

**Universidad Politécnica
de Chiapas**

UNIVERSIDAD POLITÉCNICA DE CHIAPAS

Ingeniería Biomédica



Materia

Telesalud

Reporte BD

Profesor

Dr. Santoyo Espinosa Enrique Tadeo

Alumnos

Domínguez González Marco Antonio 181194

Laura Patricia Espinoza Hernández 183160

Enrique Alonso Gutiérrez Marina 191153

Grado y grupo

8° "B"

Fecha de entrega

26 de julio de 2021

Para desarrollar este proyecto, nos enfocamos en problemáticas actuales, donde se han encontrado diversos problemas a resolver en la organización de elementos súper importantes, mismos donde se requiere de orden al momento de atender una urgencia, para ello optamos por desarrollar en este caso una Base de Datos enfocada a medicamentos en una farmacia x, esperando así mismo el funcionamiento esperado.

A continuación, se muestra el proceso llevado a cabo para lograr el trabajo:

Código implementado para crear las tablas

```
/* Delimitador cambiado a ; */

/* Conectando a 127.0.0.1 por MariaDB or MySQL (TCP/IP), usuario root, usando contraseña: Yes
... */

SELECT CONNECTION_ID();

/* Juego de caracteres: utf8mb4 */

SHOW /*!50002 GLOBAL */ STATUS;

SELECT NOW();

SHOW VARIABLES;

/* Conectado. ID de Hilo: 22 */

SHOW TABLES FROM `information_schema`;

SHOW DATABASES;

/* Entrando a la sesión "Unnamed-1" */

USE `sys`;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='sys';

SHOW TABLE STATUS FROM `sys`;

SHOW FUNCTION STATUS WHERE `Db`='sys';

SHOW PROCEDURE STATUS WHERE `Db`='sys';

SHOW TRIGGERS FROM `sys`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='sys';
```

```

SHOW COLLATION;

SHOW VARIABLES;

CREATE DATABASE `Bodega` /*!40100 COLLATE 'utf8mb4_general_ci' */;

SHOW DATABASES;

/* Entrando a la sesión "Unnamed-1" */

USE `bodega`;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='bodega';

SHOW TABLE STATUS FROM `bodega`;

SHOW FUNCTION STATUS WHERE `Db`='bodega';

SHOW PROCEDURE STATUS WHERE `Db`='bodega';

SHOW TRIGGERS FROM `bodega`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='bodega';

SHOW ENGINES;

SHOW VARIABLES;

/* #5: Access violation at address 000000000040EF59 in module 'heidisql.exe'. Read of address
0000000000000200 Message CharCode:0 Msg:514 */

/* #64: Abstract Error Message CharCode:0 Msg:514 */

/* #16: Access violation at address 000000000040EF59 in module 'heidisql.exe'. Read of address
FFFFFFFFFFFFFFFF Message CharCode:0 Msg:514 */

/* #44: Abstract Error Message CharCode:0 Msg:514 */

/* #5: Abstract Error Message CharCode:0 Msg:514 */

CREATE TABLE `Administracion de Farmacos` (
    `Farmaco` CHAR(200) NULL DEFAULT NULL,
    `Tipo` CHAR(100) NULL DEFAULT NULL,
    `Tamaño` CHAR(100) NULL DEFAULT NULL,

```

```

        `Unidades` INT(2000) NULL DEFAULT NULL
    )

COMMENT='Dentro de esta base de datos se administran los farmacos de la bodega de un
distribuidor de farmacias.'

COLLATE='utf8mb4_general_ci'

;

/* Error de SQL (1439): Display width out of range for column 'Unidades' (max = 255) */

CREATE TABLE `Administracion de Farmacos` (

    `Farmaco` CHAR(200) NULL DEFAULT NULL,

    `Tipo` CHAR(100) NULL DEFAULT NULL,

    `Tamaño` CHAR(100) NULL DEFAULT NULL,

    `Unidades` INT NULL DEFAULT NULL

)

COMMENT='Dentro de esta base de datos se administran los farmacos de la bodega de un
distribuidor de farmacias.'

COLLATE='utf8mb4_general_ci'

;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='bodega';

SHOW TABLE STATUS FROM `bodega`;

SHOW FUNCTION STATUS WHERE `Db`='bodega';

SHOW PROCEDURE STATUS WHERE `Db`='bodega';

SHOW TRIGGERS FROM `bodega`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='bodega';

/* El nodo de tabla "Administracion de Farmacos" no se encontró en el árbol. */

/* Entrando a la sesión "Unnamed-1" */

```

```

USE `information_schema`;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='information_schema';

SHOW TABLE STATUS FROM `information_schema`;

SHOW FUNCTION STATUS WHERE `Db`='information_schema';

SHOW PROCEDURE STATUS WHERE `Db`='information_schema';

SHOW TRIGGERS FROM `information_schema`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='information_schema';

USE `bodega`;

SELECT * FROM `information_schema`.`COLUMNS` WHERE TABLE_SCHEMA='bodega' AND
TABLE_NAME='administracion de farmacos' ORDER BY ORDINAL_POSITION;

SHOW INDEXES FROM `administracion de farmacos` FROM `bodega`;

SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE
CONSTRAINT_SCHEMA='bodega' AND TABLE_NAME='administracion de farmacos' AND
REFERENCED_TABLE_NAME IS NOT NULL;

SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='bodega'
AND TABLE_NAME='administracion de farmacos' AND REFERENCED_TABLE_NAME IS NOT NULL;

SHOW CREATE TABLE `bodega`.`administracion de farmacos`;

SELECT tc.CONSTRAINT_NAME, cc.CHECK_CLAUSE FROM
`information_schema`.`CHECK_CONSTRAINTS` AS cc,
`information_schema`.`TABLE_CONSTRAINTS` AS tc WHERE tc.CONSTRAINT_SCHEMA='bodega'
AND tc.TABLE_NAME='administracion de farmacos' AND tc.CONSTRAINT_TYPE='CHECK' AND
tc.CONSTRAINT_SCHEMA=cc.CONSTRAINT_SCHEMA AND
tc.CONSTRAINT_NAME=cc.CONSTRAINT_NAME;

COMMENT='Dentro de esta base de datos se administran los farmacos de la bodega de un
distribuidor de farmacias.'

COLLATE='utf8mb4_general_ci'

;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='bodega';

```

```

SHOW TABLE STATUS FROM `bodega`;

SHOW FUNCTION STATUS WHERE `Db`='bodega';

SHOW PROCEDURE STATUS WHERE `Db`='bodega';

SHOW TRIGGERS FROM `bodega`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='bodega';

/* El nodo de tabla "Administracion de Farmacos" no se encontró en el árbol. */

/* Entrando a la sesión "Unnamed-1" */

USE `information_schema`;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='information_schema';

SHOW TABLE STATUS FROM `information_schema`;

SHOW FUNCTION STATUS WHERE `Db`='information_schema';

SHOW PROCEDURE STATUS WHERE `Db`='information_schema';

SHOW TRIGGERS FROM `information_schema`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='information_schema';

USE `bodega`;

SELECT * FROM `information_schema`.`COLUMNS` WHERE TABLE_SCHEMA='bodega' AND
TABLE_NAME='administracion de farmacos' ORDER BY ORDINAL_POSITION;

SHOW INDEXES FROM `administracion de farmacos` FROM `bodega`;

SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE
CONSTRAINT_SCHEMA='bodega' AND TABLE_NAME='administracion de farmacos' AND
REFERENCED_TABLE_NAME IS NOT NULL;

SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='bodega'
AND TABLE_NAME='administracion de farmacos' AND REFERENCED_TABLE_NAME IS NOT NULL;

SHOW CREATE TABLE `bodega`.`administracion de farmacos`;

SELECT tc.CONSTRAINT_NAME, cc.CHECK_CLAUSE FROM
`information_schema`.`CHECK_CONSTRAINTS` AS cc,

```

```
`information_schema`.`TABLE_CONSTRAINTS` AS tc WHERE tc.CONSTRAINT_SCHEMA='bodega'
AND tc.TABLE_NAME='administracion de farmacos' AND tc.CONSTRAINT_TYPE='CHECK' AND
tc.CONSTRAINT_SCHEMA=cc.CONSTRAINT_SCHEMA AND
tc.CONSTRAINT_NAME=cc.CONSTRAINT_NAME;
```

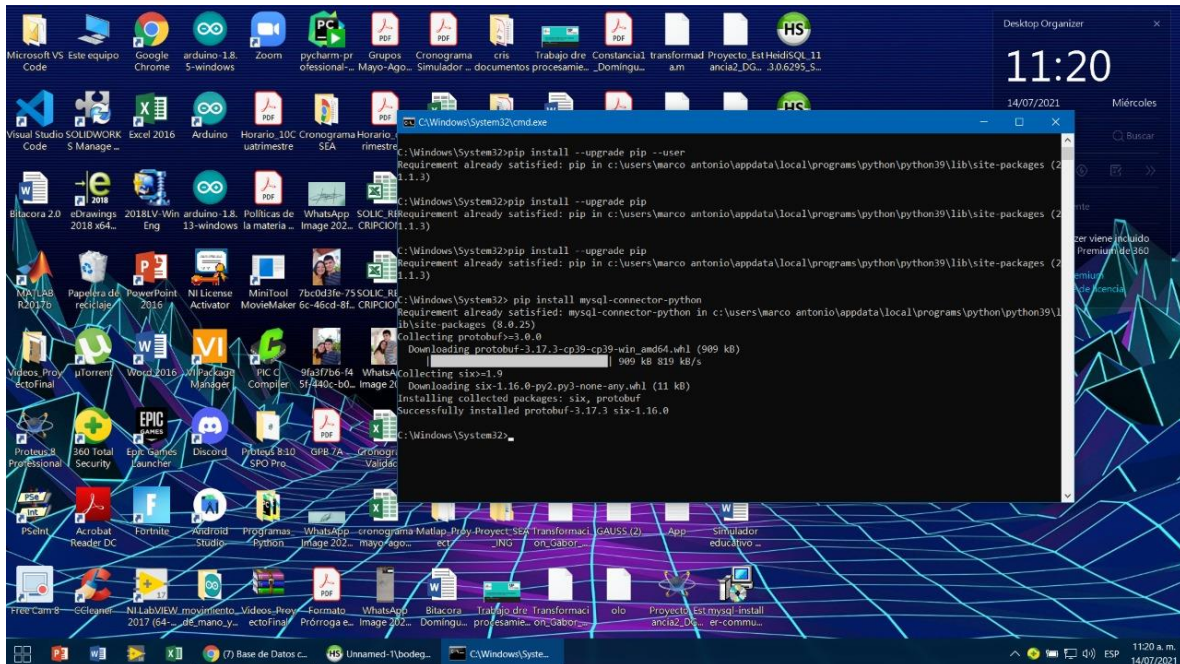


Fig.1 Instalación de paquetes mysql <<Python>>

El server ya se encuentra en una extensión de mysql siendo todo vía intranet, tal como recomendó el docente de la materia.

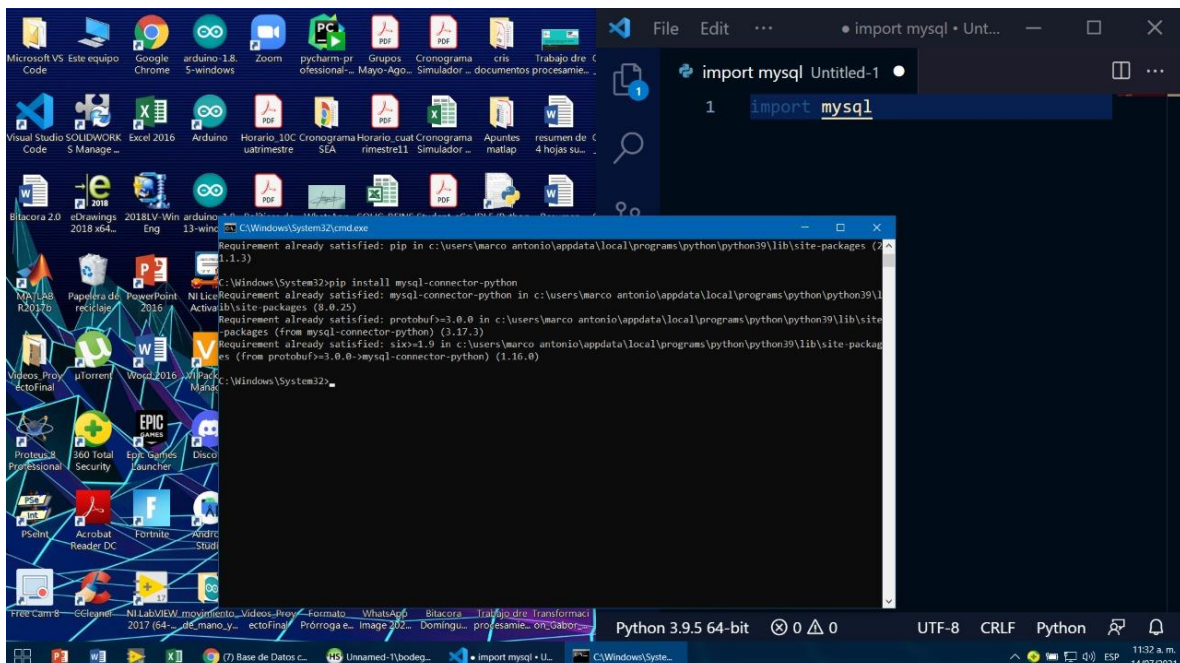
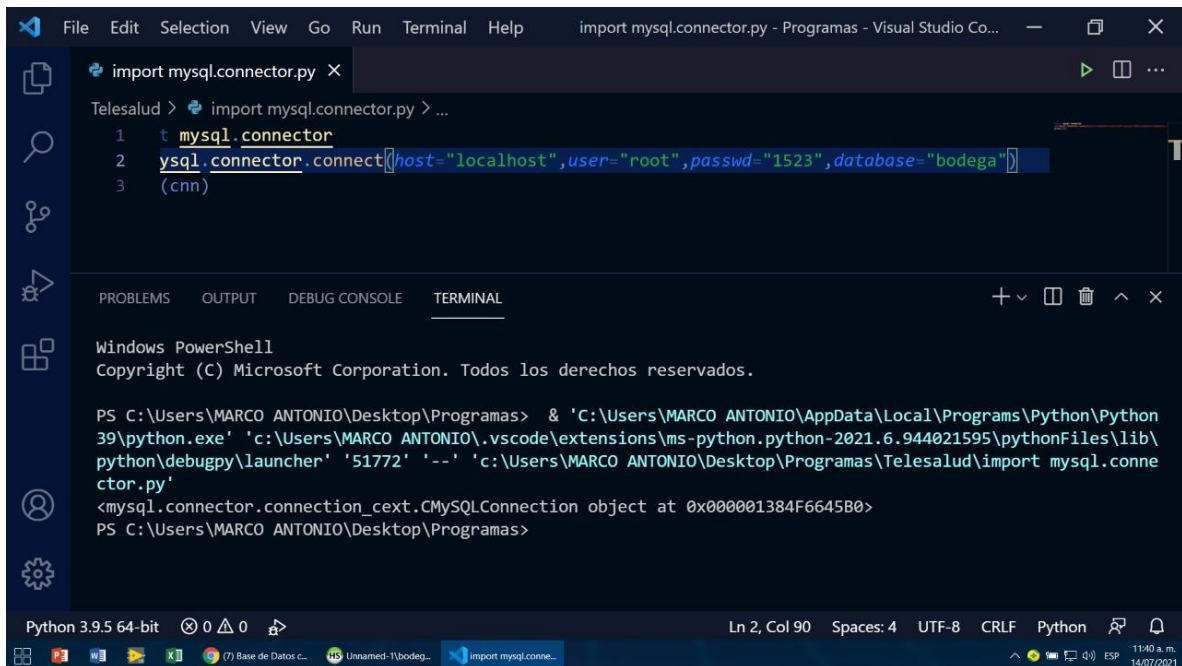


Fig. 2 Conexión Python <<mysqlServer>>



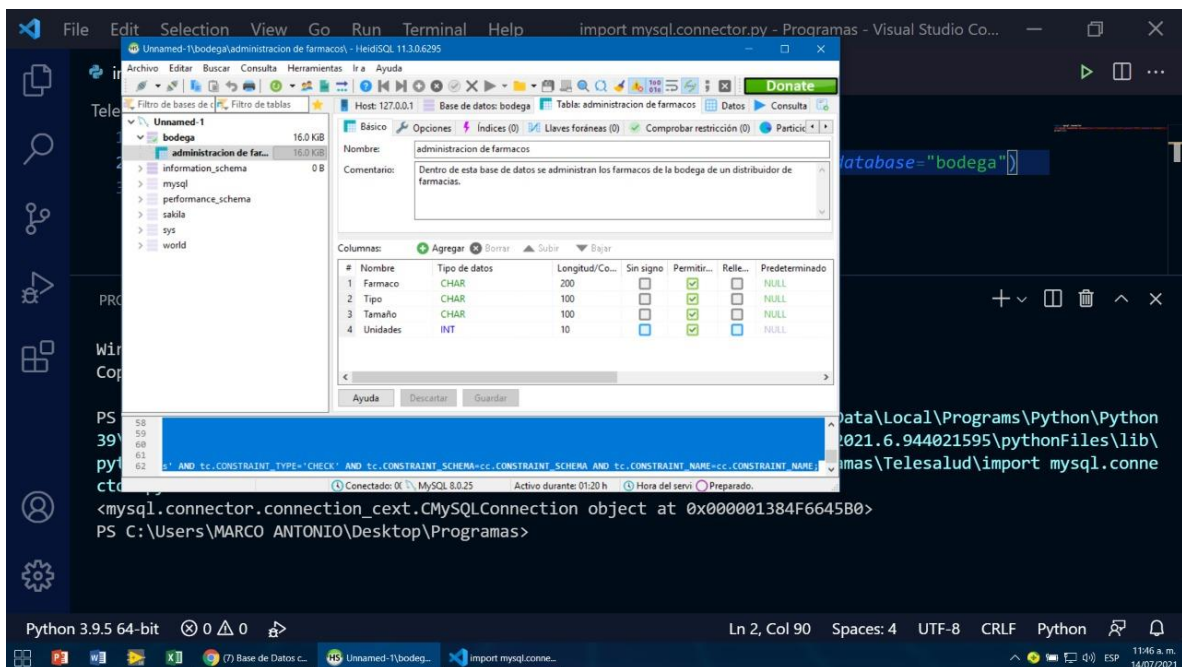
```
import mysql.connector.py x
Telesalud > import mysql.connector.py > ...
1 t mysql.connector
2 ysql.connector.connect(host="localhost",user="root",passwd="1523",database="bodega")
3 (cnn)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

PS C:\Users\MARCO ANTONIO\Desktop\Programas> & 'C:\Users\MARCO ANTONIO\AppData\Local\Programs\Python\Python
39\python.exe' 'c:\Users\MARCO ANTONIO\.vscode\extensions\ms-python.python-2021.6.944021595\pythonFiles\lib\
python\debugpy\launcher' '51772' '--' 'c:\Users\MARCO ANTONIO\Desktop\Programas\Telesalud\import mysql.conne
ctor.py'
<mysql.connector.connection_cext.CMySQLConnection object at 0x000001384F6645B0>
PS C:\Users\MARCO ANTONIO\Desktop\Programas>
```

Fig. 3 Conexión python con la base de datos exitosa.



Unamed-1\bodega\administracion de farmacos - HeidiSQL 11.3.0.6295

Host: 127.0.0.1 Base de datos: bodega Tabla: administracion de farmacos Datos Consultar

Nombre: administracion de farmacos
Comentario: Dentro de esta base de datos se administran los farmacos de la bodega de un distribuidor de farmacias.

#	Nombre	Tipo de datos	Longitud/Co...	Sin signo	Permitir...	Relle...	Predeterminado
1	Farmaco	CHAR	200	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
2	Tipo	CHAR	100	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
3	Tamaño	CHAR	100	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL
4	Unidades	INT	10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL

Conector: Of MySQL 8.0.25 Activo durante: 01:20 h Hora del servi Preparado

```
PS C:\Users\MARCO ANTONIO\Desktop\Programas> & 'C:\Users\MARCO ANTONIO\AppData\Local\Programs\Python\Python
39\python.exe' 'c:\Users\MARCO ANTONIO\.vscode\extensions\ms-python.python-2021.6.944021595\pythonFiles\lib\
python\debugpy\launcher' '51772' '--' 'c:\Users\MARCO ANTONIO\Desktop\Programas\Telesalud\import mysql.conne
ctor.py'
<mysql.connector.connection_cext.CMySQLConnection object at 0x000001384F6645B0>
PS C:\Users\MARCO ANTONIO\Desktop\Programas>
```

Fig.4 Base de datos creada en heidi +mysqlServer

Código implementado par las tablas

```
import mysql.connector
```

```
cnn=mysql.connector.connect(host="localhost",user="root",passwd="1523",database="bodega")
```

```
print(cnn)
```

codigo de python usando idle visual studio code

Código de la base de datos

```
CREATE TABLE `administracion de farmacos` (
```

```
    `Farmaco` CHAR(200) NOT NULL DEFAULT NULL COLLATE 'utf8mb4_general_ci',
```

```
    `Tipo` CHAR(100) NULL DEFAULT NULL COLLATE 'utf8mb4_general_ci',
```

```
    `Tamaño` CHAR(100) NULL DEFAULT NULL COLLATE 'utf8mb4_general_ci',
```

```
    `Unidades` INT(10) NULL DEFAULT NULL,
```

```
    INDEX `Farmaco` (`Farmaco`)
```

```
)
```

```
COMMENT='Dentro de esta base de datos se administran los farmacos de la bodega de un  
distribuidor de farmacias.'
```

```
COLLATE='utf8mb4_general_ci'
```

```
ENGINE=InnoDB
```

```
;
```

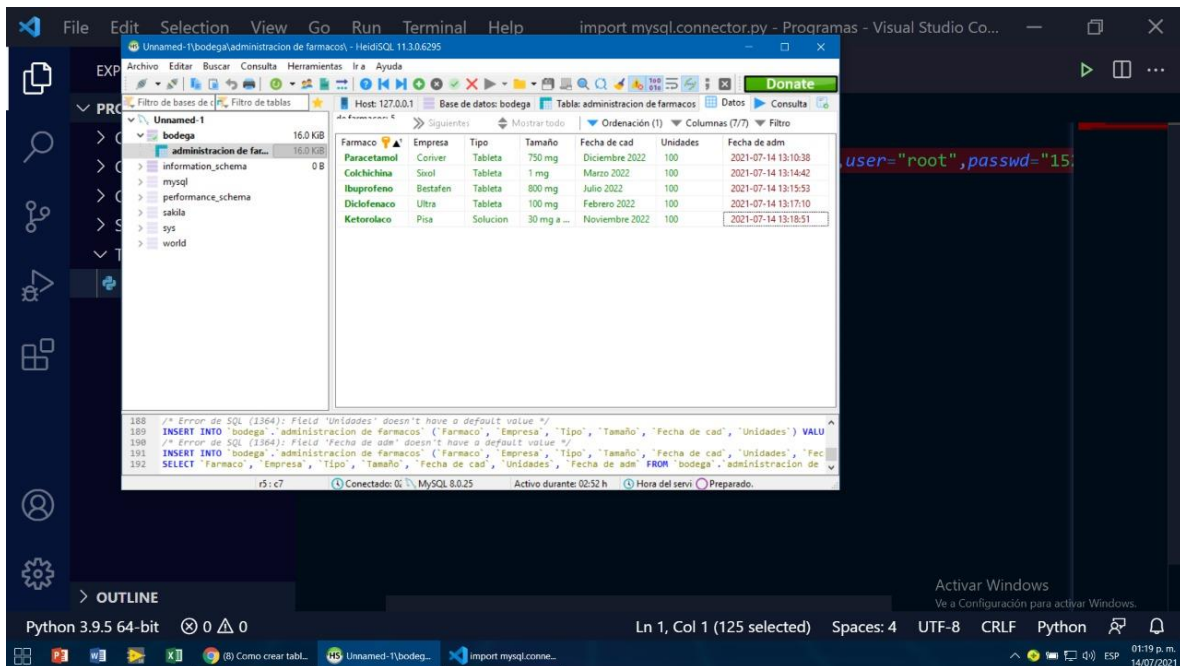


Fig. 5 Nuestra base de datos

Código implementado

```
/* Delimitador cambiado a ; */
```

```
/* Conectando a 127.0.0.1 por MariaDB or MySQL (TCP/IP), usuario root, usando contraseña: Yes ... */
```

```
SELECT CONNECTION_ID();
```

```
/* Juego de caracteres: utf8mb4 */
```

```
SHOW /*!50002 GLOBAL */ STATUS;
```

```
SELECT NOW();
```

```
SHOW VARIABLES;
```

```
/* Conectado. ID de Hilo: 22 */
```

```
SHOW TABLES FROM `information_schema`;
```

```
SHOW DATABASES;
```

```
/* Entrando a la sesión "Unnamed-1" */
```

```
USE `sys`;
```

```
SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='sys';

SHOW TABLE STATUS FROM `sys`;

SHOW FUNCTION STATUS WHERE `Db`='sys';

SHOW PROCEDURE STATUS WHERE `Db`='sys';

SHOW TRIGGERS FROM `sys`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='sys';

SHOW COLLATION;

SHOW VARIABLES;

CREATE DATABASE `Bodega` /*!40100 COLLATE 'utf8mb4_general_ci' */;

SHOW DATABASES;

/* Entrando a la sesión "Unnamed-1" */

USE `bodega`;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='bodega';

SHOW TABLE STATUS FROM `bodega`;

SHOW FUNCTION STATUS WHERE `Db`='bodega';

SHOW PROCEDURE STATUS WHERE `Db`='bodega';

SHOW TRIGGERS FROM `bodega`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='bodega';

SHOW ENGINES;

SHOW VARIABLES;

/* #5: Access violation at address 000000000040EF59 in module 'heidisql.exe'. Read of address
0000000000000200 Message CharCode:0 Msg:514 */

/* #64: Abstract Error Message CharCode:0 Msg:514 */
```

/* #16: Access violation at address 000000000040EF59 in module 'heidisql.exe'. Read of address
FFFFFFFFFFFFFFFF Message CharCode:0 Msg:514 */

/* #44: Abstract Error Message CharCode:0 Msg:514 */

/* #5: Abstract Error Message CharCode:0 Msg:514 */

CREATE TABLE `Administracion de Farmacos` (

 `Farmaco` CHAR(200) NULL DEFAULT NULL,

 `Tipo` CHAR(100) NULL DEFAULT NULL,

 `Tamaño` CHAR(100) NULL DEFAULT NULL,

 `Unidades` INT(2000) NULL DEFAULT NULL

)

COMMENT='Dentro de esta base de datos se administran los farmacos de la bodega de un
distribuidor de farmacias.'

COLLATE='utf8mb4_general_ci'

;

/* Error de SQL (1439): Display width out of range for column 'Unidades' (max = 255) */

CREATE TABLE `Administracion de Farmacos` (

 `Farmaco` CHAR(200) NULL DEFAULT NULL,

 `Tipo` CHAR(100) NULL DEFAULT NULL,

 `Tamaño` CHAR(100) NULL DEFAULT NULL,

 `Unidades` INT NULL DEFAULT NULL

)

COMMENT='Dentro de esta base de datos se administran los farmacos de la bodega de un
distribuidor de farmacias.'

COLLATE='utf8mb4_general_ci'

;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='bodega';

```

SHOW TABLE STATUS FROM `bodega`;

SHOW FUNCTION STATUS WHERE `Db`='bodega';

SHOW PROCEDURE STATUS WHERE `Db`='bodega';

SHOW TRIGGERS FROM `bodega`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='bodega';

/* El nodo de tabla "Administracion de Farmacos" no se encontró en el árbol. */

/* Entrando a la sesión "Unnamed-1" */

USE `information_schema`;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='information_schema';

SHOW TABLE STATUS FROM `information_schema`;

SHOW FUNCTION STATUS WHERE `Db`='information_schema';

SHOW PROCEDURE STATUS WHERE `Db`='information_schema';

SHOW TRIGGERS FROM `information_schema`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='information_schema';

USE `bodega`;

SELECT * FROM `information_schema`.`COLUMNS` WHERE TABLE_SCHEMA='bodega' AND
TABLE_NAME='administracion de farmacos' ORDER BY ORDINAL_POSITION;

SHOW INDEXES FROM `administracion de farmacos` FROM `bodega`;

SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE
CONSTRAINT_SCHEMA='bodega' AND TABLE_NAME='administracion de farmacos' AND
REFERENCED_TABLE_NAME IS NOT NULL;

SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='bodega'
AND TABLE_NAME='administracion de farmacos' AND REFERENCED_TABLE_NAME IS NOT NULL;

SHOW CREATE TABLE `bodega`.`administracion de farmacos`;

SELECT tc.CONSTRAINT_NAME, cc.CHECK_CLAUSE FROM
`information_schema`.`CHECK_CONSTRAINTS` AS cc,

```

```
`information_schema`.`TABLE_CONSTRAINTS` AS tc WHERE tc.CONSTRAINT_SCHEMA='bodega'  
AND tc.TABLE_NAME='administracion de farmacos' AND tc.CONSTRAINT_TYPE='CHECK' AND  
tc.CONSTRAINT_SCHEMA=cc.CONSTRAINT_SCHEMA AND  
tc.CONSTRAINT_NAME=cc.CONSTRAINT_NAME;
```

```
/* #28: Abstract Error Message CharCode:0 Msg:514 */
```

```
/* #36: Abstract Error Message CharCode:0 Msg:514 */
```

```
/* #51: Abstract Error Message CharCode:0 Msg:514 */
```

```
/* #82: Abstract Error Message CharCode:0 Msg:514 */
```

```
ALTER TABLE `administracion de farmacos`
```

```
    CHANGE COLUMN `Farmaco` `Farmaco` VARCHAR(50) NOT NULL DEFAULT '' COLLATE  
'utf8mb4_general_ci' FIRST,
```

```
    CHANGE COLUMN `Tipo` `Tipo` VARCHAR(50) NOT NULL COLLATE 'utf8mb4_general_ci'  
AFTER `Farmaco`,
```

```
    CHANGE COLUMN `Tamaño` `Tamaño` VARCHAR(50) NOT NULL COLLATE  
'utf8mb4_general_ci' AFTER `Tipo`,
```

```
    CHANGE COLUMN `Unidades` `Unidades` INT(10) NOT NULL AFTER `Tamaño`,
```

```
    ADD COLUMN `Fecha` TIMESTAMP NOT NULL AFTER `Unidades`,
```

```
    ADD INDEX `Farmaco` (`Farmaco`);
```

```
SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE  
`SCHEMA_NAME`='bodega';
```

```
SHOW TABLE STATUS FROM `bodega`;
```

```
SHOW FUNCTION STATUS WHERE `Db`='bodega';
```

```
SHOW PROCEDURE STATUS WHERE `Db`='bodega';
```

```
SHOW TRIGGERS FROM `bodega`;
```

```
SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`  
WHERE `EVENT_SCHEMA`='bodega';
```

```
SELECT * FROM `information_schema`.`COLUMNS` WHERE TABLE_SCHEMA='bodega' AND  
TABLE_NAME='administracion de farmacos' ORDER BY ORDINAL_POSITION;
```

```
SHOW INDEXES FROM `administracion de farmacos` FROM `bodega`;
```

```

SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE
CONSTRAINT_SCHEMA='bodega' AND TABLE_NAME='administracion de farmacos' AND
REFERENCED_TABLE_NAME IS NOT NULL;

SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='bodega'
AND TABLE_NAME='administracion de farmacos' AND REFERENCED_TABLE_NAME IS NOT NULL;

/* Entrando a la sesión "Unnamed-1" */

SHOW CREATE TABLE `bodega`.`administracion de farmacos`;

SELECT tc.CONSTRAINT_NAME, cc.CHECK_CLAUSE FROM
`information_schema`.`CHECK_CONSTRAINTS` AS cc,
`information_schema`.`TABLE_CONSTRAINTS` AS tc WHERE tc.CONSTRAINT_SCHEMA='bodega'
AND tc.TABLE_NAME='administracion de farmacos' AND tc.CONSTRAINT_TYPE='CHECK' AND
tc.CONSTRAINT_SCHEMA=cc.CONSTRAINT_SCHEMA AND
tc.CONSTRAINT_NAME=cc.CONSTRAINT_NAME;

SELECT * FROM `bodega`.`administracion de farmacos` LIMIT 1000;

SELECT * FROM `bodega`.`administracion de farmacos` ORDER BY `Farmaco` ASC LIMIT 1000;

SHOW VARIABLES;

SHOW GLOBAL VARIABLES;

SHOW /*!50002 GLOBAL */ STATUS;

SELECT * FROM `information_schema`.`COLUMNS` WHERE TABLE_SCHEMA='information_schema'
AND TABLE_NAME='PROCESSLIST' ORDER BY ORDINAL_POSITION;

SELECT `ID`, `USER`, `HOST`, `DB`, `COMMAND`, `TIME`, `STATE`, LEFT(`INFO`, 51200) AS `Info`
FROM `information_schema`.`PROCESSLIST`;

SHOW /*!50002 GLOBAL */ STATUS LIKE 'Com\_%';

SHOW CREATE TABLE `bodega`.`administracion de farmacos`;

SELECT * FROM `bodega`.`administracion de farmacos` ORDER BY `Farmaco` ASC LIMIT 1000;

/* #5: Abstract Error Message CharCode:0 Msg:514 */

/* #76: Access violation at address 00000000011B1410 in module 'heidisql.exe'. Execution of
address 00000000011B1410 Message CharCode:27 Msg:256 */

/* #8: Abstract Error Message CharCode:0 Msg:514 */

/* #74: Abstract Error Message CharCode:0 Msg:514 */

```

/* #35: Abstract Error Message CharCode:0 Msg:514 */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Tipo`, `Tamaño`) VALUES ('Paracetamol', 'Tableta', '750 mg');
```

/* Error de SQL (1364): Field 'Unidades' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Tipo`, `Tamaño`) VALUES ('Paracetamol', 'Tableta', '750 mg');
```

/* Error de SQL (1364): Field 'Unidades' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Tipo`, `Tamaño`) VALUES ('Paracetamol', 'Tableta', '750 mg');
```

/* Error de SQL (1364): Field 'Unidades' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Tipo`) VALUES ('Paracetamol', 'Tableta');
```

/* Error de SQL (1364): Field 'Tamaño' doesn't have a default value */

/* #25: Abstract Error Message CharCode:0 Msg:514 */

/* #98: Abstract Error Message CharCode:0 Msg:514 */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Tipo`) VALUES ('Paracetamol', 'Tableta');
```

/* Error de SQL (1364): Field 'Tamaño' doesn't have a default value */

/* #1634496361: Access violation at address 0000001000000000 in module 'heidisql.exe'.
Execution of address 0000001000000000 Message CharCode:13 Msg:256 */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Tipo`, `Tamaño`) VALUES ('Paracetamol', 'Tableta', '750 mg');
```

/* Error de SQL (1364): Field 'Unidades' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Tipo`, `Tamaño`, `Unidades`) VALUES ('Paracetamol', 'Tableta', '750 mg', '100');
```

/* Error de SQL (1364): Field 'Fecha' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Tipo`, `Tamaño`, `Unidades`, `Fecha`) VALUES ('Paracetamol', 'Tableta', '750 mg', '100', '2021-07-14 13:10:38');
```



```
SELECT `Farmaco`, `Tipo`, `Tamaño`, `Unidades`, `Fecha` FROM `bodega`.`administracion de
farmacos` WHERE `Farmaco`='Paracetamol' AND `Tipo`='Tableta' AND `Tamaño`='750 mg' AND
`Unidades`=100 AND `Fecha`='2021-07-14 13:10:38' LIMIT 1;
```

```
/* #16: Abstract Error Message CharCode:0 Msg:514 */
```

```
/* #36: Abstract Error Message CharCode:0 Msg:514 */
```

```
/* #17: Abstract Error Message CharCode:0 Msg:514 */
```

```
ALTER TABLE `administracion de farmacos`
```

```
    ADD COLUMN `Empresa` VARCHAR(50) NOT NULL DEFAULT '' AFTER `Farmaco`,
```

```
    CHANGE COLUMN `Tamaño` `Tamaño` CHAR(50) NOT NULL COLLATE 'utf8mb4_general_ci'
AFTER `Tipo`,
```

```
    ADD COLUMN `F
```

```
echa de cad` VARCHAR(50) NOT NULL AFTER `Tamaño`,
```

```
    CHANGE COLUMN `Unidades` `Unidades` VARCHAR(10) NOT NULL AFTER `F
```

```
echa de cad`,
```

```
    CHANGE COLUMN `Fecha` `Fecha de adm` TIMESTAMP NOT NULL AFTER `Unidades`,
```

```
    DROP INDEX `Farmaco`,
```

```
    ADD PRIMARY KEY (`Farmaco`);
```

```
SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='bodega';
```

```
SHOW TABLE STATUS FROM `bodega`;
```

```
SHOW FUNCTION STATUS WHERE `Db`='bodega';
```

```
SHOW PROCEDURE STATUS WHERE `Db`='bodega';
```

```
SHOW TRIGGERS FROM `bodega`;
```

```
SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='bodega';
```

```
SELECT * FROM `information_schema`.`COLUMNS` WHERE TABLE_SCHEMA='bodega' AND
TABLE_NAME='administracion de farmacos' ORDER BY ORDINAL_POSITION;
```

```
SHOW INDEXES FROM `administracion de farmacos` FROM `bodega`;
```

```

SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE
CONSTRAINT_SCHEMA='bodega' AND TABLE_NAME='administracion de farmacos' AND
REFERENCED_TABLE_NAME IS NOT NULL;

SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='bodega'
AND TABLE_NAME='administracion de farmacos' AND REFERENCED_TABLE_NAME IS NOT NULL;

/* Entrando a la sesión "Unnamed-1" */

SHOW CREATE TABLE `bodega`.`administracion de farmacos`;

SELECT tc.CONSTRAINT_NAME, cc.CHECK_CLAUSE FROM
`information_schema`.`CHECK_CONSTRAINTS` AS cc,
`information_schema`.`TABLE_CONSTRAINTS` AS tc WHERE tc.CONSTRAINT_SCHEMA='bodega'
AND tc.TABLE_NAME='administracion de farmacos' AND tc.CONSTRAINT_TYPE='CHECK' AND
tc.CONSTRAINT_SCHEMA=cc.CONSTRAINT_SCHEMA AND
tc.CONSTRAINT_NAME=cc.CONSTRAINT_NAME;

SELECT * FROM `bodega`.`administracion de farmacos` ORDER BY `Farmaco` ASC LIMIT 1000;

INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`) VALUES ('Colchichina');

/* Error de SQL (1364): Field 'Tipo' doesn't have a default value */

INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`) VALUES ('Colchichina');

/* Error de SQL (1364): Field 'Tipo' doesn't have a default value */

INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`) VALUES ('Colchichina');

/* Error de SQL (1364): Field 'Tipo' doesn't have a default value */

SELECT * FROM `bodega`.`administracion de farmacos` ORDER BY `Farmaco` ASC, `F
echa de cad` ASC LIMIT 1000;

ALTER TABLE `administracion de farmacos`

CHANGE COLUMN `F
echa de cad` `Fecha de cad` VARCHAR(50) NOT NULL COLLATE 'utf8mb4_general_ci' AFTER
`Tamaño`;

SELECT `DEFAULT_COLLATION_NAME` FROM `information_schema`.`SCHEMATA` WHERE
`SCHEMA_NAME`='bodega';

SHOW TABLE STATUS FROM `bodega`;

```

```

SHOW FUNCTION STATUS WHERE `Db`='bodega';

SHOW PROCEDURE STATUS WHERE `Db`='bodega';

SHOW TRIGGERS FROM `bodega`;

SELECT *, EVENT_SCHEMA AS `Db`, EVENT_NAME AS `Name` FROM information_schema.`EVENTS`
WHERE `EVENT_SCHEMA`='bodega';

SELECT * FROM `information_schema`.`COLUMNS` WHERE TABLE_SCHEMA='bodega' AND
TABLE_NAME='administracion de farmacos' ORDER BY ORDINAL_POSITION;

SHOW INDEXES FROM `administracion de farmacos` FROM `bodega`;

SELECT * FROM information_schema.REFERENTIAL_CONSTRAINTS WHERE
CONSTRAINT_SCHEMA='bodega' AND TABLE_NAME='administracion de farmacos' AND
REFERENCED_TABLE_NAME IS NOT NULL;

SELECT * FROM information_schema.KEY_COLUMN_USAGE WHERE TABLE_SCHEMA='bodega'
AND TABLE_NAME='administracion de farmacos' AND REFERENCED_TABLE_NAME IS NOT NULL;

/* Entrando a la sesión "Unnamed-1" */

SHOW CREATE TABLE `bodega`.`administracion de farmacos`;

SELECT tc.CONSTRAINT_NAME, cc.CHECK_CLAUSE FROM
`information_schema`.`CHECK_CONSTRAINTS` AS cc,
`information_schema`.`TABLE_CONSTRAINTS` AS tc WHERE tc.CONSTRAINT_SCHEMA='bodega'
AND tc.TABLE_NAME='administracion de farmacos' AND tc.CONSTRAINT_TYPE='CHECK' AND
tc.CONSTRAINT_SCHEMA=cc.CONSTRAINT_SCHEMA AND
tc.CONSTRAINT_NAME=cc.CONSTRAINT_NAME;

SELECT * FROM `bodega`.`administracion de farmacos` ORDER BY `Farmaco` ASC LIMIT 1000;

UPDATE `bodega`.`administracion de farmacos` SET `Empresa`='Coriver', `Fecha de cad`='dic 2022'
WHERE `Farmaco`='Paracetamol';

SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM
`bodega`.`administracion de farmacos` WHERE `Farmaco`='Paracetamol';

INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`)
VALUES ('Colchichina', 'Sixol', 'Tableta', '1 mg');

/* Error de SQL (1364): Field 'Fecha de cad' doesn't have a default value */

```

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`,  
`Fecha de cad`, `Unidades`, `Fecha de adm`) VALUES ('Colchichina', 'Sixol', 'Tableta', '1 mg', 'marzo  
2022', '100', '2021-07-14 13:14:42');
```

```
SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM  
`bodega`.`administracion de farmacos` WHERE `Farmaco`='Colchichina';
```

```
UPDATE `bodega`.`administracion de farmacos` SET `Fecha de cad`='diciembre 2022' WHERE  
`Farmaco`='Paracetamol';
```

```
SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM  
`bodega`.`administracion de farmacos` WHERE `Farmaco`='Paracetamol';
```

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`) VALUES ('Ibuprofeno',  
'Bestafen');
```

```
/* Error de SQL (1364): Field 'Tipo' doesn't have a default value */
```

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`,  
`Fecha de cad`) VALUES ('Ibuprofeno', 'Bestafen', 'Tableta', '800 mg', 'Julio 2021');
```

```
/* Error de SQL (1364): Field 'Unidades' doesn't have a default value */
```

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`,  
`Fecha de cad`, `Unidades`, `Fecha de adm`) VALUES ('Ibuprofeno', 'Bestafen', 'Tableta', '800 mg',  
'Julio 2021', '100', '2021-07-14 13:15:53');
```

```
SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM  
`bodega`.`administracion de farmacos` WHERE `Farmaco`='Ibuprofeno';
```

```
UPDATE `bodega`.`administracion de farmacos` SET `Fecha de cad`='Marzo 2022' WHERE  
`Farmaco`='Colchichina';
```

```
SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM  
`bodega`.`administracion de farmacos` WHERE `Farmaco`='Colchichina';
```

```
UPDATE `bodega`.`administracion de farmacos` SET `Fecha de cad`='Diciembre 2022' WHERE  
`Farmaco`='Paracetamol';
```

```
SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM  
`bodega`.`administracion de farmacos` WHERE `Farmaco`='Paracetamol';
```

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`,  
`Fecha de cad`) VALUES ('Diclofenaco', 'Ultra', 'Tableta', '100 mg', 'Febrero 2022');
```

```
/* Error de SQL (1364): Field 'Unidades' doesn't have a default value */
```

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`,  
`Fecha de cad`, `Unidades`, `Fecha de adm`) VALUES ('Diclofenaco', 'Ultra', 'Tableta', '100 mg',  
'Febrero 2022', '100', '2021-07-14 13:17:10');
```

```
SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM  
`bodega`.`administracion de farmacos` WHERE `Farmaco`='Diclofenaco';
```

```
UPDATE `bodega`.`administracion de farmacos` SET `Fecha de cad`='Julio 2022' WHERE  
`Farmaco`='Ibuprofeno';
```

```
SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM  
`bodega`.`administracion de farmacos` WHERE `Farmaco`='Ibuprofeno';
```

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`) VALUES ('Ketorolaco');
```

/* Error de SQL (1364): Field 'Tipo' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`)  
VALUES ('Ketorolaco', 'Pisa', 'Solucion', '30 mg/ 1ml');
```

/* Error de SQL (1364): Field 'Fecha de cad' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`)  
VALUES ('Ketorolaco', 'Pisa', 'Solucion', '30 mg a 1 ml');
```

/* Error de SQL (1364): Field 'Fecha de cad' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`)  
VALUES ('Ketorolaco', 'Pisa', 'Solucion', '1 ml');
```

/* Error de SQL (1364): Field 'Fecha de cad' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`,  
`Fecha de cad`) VALUES ('Ketorolaco', 'Pisa', 'Solucion', '30 mg a 1 ml', 'Noviembre 2022');
```

/* Error de SQL (1364): Field 'Unidades' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`,  
`Fecha de cad`, `Unidades`) VALUES ('Ketorolaco', 'Pisa', 'Solucion', '30 mg a 1 ml', 'Noviembre  
2022', '100');
```

/* Error de SQL (1364): Field 'Fecha de adm' doesn't have a default value */

```
INSERT INTO `bodega`.`administracion de farmacos` (`Farmaco`, `Empresa`, `Tipo`, `Tamaño`,  
`Fecha de cad`, `Unidades`, `Fecha de adm`) VALUES ('Ketorolaco', 'Pisa', 'Solucion', '30 mg a 1 ml',  
'Noviembre 2022', '100', '2021-07-14 13:18:51');
```

`SELECT `Farmaco`, `Empresa`, `Tipo`, `Tamaño`, `Fecha de cad`, `Unidades`, `Fecha de adm` FROM `bodega`.`administracion de farmacos` WHERE `Farmaco`='Ketorolaco';`

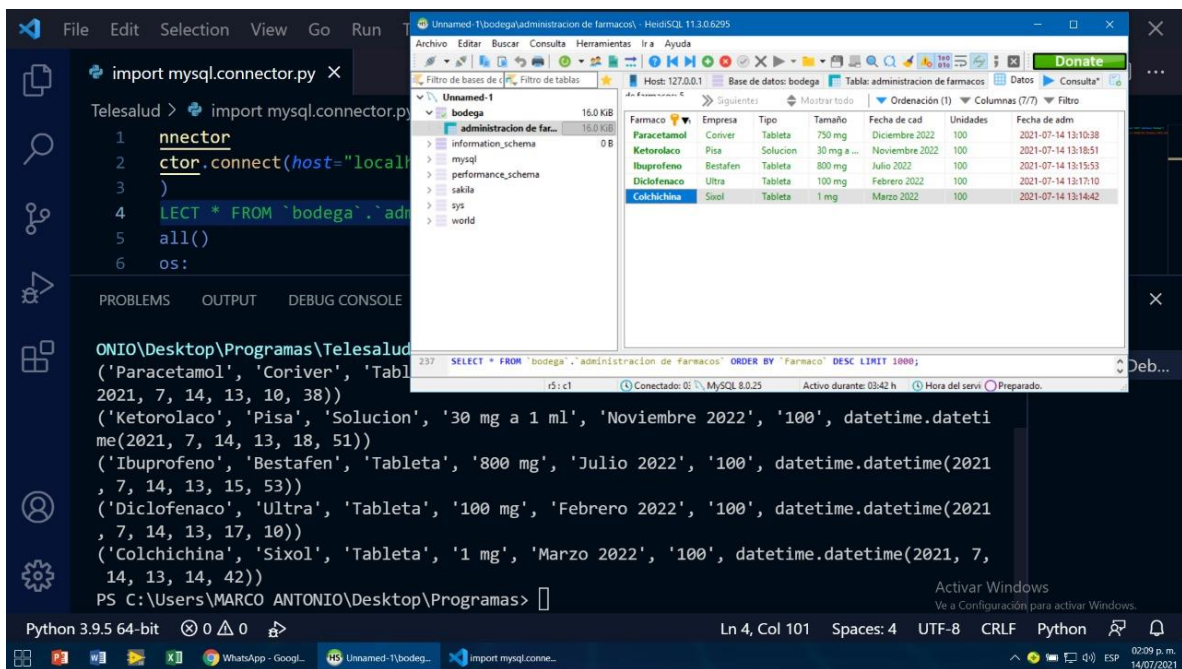
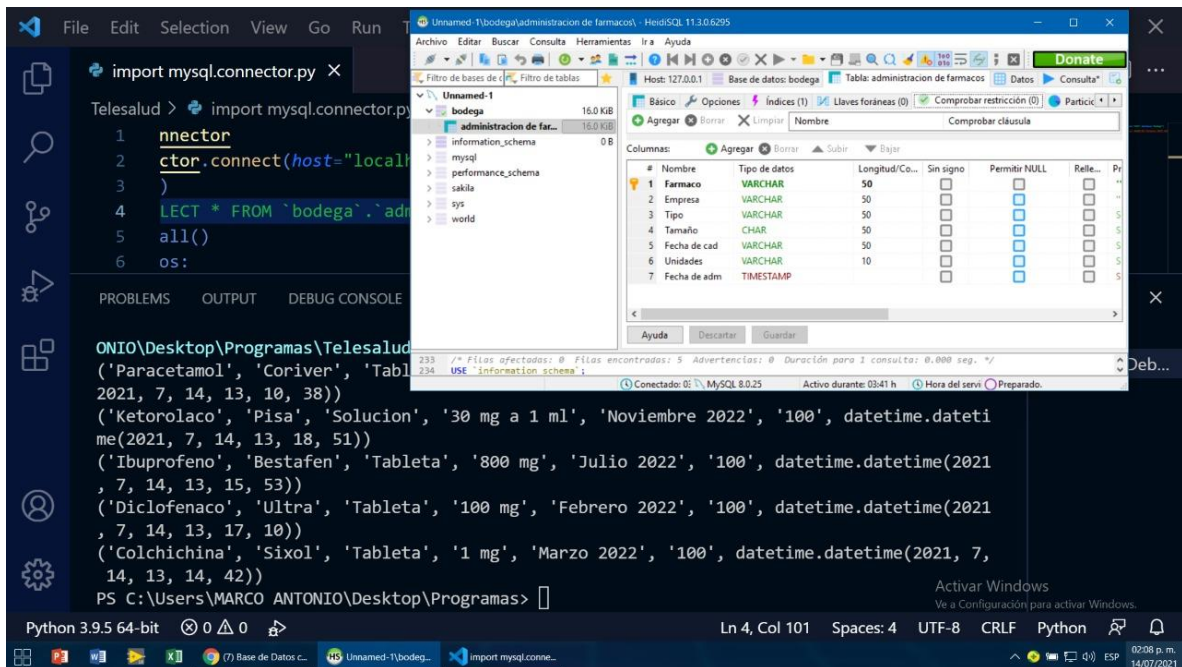


Fig. 6 Impresión de la base de datos desde Python.

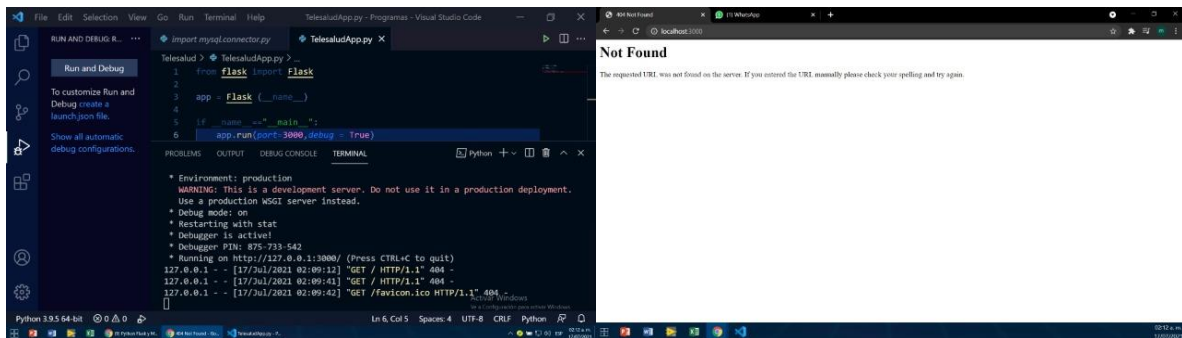


Fig. 7 Código de ubicación en la web creada al localhost:3000 del server local vía intranet, el puerto 3000 es el que estoy utilizando

Allí mismo se puede observar que en Python marca quien entro y la hora mediante la ip local, siendo esto el famoso manejo de tráfico en la web ya que se puede gestionar quien está accediendo a la web mediante su ip.

El aviso 404 no es un error es un mensaje default que Python y flask designan ya que no hay nada en la pagina

En este caso se cargó un mensaje por html, para que en vez de mostrar el default 404 nos muestre un mensaje que diga algo "Si podemos futuros Ing".

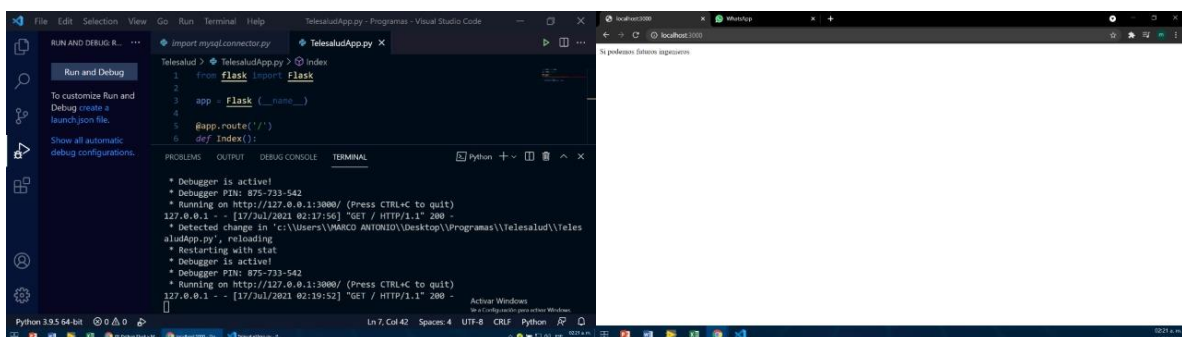
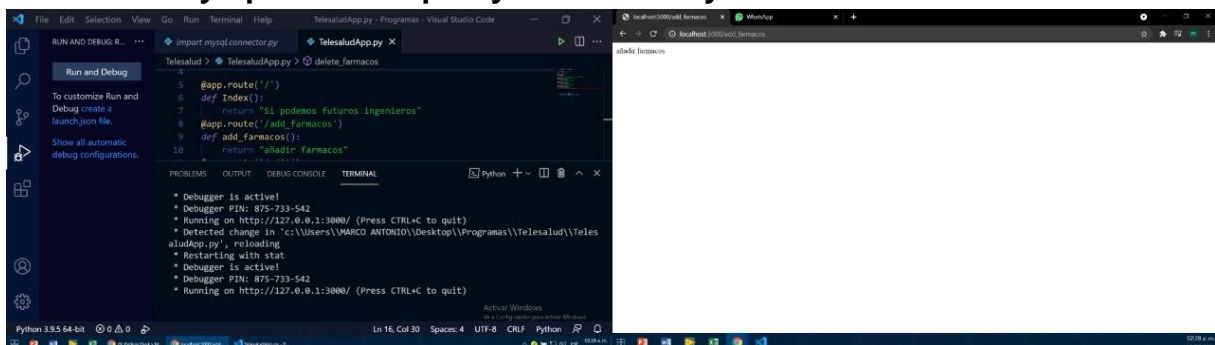


Fig. 8 Por lo tanto a partir de aquí es editar y crear rutas, para una posterior conexión mysql misma que ya tenemos y es cuestión de transcribir.



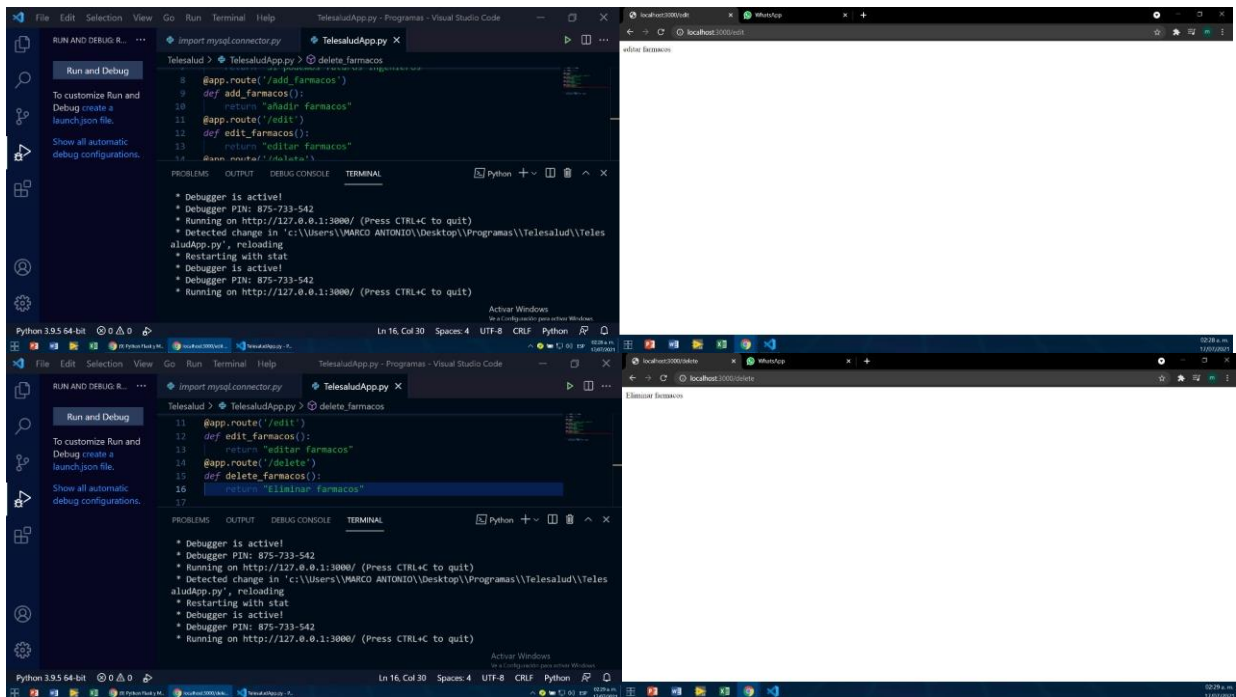


Fig. 9 Creación de distintas rutas que nos ayudaran a manejar formularios de eliminar, editar y añadir información a la base de datos adjunto cada ruta.

¿Cómo se pueden identificar las rutas?

Se identifican mediante la fórmula *direccion+/⁺nuevadireccion*

Sea *localhost:3000/edit* por ejemplo

Se pone eslasj *"/*+ el nombre de la ruta de la dirección de una nueva página en pocas palabras cada que cambiamos de página es una ruta distinta.

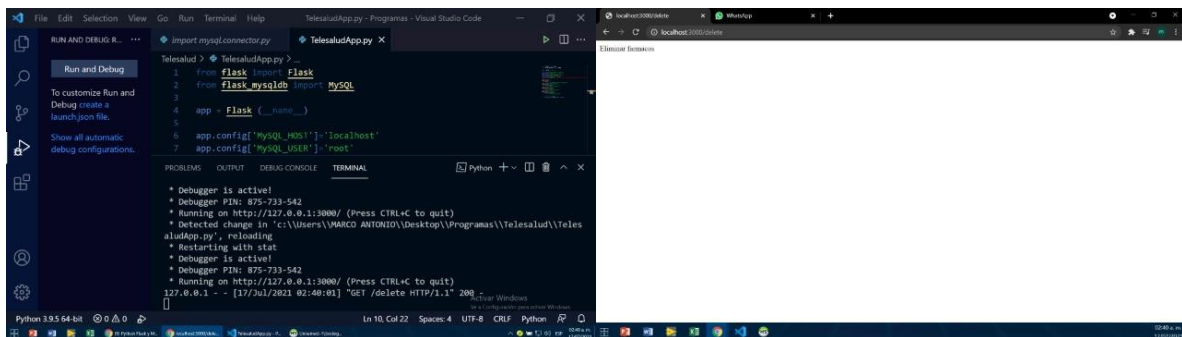


Fig.10 Conexión de la web a mysql exitosa

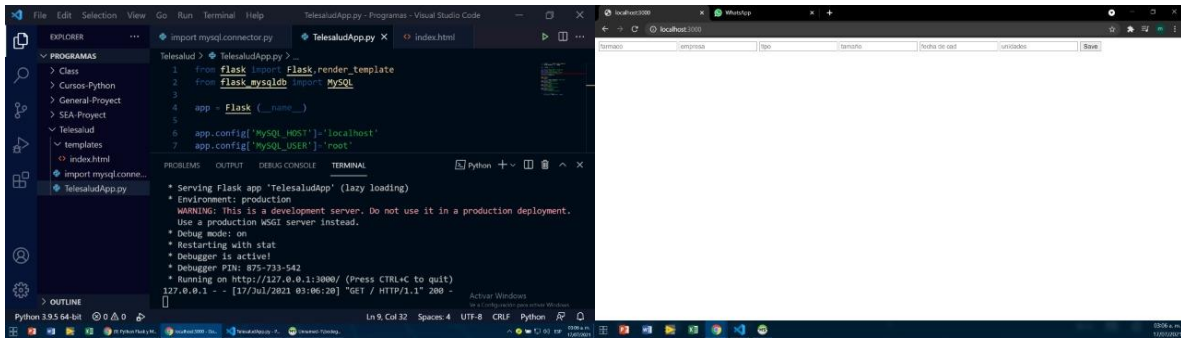


Fig.11 Ya se ha colocado un formulario sencillo de nuestra base de datos.

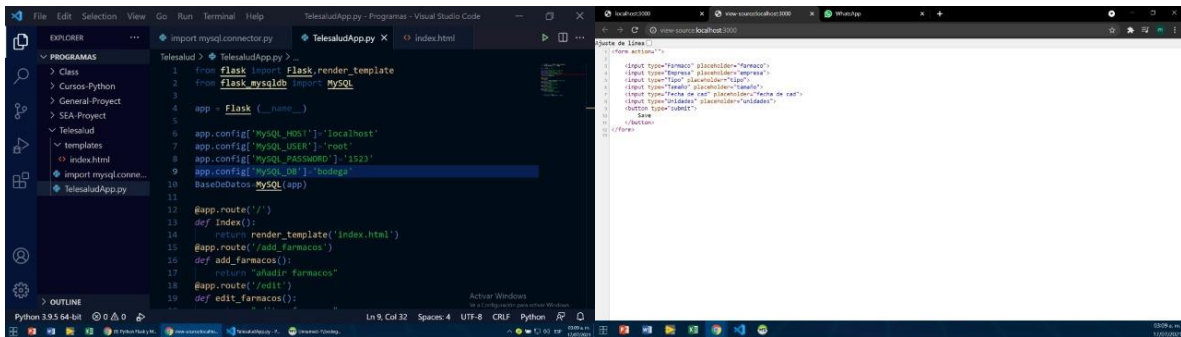


Fig.12 El código fuente de la página funciona adecuadamente.

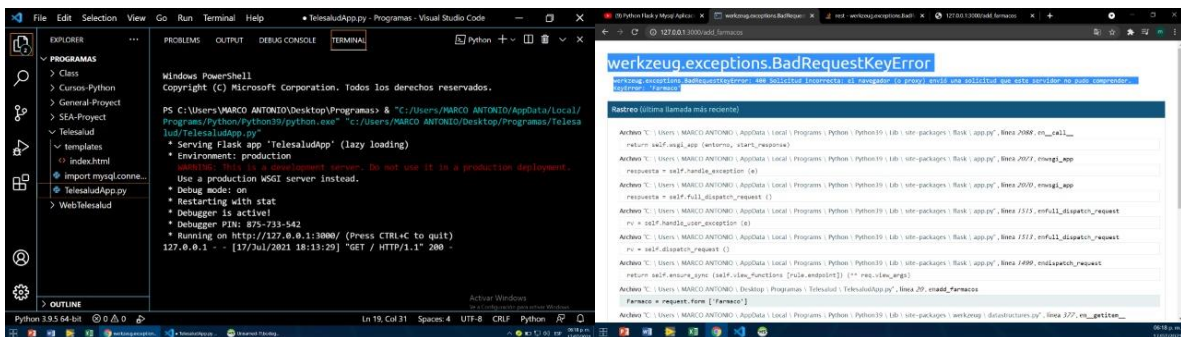


Fig.13 Tuvimos un pequeño percance en HTML.

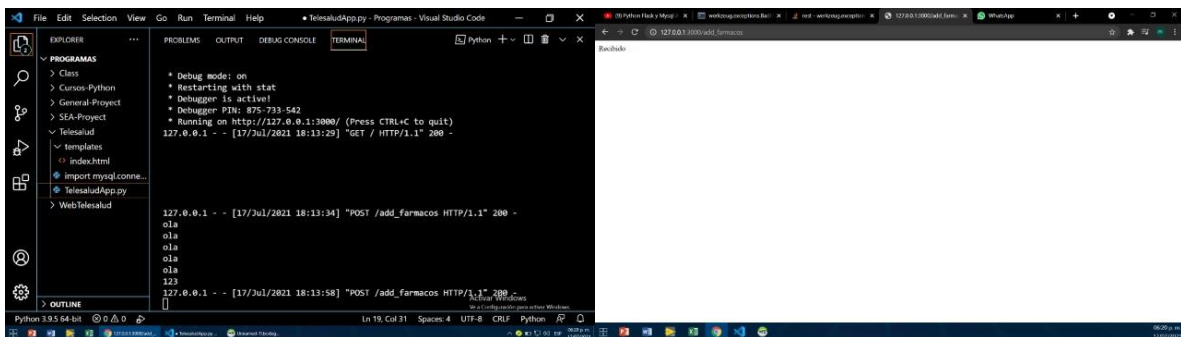


Fig.14 Mismo que logro solucionarse.

```
telesalud (
    frasco ,
    g ,
    redirección ,
    render_template ,
    solicitud ,
    sesión ,
    url_for
)
```

usuario de clase :

```
def __init__ ( yo , id , nombre de usuario , contraseña ):
```

 yo . id = id

 yo . nombre de usuario = nombre de usuario

 yo . contraseña = contraseña

```
def __repr__ ( yo ):
```

 return f '<Usuario: { self . nombre de usuario } > '

```
usuarios = []
```

```
usuarios . añadir ( Usuario ( id = 1 , nombre de usuario = 'farmacia' , contraseña = '12345' ))
```

```
aplicación = Frasco ( _nombre_ )
```

```
aplicación . clave_secreta = 'alguna clave secreta que solo puede saber'
```

```
@ aplicación . before_request
```

```
def before_request ():
```

```
    g . usuario = Ninguno
```

```
    si 'user_id' en la sesión :
```

```
        usuario = [ x para x en usuarios si x . id == sesión [ 'user_id' ] ] [ 0 ]
```

```
        g . usuario = usuario
```

```
@ aplicación . route ( '/ login' , métodos = [ 'GET' , 'POST' ] )
```

```
def login ():
```

```
    si lo solicita . método == 'POST' :
```

```
        Sesión . pop ( 'user_id' , Ninguno )
```

```
        nombre de usuario = solicitud . formulario [ 'nombre de usuario' ]
```

```
        contraseña = solicitud . formulario [ 'contraseña' ]
```

```
        usuario = [ x para x en usuarios si x . nombre de usuario == nombre de  
usuario ] [ 0 ]
```

```
        si usuario y usuario . contraseña == contraseña :
```

```
            sesión [ 'user_id' ] = usuario . identificación
```

```
            return redirect ( url_for ( 'perfil' ) )
```

```
return redirect ( url_for ( 'iniciar sesión' ))
```

```
return render_template ( 'login.html' )
```

```
@ aplicación . ruta ( '/' perfil' )
```

```
def perfil ():
```

```
    si no g . usuario :
```

```
        return redirect ( url_for ( 'iniciar sesión' ))
```

```
    return render_template ( 'perfil.html' )
```

Código para ventana de acción, donde se ejecutan las ventanas y métodos.

```
from tkinter import *
```

```
from ventana import *
```

```
def main():
```

```
    root = Tk()
```

```
    root.wm_title("Crud Python MySQL")
```

```
    app = Ventana(root)
```

```
    app.mainloop()
```

```
if __name__ == "__main__":
```

```
main()
```

Código para la ventana, interfaz y métodos.

```
from tkinter import *
from tkinter import ttk
from countries import Countries
from tkinter import messagebox

class Ventana(Frame):

    paises = Countries()

    def __init__(self, master=None):
        super().__init__(master,width=1460, height=400,bg="gray9")
        self.master = master
        self.pack()
        self.create_widgets()
        self.llenaDatos()
        self.habilitarCajas("disable")
        self.habilitarBtnOper("normal")
        self.habilitarBtnGuardar("disable")
        self.id=-1

    def habilitarCajas(self,estado):
        self.txtNombre.configure(state=estado)
```

```
self.txtTipo.configure(state=estado)

self.txtCantidad.configure(state=estado)

self.txtUnidades.configure(state=estado)


def habilitarBtnOper(self,estado):

    self.btnNuevo.configure(state=estado)

    self.btnModificar.configure(state=estado)

    self.btnEliminar.configure(state=estado)


def habilitarBtnGuardar(self,estado):

    self.btnGuardar.configure(state=estado)

    self.btnCancelar.configure(state=estado)


def limpiarCajas(self):

    self.txtNombre.delete(0,END)

    self.txtTipo.delete(0,END)

    self.txtCantidad.delete(0,END)

    self.txtUnidades.delete(0,END)


def limpiaGrid(self):

    for item in self.grid.get_children():

        self.grid.delete(item)


def llenaDatos(self):

    datos = self.países.consulta_países()
```

```
for row in datos:

    self.grid.insert("",END,text=row[0], values=(row[1],row[2],row[3],row[4]))

if len(self.grid.get_children())>0:

    self.grid.selection_set(self.grid.get_children()[0])


def fNuevo(self):

    self.habilitarCajas("normal")

    self.habilitarBtnOper("disable")

    self.habilitarBtnGuardar("normal")

    self.limpiarCajas()

    self.txtNombre.focus()


def fGuardar(self):

    if self.id== -1:

        self.paises.inserta_pais(self.txtNombre.get(),self.txtTipo.get(),self.txtCantidad.get(),
        self.txtUnidades.get())

        messagebox.showinfo("Insertar",message="Elemento insertado
correctamente")

    else:

        self.paises.modifica_pais(self.id,self.txtNombre.get(),self.txtTipo.get(),self.txtCantid
ad.get(),self.txtUnidades.get())

        self.id=-1

        messagebox.showinfo("Insertar",message="Elemento insertado
correctamente")
```

```
self.limpiaGrid()
self.llenaDatos()
self.limpiarCajas()
self.habilitarBtnGuardar("disable")
self.habilitarBtnOper("normal")
self.habilitarCajas("disable")
```

```
def fModificar(self):
    selected = self.grid.focus()
    clave = self.grid.item(selected, "text")
    if clave=="":
        messagebox.showerror("ALERTA",message="debes seleccionar un
elemento")
    else:
        self.id=clave
        self.habilitarCajas("normal")
        valores=self.grid.item(selected, "values")
        self.limpiarCajas()
        self.txtNombre.insert(0,valores[0])
        self.txtTipo.insert(0,valores[1])
        self.txtCantidad.insert(0,valores[2])
        self.txtUnidades.insert(0,valores[3])
        self.habilitarBtnOper("disable")
        self.habilitarBtnGuardar("normal")
        self.txtNombre.focus()
```



```

def fEliminar(self):

    selected = self.grid.focus()

    clave = self.grid.item(selected,"text")

    if clave=="":

        messagebox.showerror("ALERTA",message="debes seleccionar un
elemento")

    else:

        print(clave)

        valores = self.grid.item(selected,"values")

        data=str(clave) + ","+ valores[0] + ","+ valores[1]

        r= messagebox.askquestion("ELIMINAR",message="¿Deseas eliminar el
registro seleccionado?\n" + data)

        if r == messagebox.YES:

            n = self.países.elimina_pais(clave)

            if n==1:

                messagebox.showinfo("Eliminar",message="Elemento eliminado
correctamente")

                self.limpiaGrid()

                self.llenaDatos()

            else:

                messagebox.showinfo("Eliminar",message="No fue posible eliminar el
elemento")

        else:

            pass

```

```

def fCancelar(self):

    m = messagebox.askquestion("Cancelar",message="Estas seguro que
deseas cancelar la operación")

    if m == messagebox.YES:

        self.limpiarCajas()

        self.habilitarBtnGuardar("disable")

        self.habilitarBtnOper("normal")

        self.habilitarCajas("disable")

    else:

        pass


def create_widgets(self):

    frame1 = Frame(self, bg="turquoise4")

    frame1.place(x=0,y=0,width=100, height=350)


    self.btnNuevo=Button(frame1,text="New farmaco", command=self.fNuevo,
bg="gray9", fg="white")

    self.btnNuevo.place(x=5,y=50,width=90, height=30 )


    self.btnModificar=Button(frame1,text="Edit farmaco",
command=self.fModificar, bg="gray9", fg="white")

    self.btnModificar.place(x=5,y=90,width=90, height=30)


    self.btnEliminar=Button(frame1,text="Delete farmaco",
command=self.fEliminar, bg="gray9", fg="white")

    self.btnEliminar.place(x=5,y=130,width=90, height=30)

```

```
frame2 = Frame(self,bg="thistle4" )  
frame2.place(x=103,y=0,width=150, height=350)
```

```
lbl1 = Label(frame2,text="Nombre: ",bg="thistle2")  
lbl1.place(x=3,y=5)  
self.txtNombre=Entry(frame2)  
self.txtNombre.place(x=3,y=25,width=100, height=20)
```

```
lbl2 = Label(frame2,text="Tipo: ",bg="thistle2")  
lbl2.place(x=3,y=55)  
self.txtTipo=Entry(frame2)  
self.txtTipo.place(x=3,y=75,width=100, height=20)
```

```
lbl3 = Label(frame2,text="Cantidad: ",bg="thistle2")  
lbl3.place(x=3,y=105)  
self.txtCantidad=Entry(frame2)  
self.txtCantidad.place(x=3,y=125,width=100, height=20)
```

```
lbl4 = Label(frame2,text="Unidades: ",bg="thistle2")  
lbl4.place(x=3,y=155)  
self.txtUnidades=Entry(frame2)  
self.txtUnidades.place(x=3,y=175,width=100, height=20)
```

```
self.btnGuardar=Button(frame2,text="Guardar", command=self.fGuardar,  
bg="gray9", fg="white")
```

```
self.btnGuardar.place(x=10,y=310,width=60, height=30)
```

```
self.btnCancelar=Button(frame2,text="Cancelar", command=self.fCancelar,  
bg="gray9", fg="white")
```

```
self.btnCancelar.place(x=80,y=310,width=60, height=30)
```

```
frame3 = Frame(self,bg="thistle4" )
```

```
frame3.place(x=256,y=0,width=1200, height=400)
```

```
self.grid = ttk.Treeview(frame3, columns=("col1","col2","col3","col4"))
```

```
self.grid.column("#0",width=180)
```

```
self.grid.column("col1",width=250, anchor=CENTER)
```

```
self.grid.column("col2",width=250, anchor=CENTER)
```

```
self.grid.column("col3",width=250, anchor=CENTER)
```

```
self.grid.column("col4",width=250, anchor=CENTER)
```

```
self.grid.heading("#0", text="Id", anchor=CENTER)
```

```
self.grid.heading("col1", text="Nombre", anchor=CENTER)
```

```
self.grid.heading("col2", text="Tipo", anchor=CENTER)
```

```
self.grid.heading("col3", text="Cantidad", anchor=CENTER)
```

```
self.grid.heading("col4", text="Unidades", anchor=CENTER)
```

```
self.grid.pack(side=LEFT,fill=Y)
```

```
sb = Scrollbar(frame3,orient=VERTICAL)
```

```
sb.pack(side=RIGHT,fill=Y)
```

```
self.grid.config(yscrollcommand=sb.set)
```

```
sb.config(command=self.grid.yview)
```

```
self.grid["selectmode"]="browse"
```

Comandos de SQL

```
import mysql.connector
```

```
class Countries:
```

```
    def __init__(self):
```

```
        self.cnn = mysql.connector.connect(host="localhost",
```

```
            user="root",
```

```
            password="1523",
```

```
            port="3306",
```

```
            database="farmacia"
```

```
        )
```

```
    def __str__(self):
```

```
        datos=self.consulta_paises()
```

```
        aux=""
```

```
        for row in datos:
```

```
            aux=aux + str(row) + "\n"
```

```
        return aux
```

```
def consulta_paises(self):
```

```
    cur = self.cnn.cursor()
```

```
    cur.execute("SELECT * FROM adm")
```

```
    datos = cur.fetchall()
```

```
    cur.close()
```

```
    return datos
```

```
def buscar_pais(self, Id):
```

```
    cur = self.cnn.cursor()
```

```
    sql= "SELECT * FROM adm WHERE Id = {}".format(Id)
```

```
    cur.execute(sql)
```

```
    datos = cur.fetchone()
```

```
    cur.close()
```

```
    return datos
```

```
def inserta_pais(self,Nombre, Tipo, Cantidad, Unidades):
```

```
    cur = self.cnn.cursor()
```

```
    sql="INSERT INTO adm (nombre, tipo, cantidad, unidades)
```

```
VALUES('{}', '{}', '{}', '{}')".format(Nombre, Tipo, Cantidad, Unidades)
```

```
    cur.execute(sql)
```

```
    n=cur.rowcount
```

```
    self.cnn.commit()
```

```
    cur.close()
```

```
    return n
```

```
def elimina_pais(self,Id):
```

```
    cur = self.cnn.cursor()
```

```
    sql="DELETE FROM adm WHERE Id = {}".format(Id)
```

```
    cur.execute(sql)
```

```
    n=cur.rowcount
```

```
    self.cnn.commit()
```

```
    cur.close()
```

```
    return n
```

```
def modifica_pais(self,Id, Nombre, Tipo, Cantidad, Unidades):
```

```
    cur = self.cnn.cursor()
```

```
    sql="UPDATE adm SET nombre='{}', tipo='{}', cantidad='{}',
```

```
    unidades='{}' WHERE Id={}'.format(Nombre, Tipo, Cantidad, Unidades,Id)
```

```
    cur.execute(sql)
```

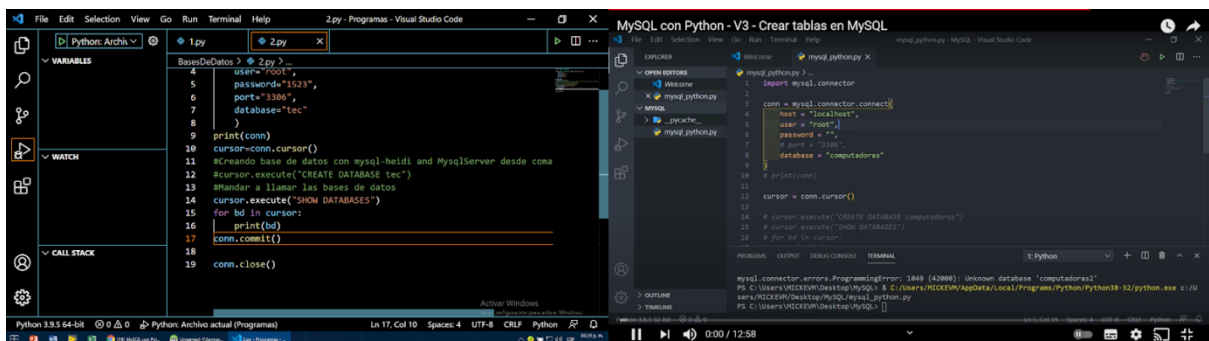
```
    n=cur.rowcount
```

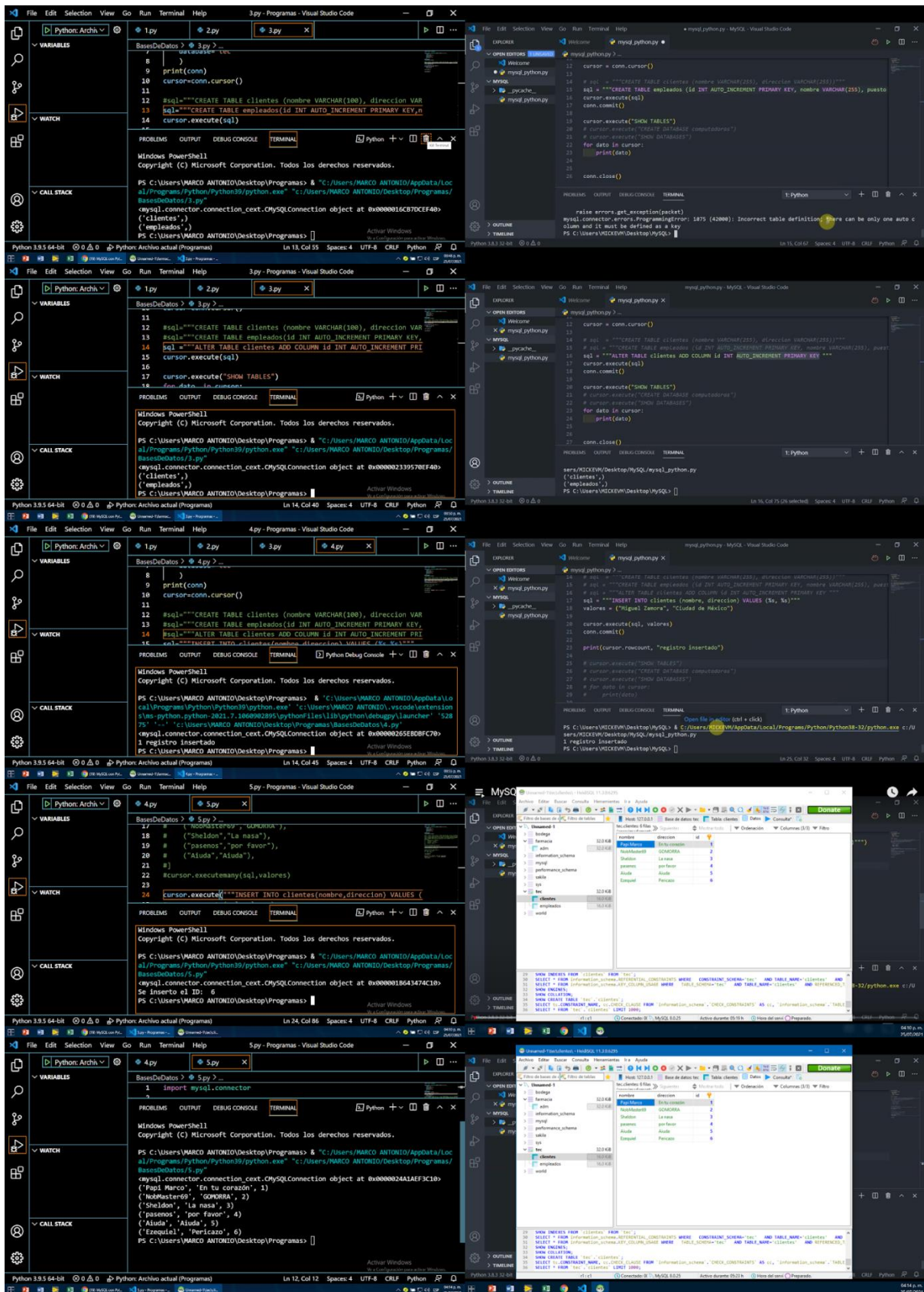
```
    self.cnn.commit()
```

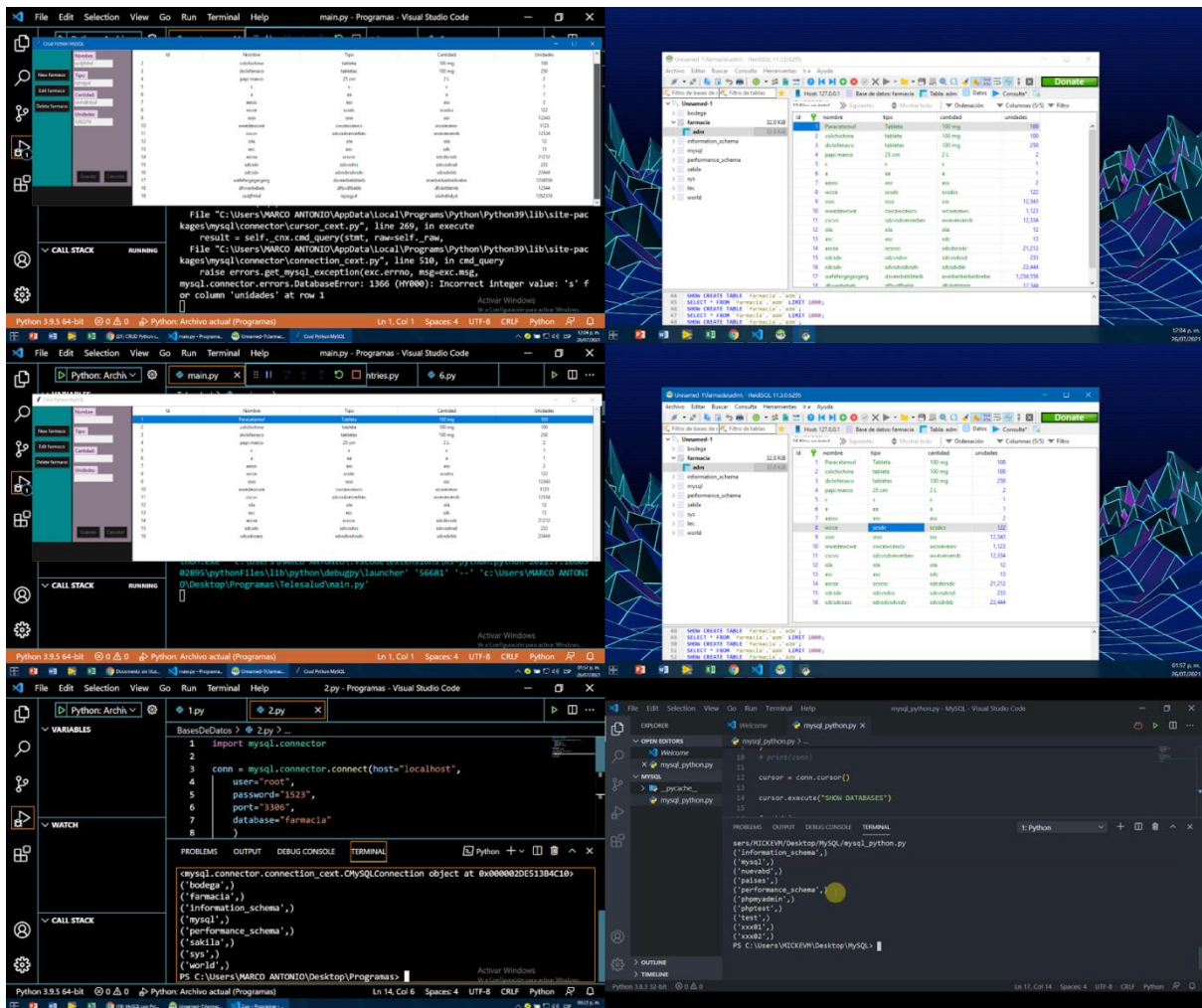
```
    cur.close()
```

```
    return n
```

Programación de la comunicación entre python y mysql, pruebas de funcionamiento vía terminal.







Proyecto terminado

Se han tenido percances en el desarrollo del proyecto, por el cual se trató de hacer una alternativa, la cual logro ser funcional, al terminarla.