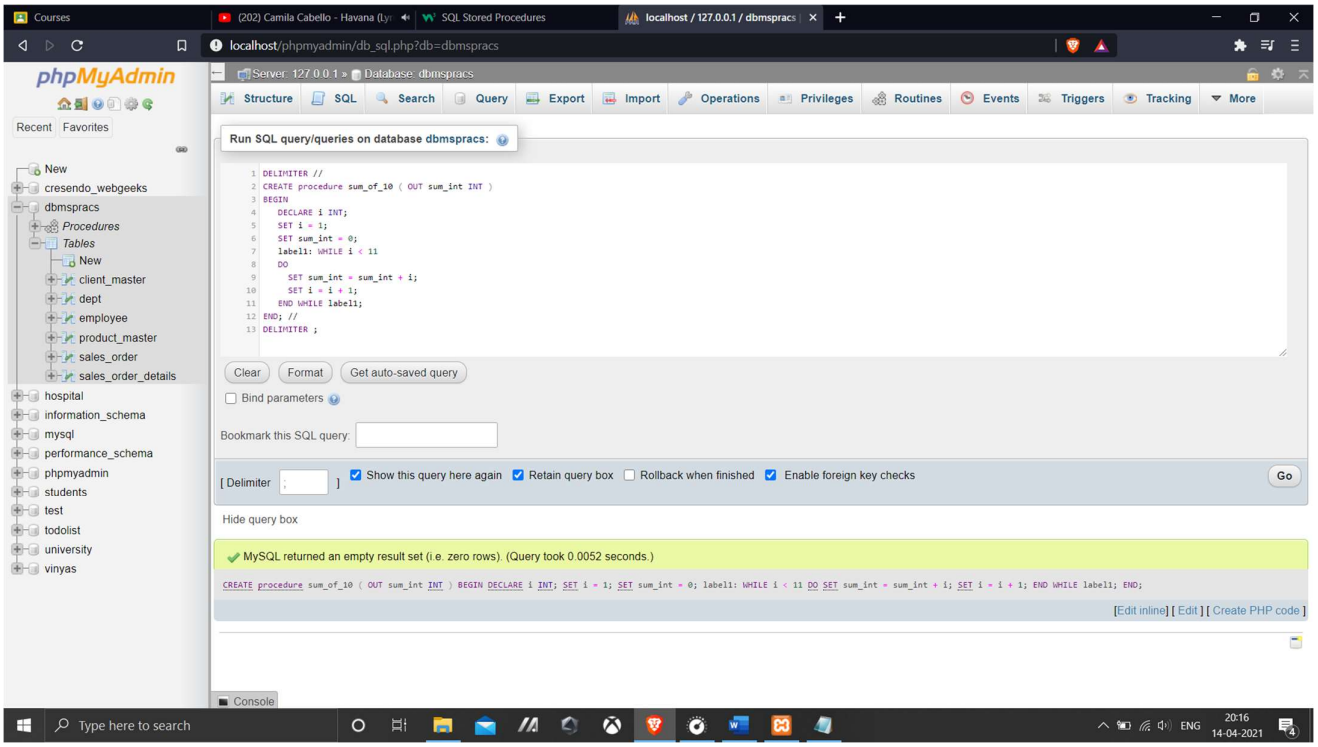

Courses
(202) Mysql - Stored Procedure with I
SQL Stored Procedures
localhost / 127.0.0.1 / dbmspracs
localhost/phpmyadmin/db_sql.php?db=dbmspracs
phpMyAdmin
Recent
Favorites
New
cresendo_webgeeks
dbmspracs
Procedures
Tables
New
client_master
dept
employee
product_master
sales_order
sales_order_details
hospital
information_schema
mysql
performance_schema
phpmyadmin
students
test
todolist
university
vinyas
Server: 127.0.0.1 » Database: dbmspracs
Structure
SQL
Search
Query
Export
Import
Operations
Privileges
Routines
Events
Triggers
Tracking
More
Bookmark this SQL query:
[Delimiter ;]
Show this query here again
Retain query box
Rollback when finished
Enable foreign key checks
Go
Hide query box
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0005 seconds.)
CALL sal_grades(75000,@grade)
[Edit inline] [Edit] [Create PHP code]
Showing rows 0 - 0 (1 total, Query took 0.0002 seconds.)
SELECT @grade
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all
Number of rows: 25
Filter rows: Search this table
+ Options
@grade
A
Show all
Number of rows: 25
Filter rows: Search this table
Query results operations
Print
Copy to clipboard
Export
Display chart
Create view
Console
Type here to search
ENG
20:06
14-04-2021

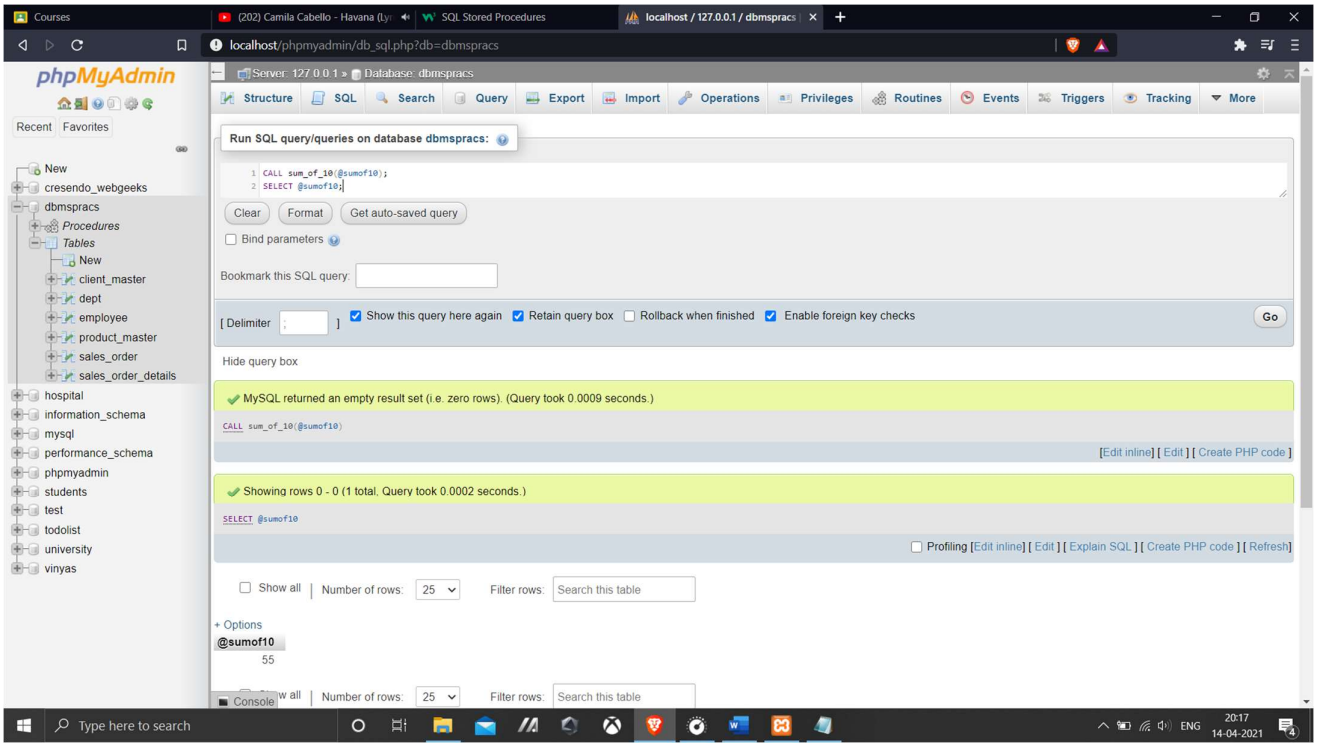

Courses
(202) Mysql - Stored Procedure with I
SQL Stored Procedures
localhost / 127.0.0.1 / dbmspracs
localhost/phpmyadmin/db_sql.php?db=dbmspracs
phpMyAdmin
Recent Favorites
New
cresendo_webgeeks
dbmspracs
Procedures
Tables
New
client_master
dept
employee
product_master
sales_order
sales_order_details
hospital
information_schema
mysql
performance_schema
phpmyadmin
students
test
todolist
university
vinyas
Server: 127.0.0.1 » Database: dbmspracs
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking More
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0003 seconds.)
CALL sal_grades(45000,@grade)
[Edit inline] [Edit] [Create PHP code]
Showing rows 0 - 0 (1 total, Query took 0.0002 seconds.)
SELECT @grade
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table
+ Options
@grade
B
Show all | Number of rows: 25 Filter rows: Search this table
Query results operations
Print Copy to clipboard Export Display chart Create view
Bookmark this SQL query
Label:
Let every user access this bookmark
Bookmark this SQL query
Console
Type here to search
ENG
20:06
14-04-2021


Courses
(202) Mysql - Stored Procedure with I
SQL Stored Procedures
localhost / 127.0.0.1 / dbmspracs
localhost/phpmyadmin/db_sql.php?db=dbmspracs
phpMyAdmin
Recent Favorites
New
cresendo_webgeeks
dbmspracs
Procedures
Tables
New
client_master
dept
employee
product_master
sales_order
sales_order_details
hospital
information_schema
mysql
performance_schema
phpmyadmin
students
test
todolist
university
vinyas
Server: 127.0.0.1 » Database: dbmspracs
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking More
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0003 seconds.)
CALL sal_grades(25000,@grade)
[Edit inline] [Edit] [Create PHP code]
Showing rows 0 - 0 (1 total, Query took 0.0002 seconds.)
SELECT @grade
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table
+ Options
@grade
C
Show all | Number of rows: 25 Filter rows: Search this table
Query results operations
Print Copy to clipboard Export Display chart Create view
Bookmark this SQL query
Label:
Let every user access this bookmark
Bookmark this SQL query
Console
Type here to search
ENG
20:06
14-04-2021

2. To Create Table Employee: 2. Write a block to display sum of 1 to 10 numbers

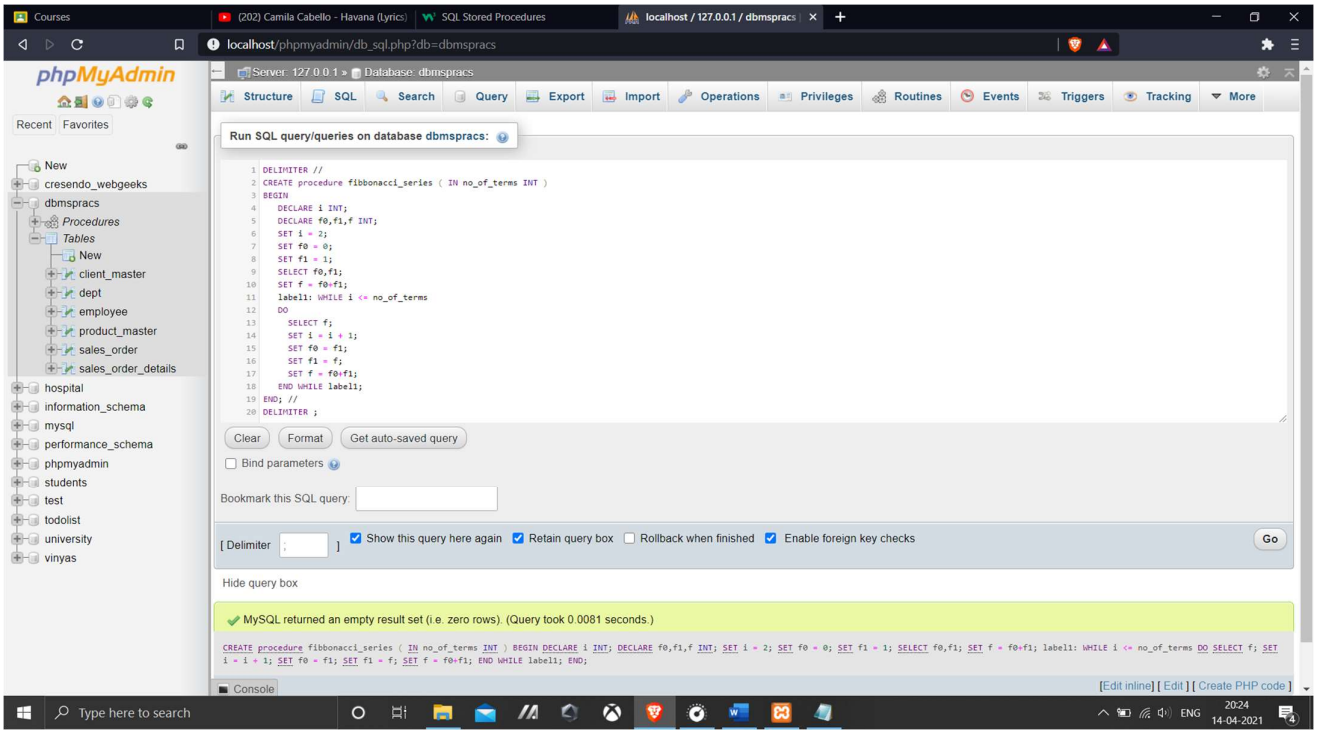
```
DELIMITER //
CREATE procedure sum_of_10 ( OUT sum_int INT )
BEGIN
  DECLARE i INT;
  SET i = 1;
  SET sum_int = 0;
  label1: WHILE i < 11
  DO
       SET sum_int = sum_int + i;
    SET i = i + 1;
  END WHILE label1;
END; //
DELIMITER ;

CALL sum_of_10(@sumof10);
SELECT @sumof10;
```

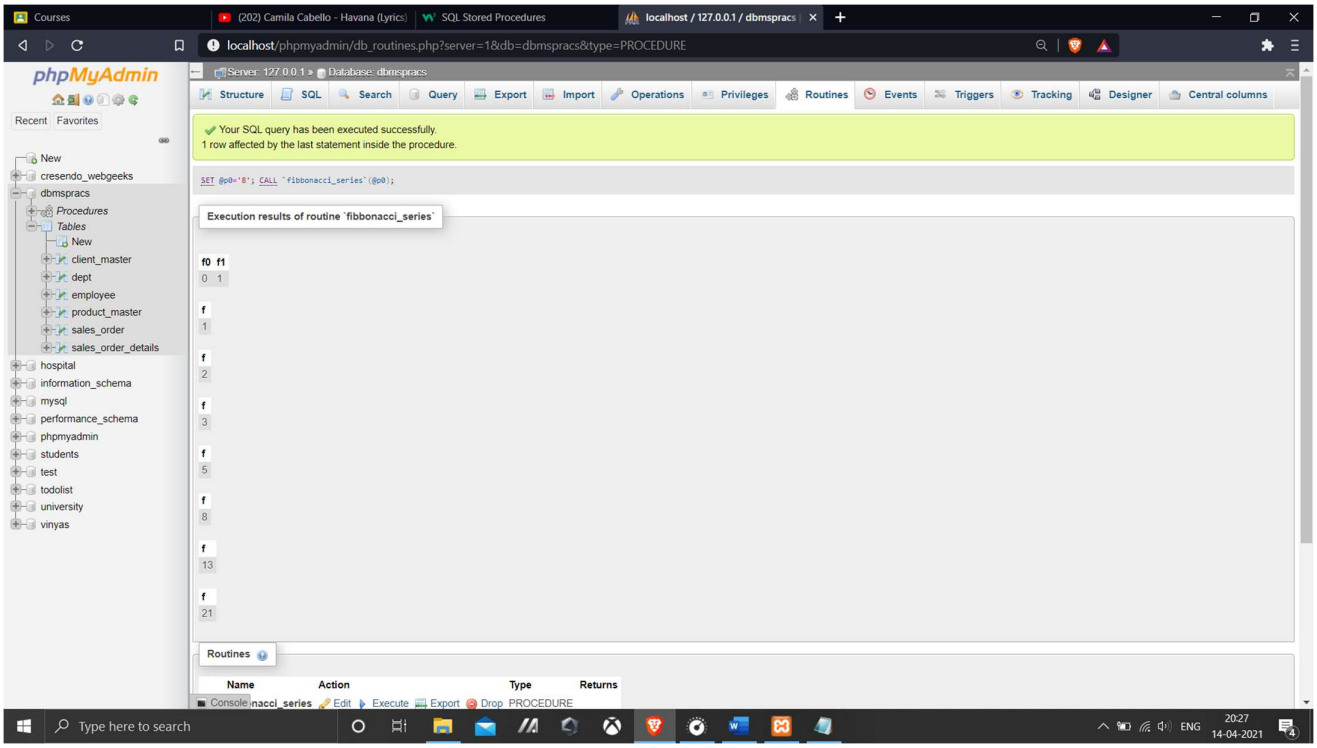

Courses
(202) Camila Cabello - Havana (Ly
SQL Stored Procedures
localhost / 127.0.0.1 / dbmspracs
localhost/phpmyadmin/db_sql.php?db=dbmspracs
phpMyAdmin
Recent Favorites
New
cresendo_webgeeks
dbmspracs
Procedures
Tables
New
client_master
dept
employee
product_master
sales_order
sales_order_details
hospital
information_schema
mysql
performance_schema
phpmyadmin
students
test
todolist
university
vinyas
Server: 127.0.0.1 » Database: dbmspracs
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking More
Run SQL query/queries on database dbmspracs:
DELIMITER //
CREATE procedure sum_of_10 (OUT sum_int INT)
BEGIN
DECLARE i INT;
SET i = 1;
SET sum_int = 0;
label1: WHILE i < 11
DO
SET sum_int = sum_int + i;
SET i = i + 1;
END WHILE label1;
END; //
DELIMITER ;
Clear Format Get auto-saved query
Bind parameters
Bookmark this SQL query:
[Delimiter ;] Show this query here again Retain query box Rollback when finished Enable foreign key checks
Go
Hide query box
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0052 seconds.)
CREATE procedure sum_of_10 (OUT sum_int INT) BEGIN DECLARE i INT; SET i = 1; SET sum_int = 0; label1: WHILE i < 11 DO SET sum_int = sum_int + i; SET i = i + 1; END WHILE label1; END;
[Edit inline] [Edit] [Create PHP code]
Console
Type here to search
ENG
20:16
14-04-2021


Courses
(202) Camila Cabello - Havana (Ly
SQL Stored Procedures
localhost / 127.0.0.1 / dbmspracs
localhost/phpmyadmin/db_sql.php?db=dbmspracs
phpMyAdmin
Recent Favorites
New
cresendo_webgeeks
dbmspracs
Procedures
Tables
New
client_master
dept
employee
product_master
sales_order
sales_order_details
hospital
information_schema
mysql
performance_schema
phpmyadmin
students
test
todolist
university
vinyas
Server: 127.0.0.1 » Database: dbmspracs
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking More
Run SQL query/queries on database dbmspracs:
CALL sum_of_10(@sumof10);
SELECT @sumof10;
Clear Format Get auto-saved query
Bind parameters
Bookmark this SQL query:
[Delimiter ;] Show this query here again Retain query box Rollback when finished Enable foreign key checks
Go
Hide query box
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0009 seconds.)
CALL sum_of_10(@sumof10)
[Edit inline] [Edit] [Create PHP code]
Showing rows 0 - 0 (1 total, Query took 0.0002 seconds.)
SELECT @sumof10
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table
+ Options
@sumof10
55
Console
Type here to search
ENG
20:17
14-04-2021

3. Write a block to display Fibonacci series upto 8th term (start with 0,1)

```
DELIMITER //
CREATE procedure fibbonacci_series ( IN no_of_terms INT )
BEGIN
  DECLARE i INT;
  DECLARE f0,f1,f INT;
  SET i = 2;
  SET f0 = 0;
  SET f1 = 1;
  SELECT f0,f1;
  SET f = f0+f1;
  label1: WHILE i <= no_of_terms
  DO
       SELECT f;
    SET i = i + 1;
    SET f0 = f1;
    SET f1 = f;
    SET f = f0+f1;
  END WHILE label1;
END; //
DELIMITER ;

SET @p0='8';
CALL `fibbonacci_series`(@p0);
```

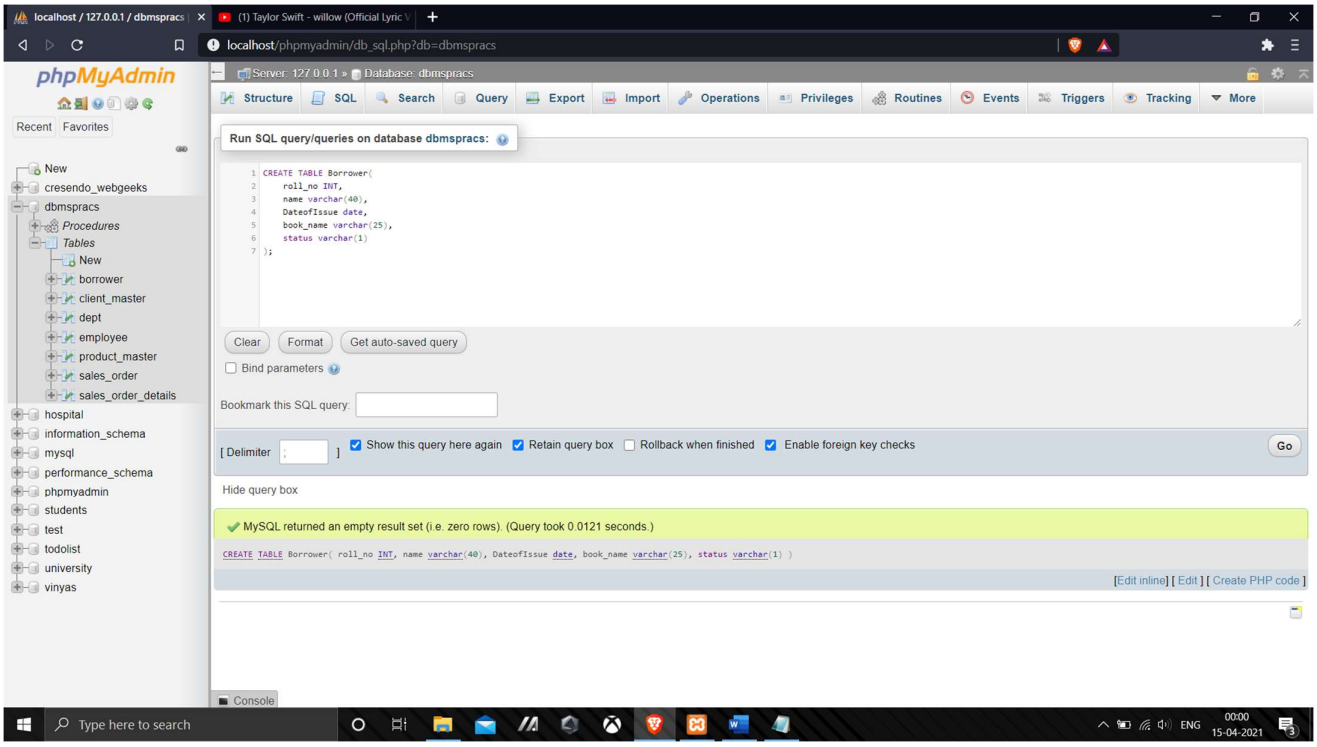

Courses
(202) Camila Cabello - Havana (Lyrics)
SQL Stored Procedures
localhost / 127.0.0.1 / dbmspracs
localhost/phpmyadmin/db_sql.php?db=dbmspracs
phpMyAdmin
Recent Favorites
New
cresendo_webgeeks
dbmspracs
Procedures
Tables
New
client_master
dept
employee
product_master
sales_order
sales_order_details
hospital
information_schema
mysql
performance_schema
phpmyadmin
students
test
todolist
university
vinyas
Server: 127.0.0.1 » Database: dbmspracs
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking More
Run SQL query/queries on database dbmspracs:
DELIMITER //
CREATE procedure fibbonacci_series (IN no_of_terms INT)
BEGIN
DECLARE i INT;
DECLARE f0,f1,f INT;
SET i = 2;
SET f0 = 0;
SET f1 = 1;
SELECT f0,f1;
SET f = f0+f1;
label1: WHILE i <= no_of_terms
DO
SELECT f;
SET i = i + 1;
SET f0 = f1;
SET f1 = f;
SET f = f0+f1;
END WHILE label1;
END; //
DELIMITER ;
Clear Format Get auto-saved query
Bind parameters
Bookmark this SQL query:
[Delimiter ;] Show this query here again Retain query box Rollback when finished Enable foreign key checks
Go
Hide query box
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0081 seconds.)
CREATE procedure fibbonacci_series (IN no_of_terms INT) BEGIN DECLARE i INT; DECLARE f0,f1,f INT; SET i = 2; SET f0 = 0; SET f1 = 1; SELECT f0,f1; SET f = f0+f1; label1: WHILE i <= no_of_terms DO SELECT f; SET i = i + 1; SET f0 = f1; SET f1 = f; SET f = f0+f1; END WHILE label1; END;
Console
[Edit inline] [Edit] [Create PHP code]
Type here to search
ENG
20:24
14-04-2021


Courses
(202) Camila Cabello - Havana (Lyrics)
SQL Stored Procedures
localhost / 127.0.0.1 / dbmspracs
localhost/phpmyadmin/db_routines.php?server=1&db=dbmspracs&type=PROCEDURE
phpMyAdmin
Recent Favorites
New
cresendo_webgeeks
dbmspracs
Procedures
Tables
New
client_master
dept
employee
product_master
sales_order
sales_order_details
hospital
information_schema
mysql
performance_schema
phpmyadmin
students
test
todolist
university
vinyas
Server: 127.0.0.1 » Database: dbmspracs
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking Designer Central columns
Your SQL query has been executed successfully.
1 row affected by the last statement inside the procedure.
SET @p0='8'; CALL `fibbonacci_series`(@p0);
Execution results of routine `fibbonacci_series`
f0 f1
0 1
f
1
f
2
f
3
f
5
f
8
f
13
f
21
Routines
Name Action Type Returns
nacci_series Edit Execute Export Drop PROCEDURE
Console
Type here to search
ENG
20:27
14-04-2021

Task 2

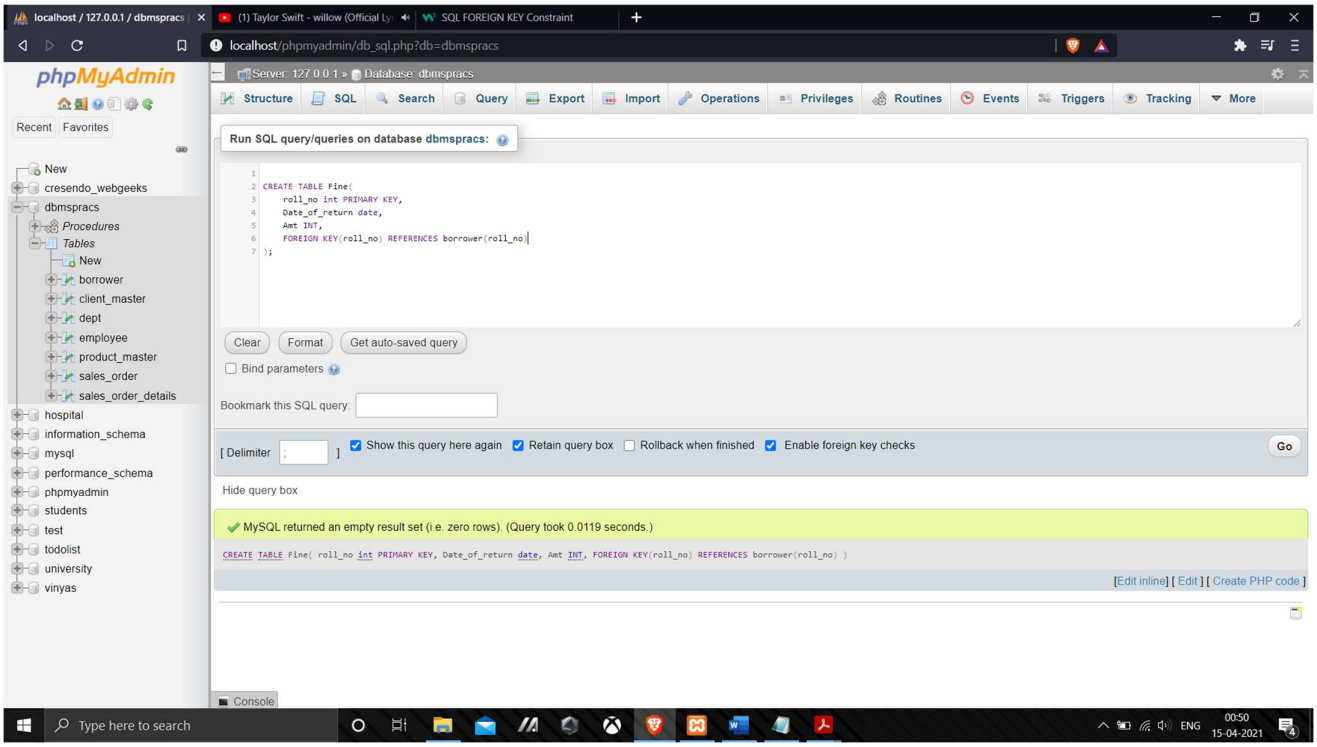
1. Creating Table Borrower:

```
CREATE TABLE Borrower(
    roll_no INT PRIMARY KEY,
    name varchar(40),
    DateofIssue date,
    book_name varchar(25),
    status varchar(1)
);
```



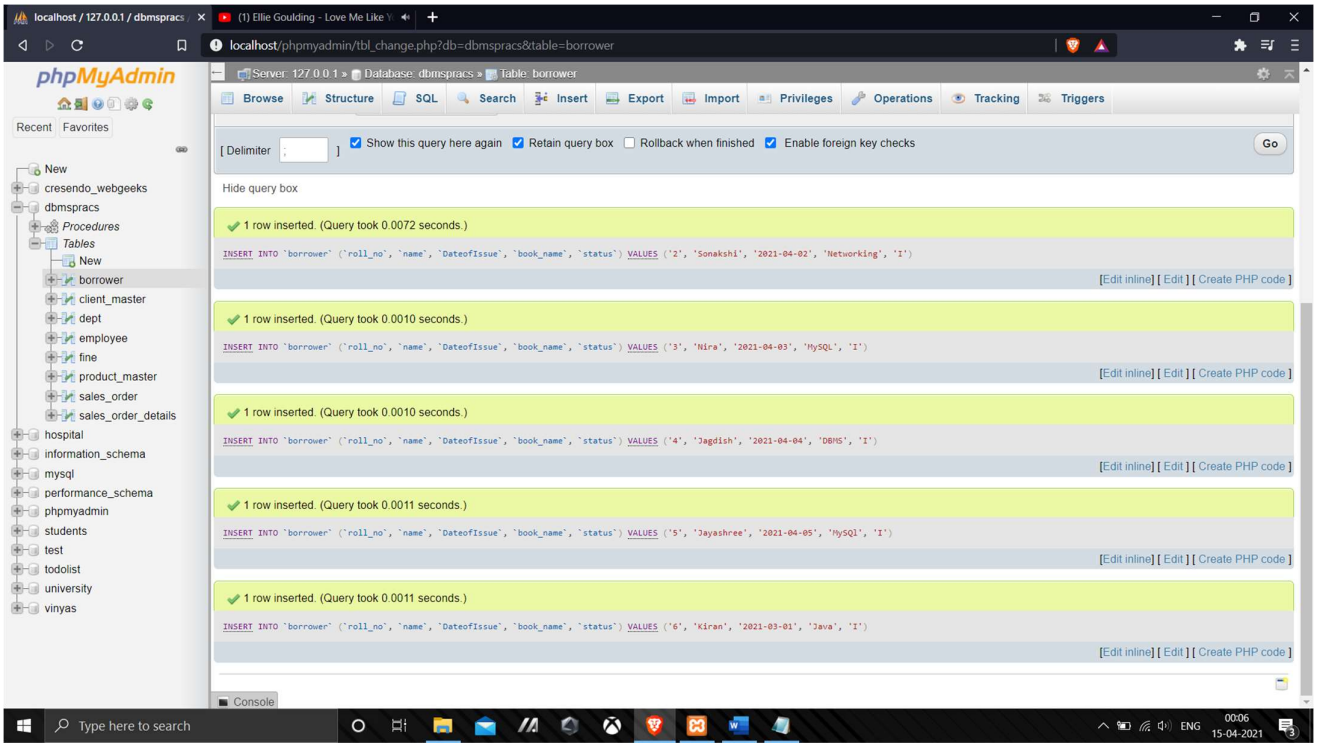
2. Creating Table fine:

```
CREATE TABLE Fine(
    roll_no int PRIMARY KEY,
    Date_of_return date,
    Amt INT,
    FOREIGN KEY(roll_no) REFERENCES borrower(roll_no)
);
```



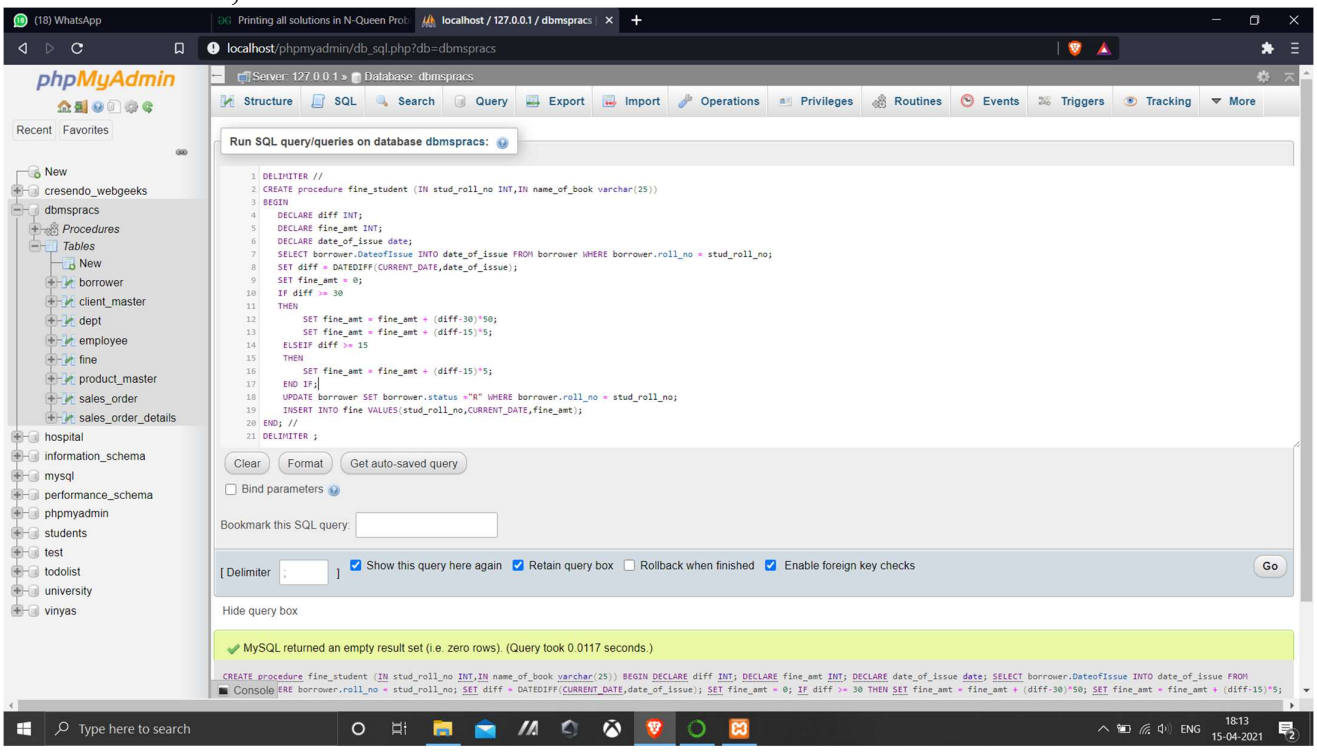
3. Inserting the necessary values:

```
INSERT INTO `borrower` (`roll_no`, `name`, `DateofIssue`, `book_name`, `status`) VALUES ('1', 'Amrita', '2021-04-01', 'Java', 'I');

INSERT INTO `borrower` (`roll_no`, `name`, `DateofIssue`, `book_name`, `status`) VALUES ('2', 'Sonakshi', '2021-04-02', 'Networking', 'I');

INSERT INTO `borrower` (`roll_no`, `name`, `DateofIssue`, `book_name`, `status`) VALUES ('3', 'Nira', '2021-04-03', 'MySQL', 'I');

INSERT INTO `borrower` (`roll_no`, `name`, `DateofIssue`, `book_name`, `status`) VALUES ('4', 'Jagdish', '2021-04-04', 'DBMS', 'I');

INSERT INTO `borrower` (`roll_no`, `name`, `DateofIssue`, `book_name`, `status`) VALUES ('5', 'Jayashree', '2021-04-05', 'MySQl', 'I');

INSERT INTO `borrower` (`roll_no`, `name`, `DateofIssue`, `book_name`, `status`) VALUES ('6', 'Kiran', '2021-03-01', 'Java', 'I');
```

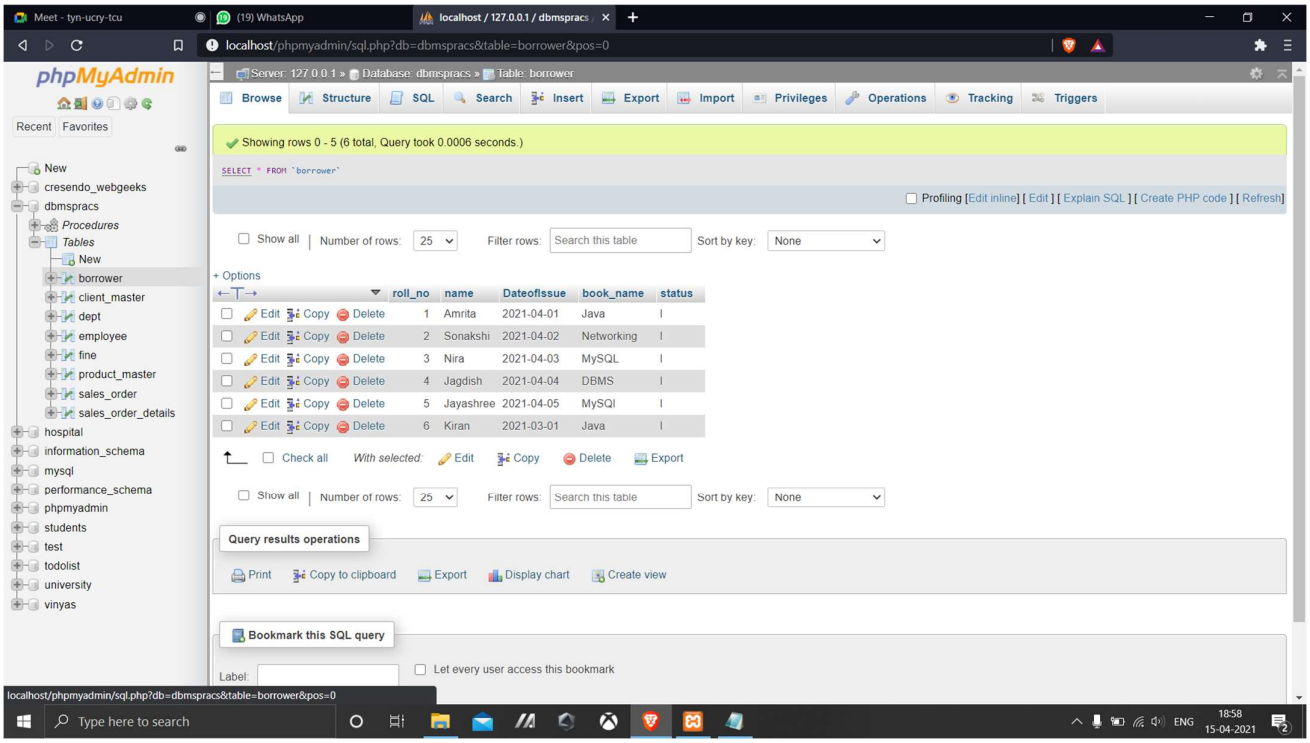


4. Creating the Procedure:

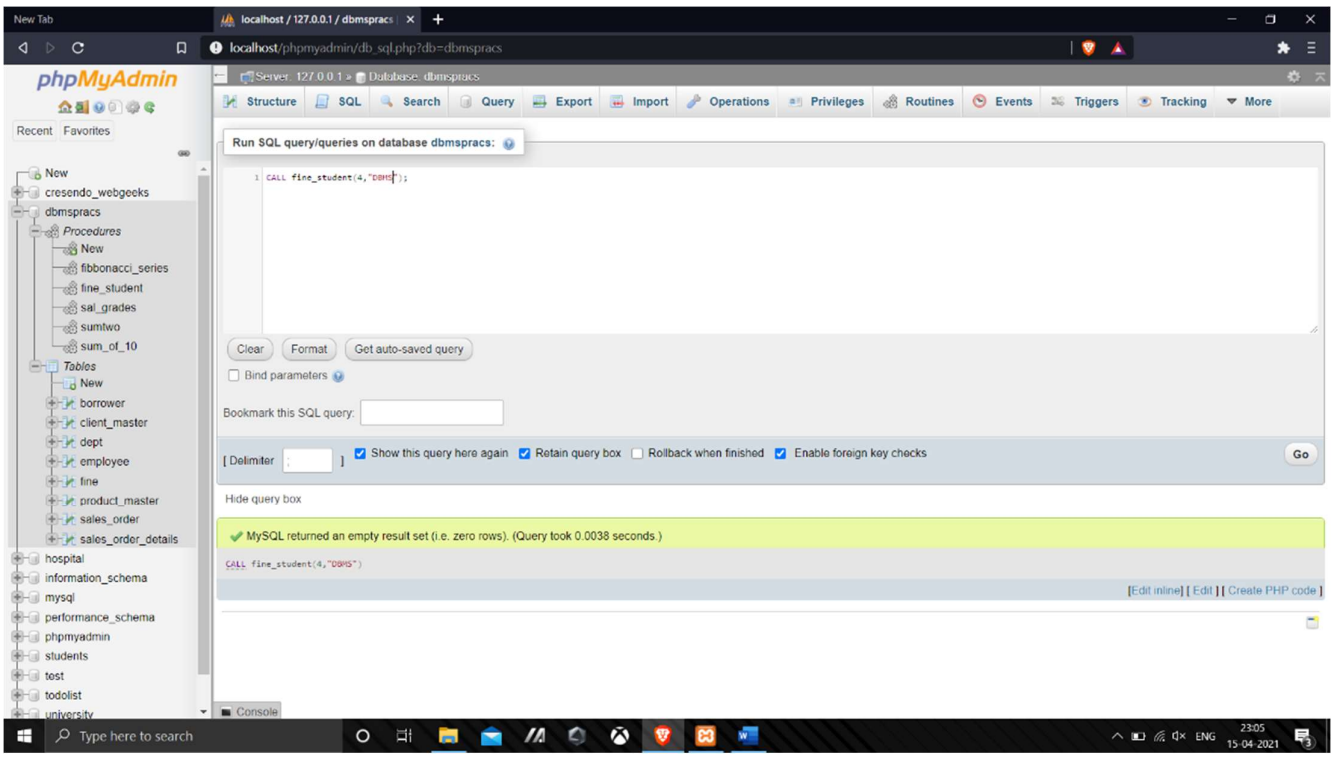
```
DELIMITER //
CREATE procedure fine_student (IN stud_roll_no INT,IN name_of_book
varchar(25))
BEGIN
  DECLARE diff INT;
  DECLARE fine_amt INT;
  DECLARE date_of_issue date;
  SELECT borrower.DateofIssue INTO date_of_issue FROM borrower WHERE
borrower.roll_no = stud_roll_no;
  SET diff = DATEDIFF(CURRENT_DATE,date_of_issue);
  SET fine_amt = 0;
  IF diff >= 30
  THEN
            SET fine_amt = fine_amt + (diff-30)*50;
     SET fine_amt = fine_amt + (diff-15)*5;
   ELSEIF diff >= 15
   THEN
      SET fine_amt = fine_amt + (diff-15)*5;
   END IF;
   UPDATE borrower SET borrower.status ="R" WHERE borrower.roll_no =
stud_roll_no;
   INSERT INTO fine VALUES(stud_roll_no,CURRENT_DATE,fine_amt);
END; //
DELIMITER ;
```



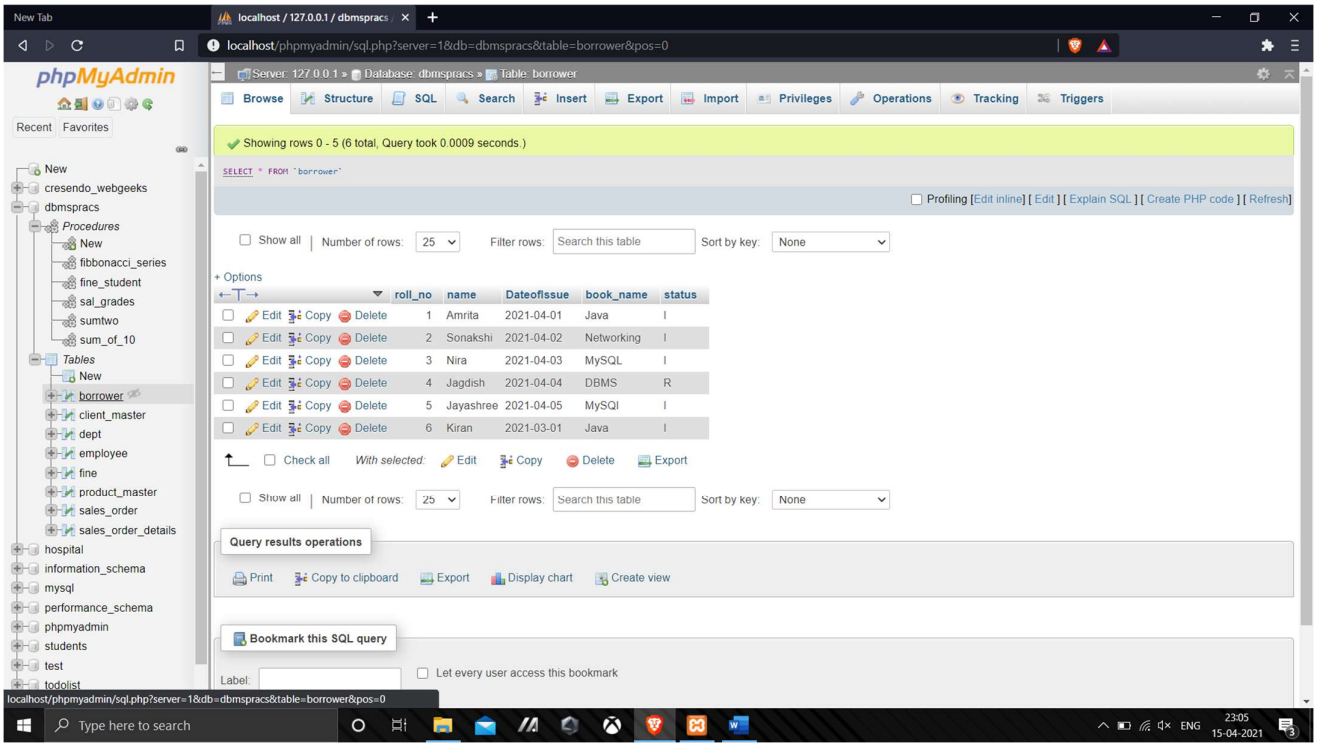
Before Calling fine_student procedure:

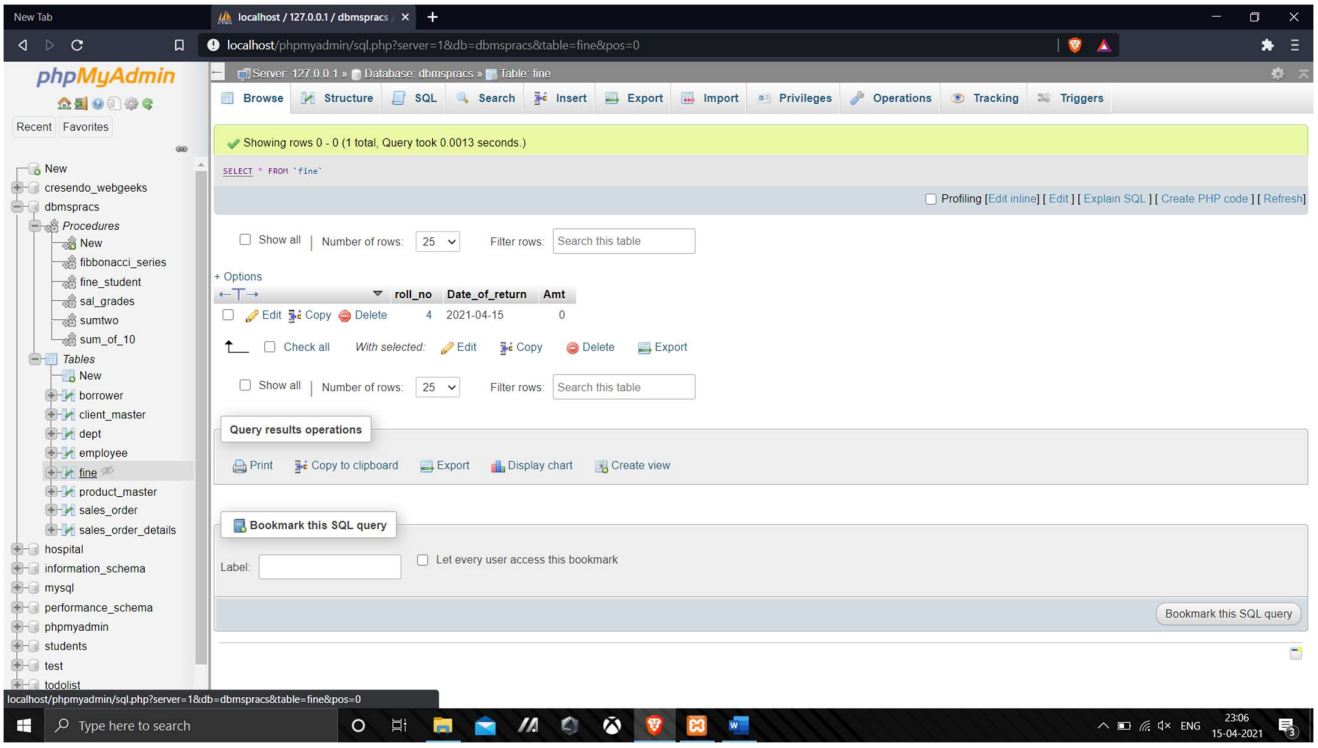


Calling fine_student procedure:



After Calling fine_student procedure:





Brendon Lucas
8953
BE Comps B

Exp - 8 - Postlab

1) Advantages of PL/SQL vs SQL

(a) PL/SQL is a database programming language using SQL, while SQL is only database query language.

(b) PL/SQL allows ____, declaration of data variable while SQL does not.

(c) Control structures are available like, For loop, while loop whereas SQL does not support Control structures.

(d) PL/SQL block performs Group of Operations as single block whereas SQL performs Single Query operation

(e) It does not interacts directly with the database server and PL/SQL is application oriented Language in Contrast SQL is Data oriented language and d It directly interacts with database server.

(f) PL/SQL is accustomed write program blocks, functions, procedures, triggers and packages. in Constraxt SQL is used to write queries, DDL and DML Statements.

Q2 Explain data types of PL/SQL

Ans

(1) NUMERIC DATATYPES:-

(a) INT

ANSI specific integer type with maximum precision of 38 decimal digits

(b) FLOAT.

ANSI and IBM specific floating point type with maximum precision of 126 binary digits.

(2) CHARACTER DATATYPES:-

Varchar2 :- Variable-length character string with maximum size of 32,767 bytes

(3) DATETIME and INTERVAL DATATYPES:-

DAY

01 to 31 (valid by Month & Year).

MONTH

01 to 12.

YEAR

-4712 to 9999

LARGE OBJECT (LOB) DATATYPES

BLOB

Used to store large binary object upto 128 TB.