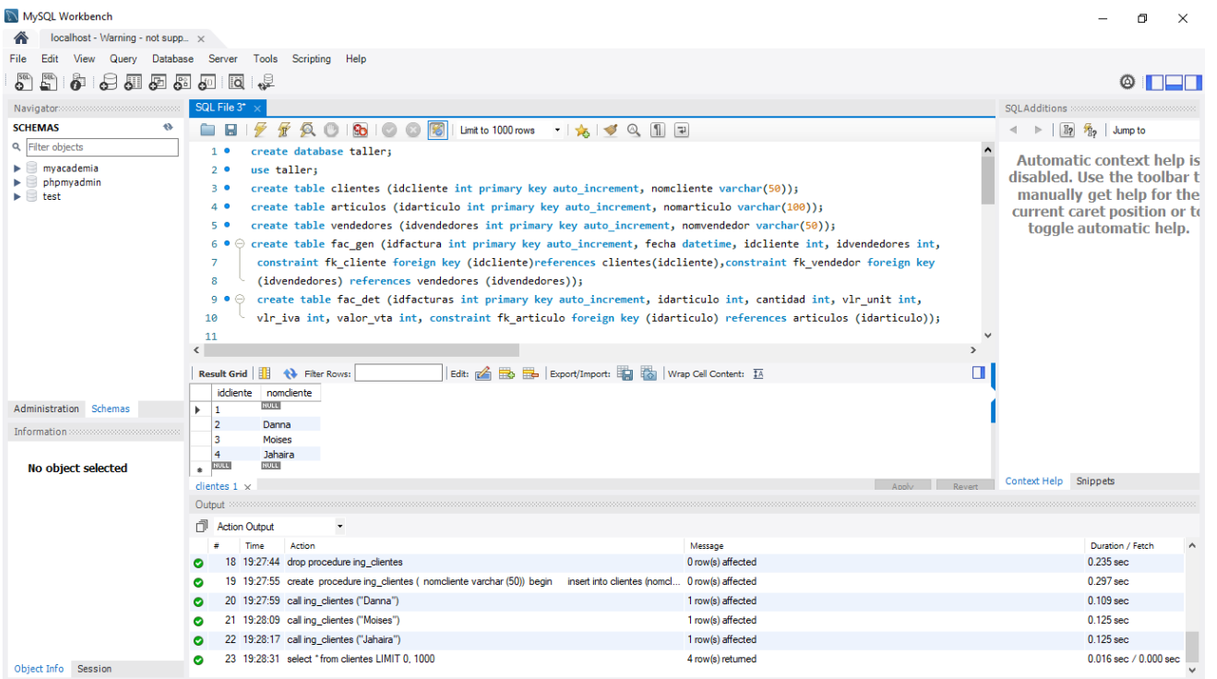
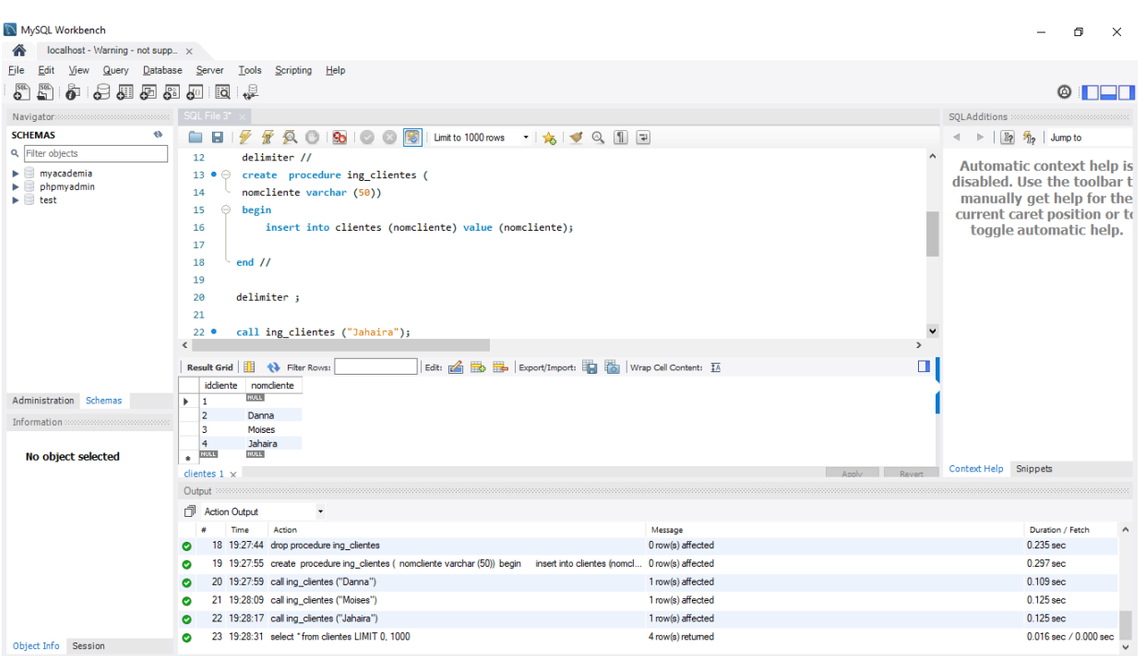
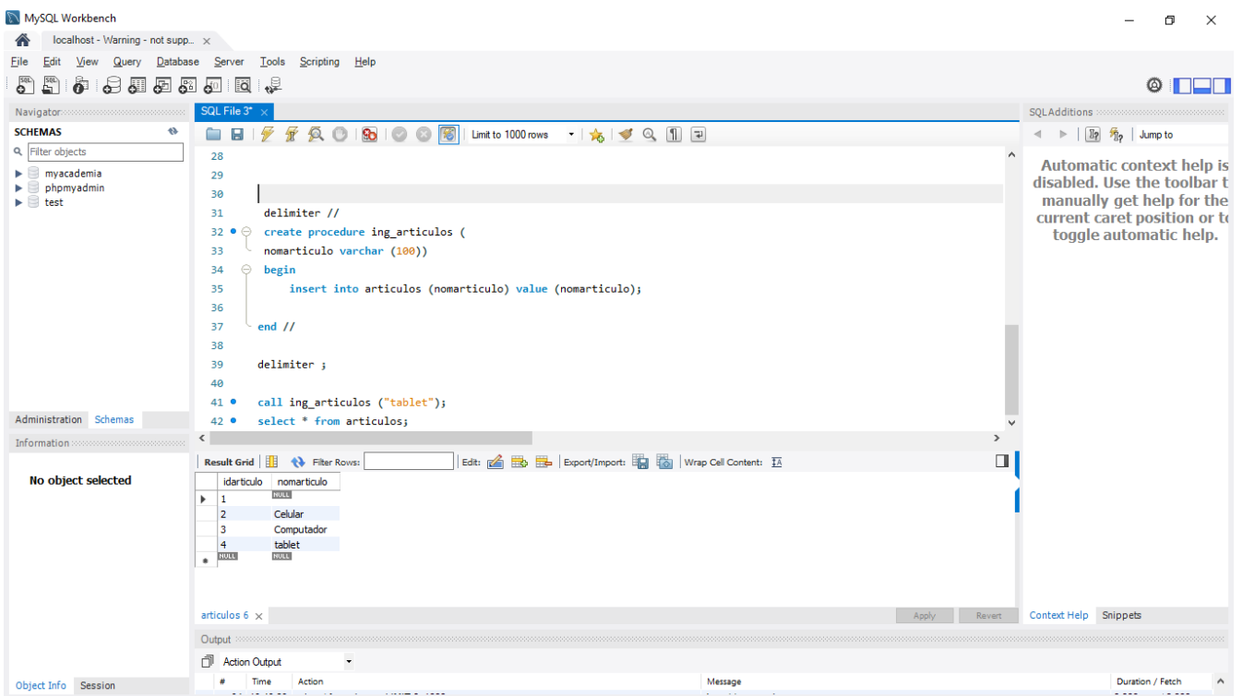
Base de datos – Moises Zabaleta, Danna Beltran, Jahaira Palencia

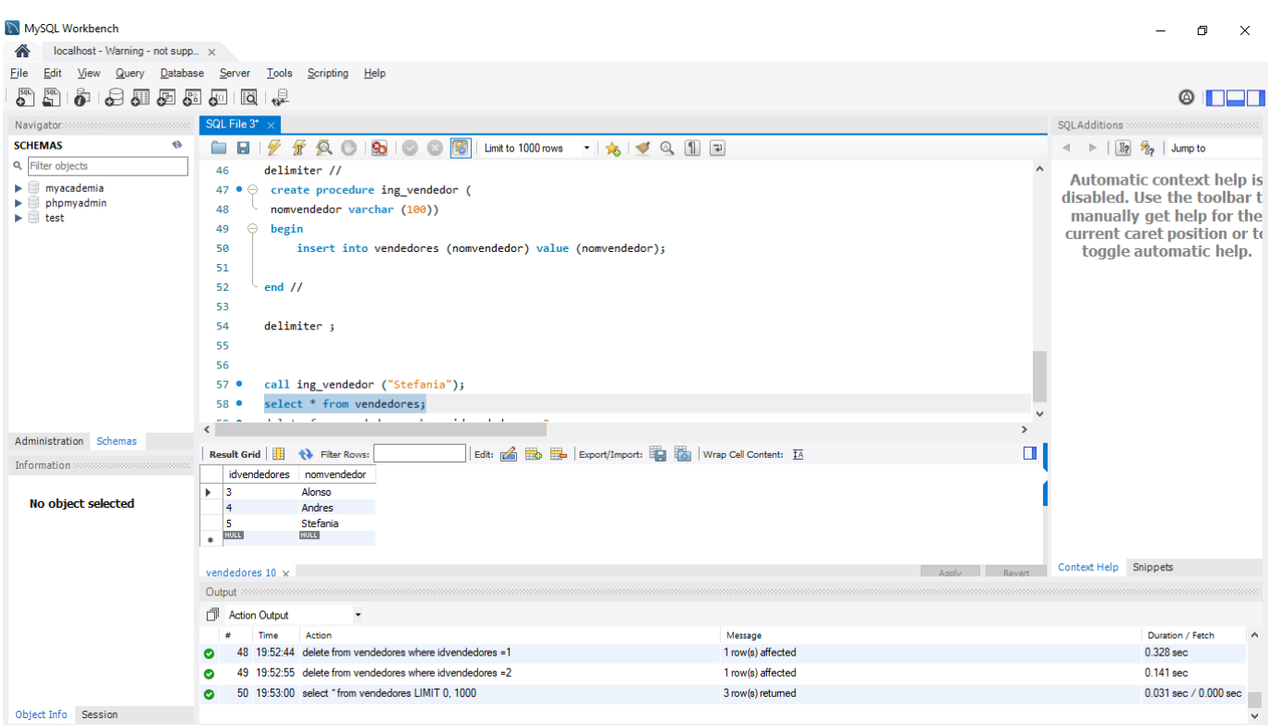
1. Base de datos



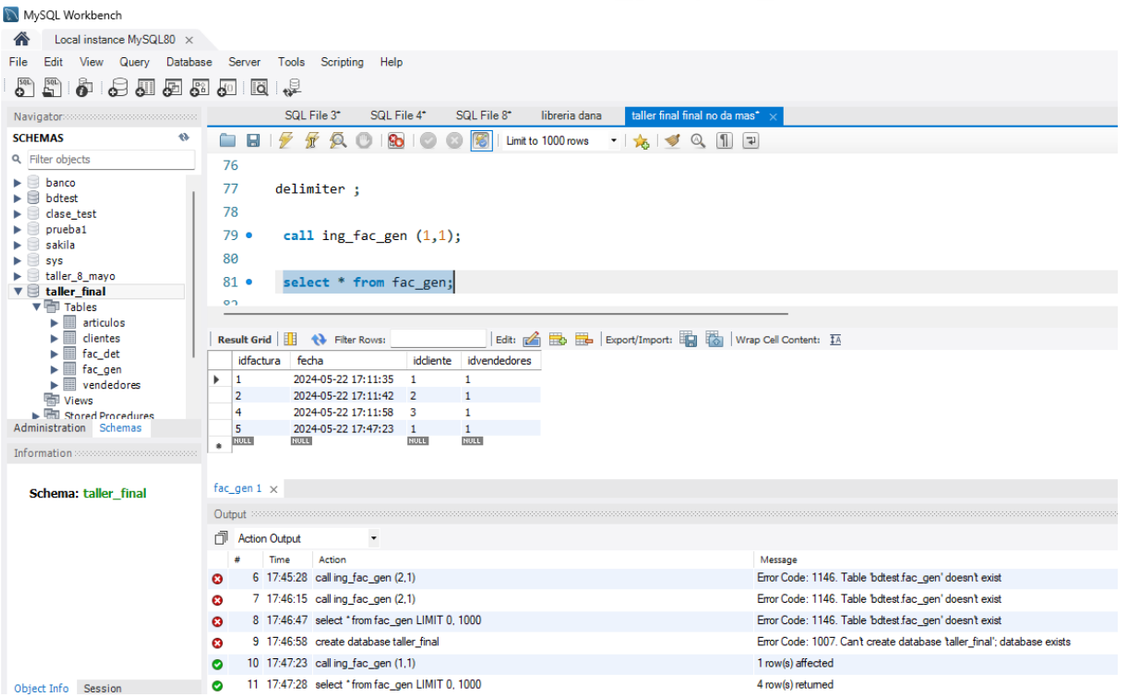
2. Procedimientos almacenados

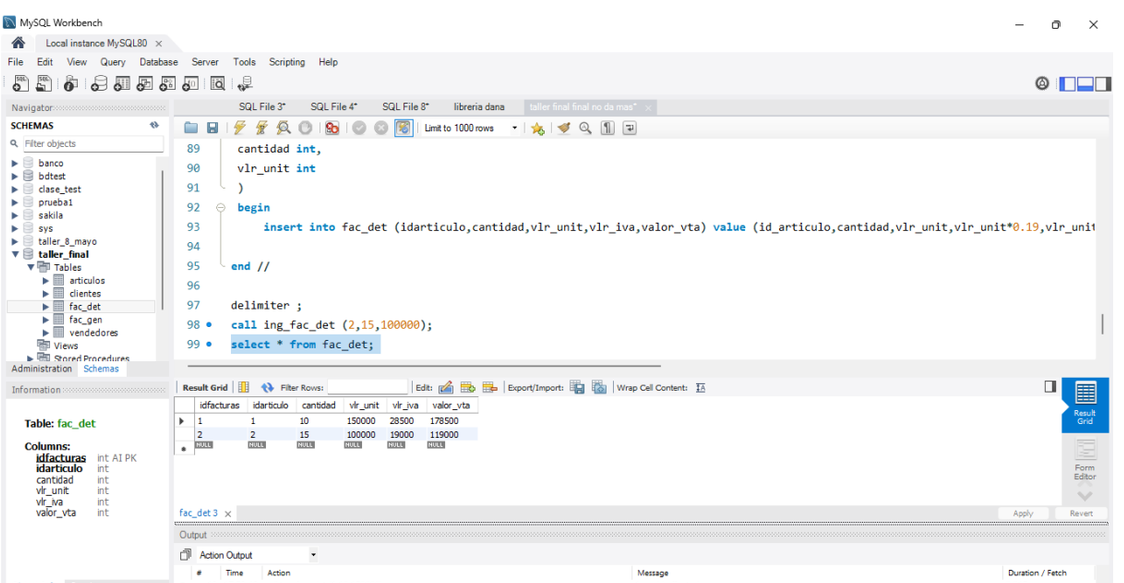



MySQL Workbench
localhost - Warning - not supp...
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
myacademia
phpmyadmin
test
SQL File 3*
Limit to 1000 rows
delimiter //
create procedure ing_articulos (
nomarticulo varchar (100))
begin
insert into articulos (nomarticulo) value (nomarticulo);
end //
delimiter ;
call ing_articulos ("tablet");
select * from articulos;
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar t manually get help for the current caret position or t toggle automatic help.
Result Grid
Filter Rows:
Edit:
Export/Import:
Wrap Cell Content:
idarticulo nomarticulo
1 NULL
2 Celular
3 Computador
4 tablet
NULL NULL
Administration Schemas
Information
No object selected
articulos 6
Apply
Revert
Context Help
Snippets
Output
Action Output
Time Action Message Duration / Fetch
Object Info
Session


MySQL Workbench
localhost - Warning - not supp...
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
myacademia
phpmyadmin
test
SQL File 3*
Limit to 1000 rows
delimiter //
create procedure ing_vendedor (
nomvendedor varchar (100))
begin
insert into vendedores (nomvendedor) value (nomvendedor);
end //
delimiter ;
call ing_vendedor ("Stefania");
select * from vendedores;
SQLAdditions
Jump to
Automatic context help is disabled. Use the toolbar t manually get help for the current caret position or t toggle automatic help.
Result Grid
Filter Rows:
Edit:
Export/Import:
Wrap Cell Content:
idvendedores nomvendedor
3 Alonso
4 Andres
5 Stefania
NULL NULL
Administration Schemas
Information
No object selected
vendedores 10
Apply
Revert
Context Help
Snippets
Output
Action Output
Time Action Message Duration / Fetch
48 19:52:44 delete from vendedores where idvendedores =1 1 row(s) affected 0.328 sec
49 19:52:55 delete from vendedores where idvendedores =2 1 row(s) affected 0.141 sec
50 19:53:00 select * from vendedores LIMIT 0, 1000 3 row(s) returned 0.031 sec / 0.000 sec
Object Info
Session

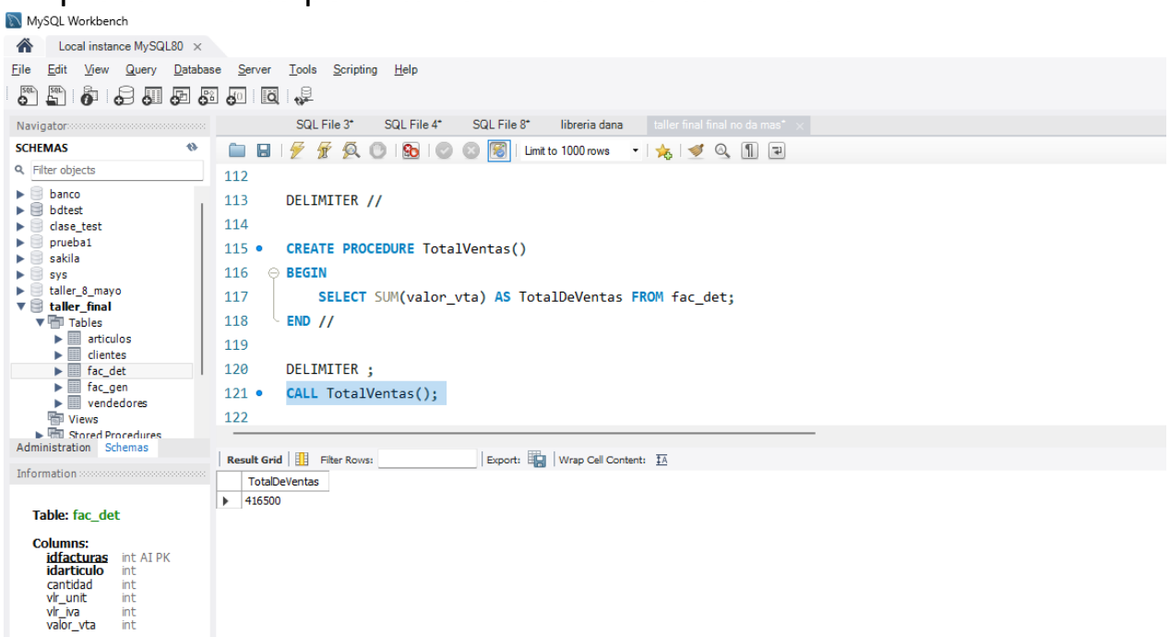
3. Procedimiento almacenado en las tablas fac_gen y fac_det





4. FUNCION SUM()

a. procedimiento permita ver el total de las ventas



b. procedimiento permita ver el total de las ventas por cliente

```
130 CREATE PROCEDURE TotalVentas_Cliente(
131     in ing_idcliente INT
132 )
133 BEGIN
134     SELECT
135         idcliente,
136         SUM(fac_det.valor_vta) AS TotalDeVentas
137     FROM
138         fac_gen
139     JOIN
140         fac_det ON idfactura = idfactura
141     WHERE
142         fac_gen.idcliente = ing_idcliente
143         GROUP BY
144         fac_gen.idcliente;
145 END //
146 DELIMITER ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Conte

idcliente	TotalDeVentas
2	2082500

c. Crear un procedimiento permita ver el total de las ventas por vendedor

```
Limit to 1000 rows

154     DELIMITER //
155     CREATE PROCEDURE TotalVentas_vendedor(
156         in ing_idvendedores INT
157     )
158     BEGIN
159         SELECT
160             idvendedores,
161             SUM(fac_det.valor_vta) AS TotalDeVentas
162         FROM
163             fac_gen
164         JOIN
165             fac_det ON idfactura = idfactura
166         WHERE
167             fac_gen.idvendedores = ing_idvendedores
168             GROUP BY
169             fac_gen.idvendedores;
170     END //
171     DELIMITER ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

idvendedores	TotalDeVentas
2	833000

Result 18

5. FUNCION AVG()

a. Crear un procedimiento permita ver el promedio de las ventas

```
177
178
179     -- ----------------------------------
180     DELIMITER //
181     CREATE PROCEDURE promedioVentas()
182     BEGIN
183         SELECT AVG(valor_vta) AS PromedioDeVentas FROM fac_det;
184     END //
185
186     DELIMITER ;
187
188     CALL promedioVentas();
189
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

TotalDeVentas
133875.0000

b. Crear un procedimiento permita ver el promedio de las ventas por cliente

```
196     DELIMITER //
197     CREATE PROCEDURE promedioVentas_Cliente(
198         in ing_idcliente INT
199     )
200     BEGIN
201         SELECT
202             idcliente,
203             AVG(fac_det.valor_vta) AS PromedioDeVentas
204         FROM
205             fac_gen
206         JOIN
207             fac_det ON idfactura = idfactura
208         WHERE
209             fac_gen.idcliente = ing_idcliente
210             GROUP BY
211             fac_gen.idcliente;
212     END //
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

idcliente	TotalDeVentas
3	43987.5000

c. Crear un procedimiento permita ver el promedio de las ventas por vendedor

```
DELIMITER //
CREATE PROCEDURE promedioVentas_vendedor(
    in ing_idvendedores INT
)
BEGIN
    SELECT
        idvendedores,
        AVG(fac_det.valor_vta) AS PromedioDeVentas
    FROM
        fac_gen
    JOIN
        fac_det ON idfactura = idfactura
    WHERE
        fac_gen.idvendedores = ing_idvendedores
        GROUP BY
        fac_gen.idvendedores;
END //
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

idvendedores	PromedioDeVentas
2	88780.2632