

# **IES GONZALO NAZARENO**

**PROFESOR: CONCEPCIÓN GUISADO JURADO**

**ASIGNATURA: GESTIÓN DE BASES DE DATOS**

**CURSO: 1º ASIR**

**PRACTICA: PROYECTO PYTHON-BBDD POSTGRESQL**

# ÍNDICE

- 1- Muestra el nombre de cada empresa junto con su respectiva localidad.**
- 2- Muestra cargos que empiecen por una subcadena especificada.**
- 3- Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado**
- 4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.**
- 5- Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.**
- 6- Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.**

**Ejercicio1: Muestra el nombre de cada empresa junto con su respectiva localidad.**

```
def mostrar_empresas_por_localidad(db):  
    cursor = db.cursor()  
  
    try:  
        cursor.execute("SELECT Localidad, COUNT(*) FROM EMPRESA GROUP BY  
Localidad")  
  
        resultados = cursor.fetchall()  
  
        localidad_actual = None  
  
        total_empresas = 0  
  
        for localidad, nombre_empresa in resultados:  
  
            if localidad != localidad_actual:  
  
                print(f"Localidad: {localidad}")  
  
                localidad_actual = localidad  
  
                print(f"Nº de empresa: {nombre_empresa}")  
  
                total_empresas += 1  
  
        cursor.execute("SELECT COUNT(*) FROM EMPRESA")  
  
        total_empresas_en_db = cursor.fetchone()[0]  
  
        print(f"Total de empresas: {total_empresas_en_db}")  
  
    except Exception as e:  
  
        print("Se ha producido un error al ejecutar la consulta:", e)  
  
    finally:  
  
        cursor.close()
```

```
alejandro@debian:~/github/Proyecto_Python_BBDD_PostgreSQL/BBDD-Python PostgreSQL$ python3 programa.py
```

- 1- Muestra el nombre de cada empresa junto con su respectiva localidad.
- 2- Muestra cargos que empiecen por una subcadena especificada.
- 3- Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado
- 4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.
- 5- Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.
- 6- Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.
- 7- Salir

Selecciona una opción: 1

```
Localidad: Cordoba
Nº de empresa: 1
Localidad: Sevilla
Nº de empresa: 2
Localidad: Malaga
Nº de empresa: 1
Localidad: Granada
Nº de empresa: 1
Localidad: Cadiz
Nº de empresa: 1
Localidad: Utrera
Nº de empresa: 1
Localidad: Huelva
Nº de empresa: 1
Localidad: Almeria
Nº de empresa: 1
Localidad: Jaen
Nº de empresa: 1
Total de empresas: 10
```

**Ejercicio2:Muestra cargos que empiecen por una subcadena especificada.**

**def buscar\_cargos(conexion, subcadena):**

```
    cursor = conexion.cursor()
```

```
    consulta = ("SELECT Cargo FROM persona_de_contacto WHERE Cargo LIKE %s")
```

```
    cursor.execute(consulta, (subcadena + '%',))
```

```
    cargos_encontrados = cursor.fetchall()
```

```
    if cargos_encontrados:
```

```
        print("Cargos encontrados que comienzan con '{}':".format(subcadena))
```

```
        for cargo in cargos_encontrados:
```

```
            print(cargo[0])
```

else:

**print("No se encontraron cargos que comiencen con '{}".format(subcadena))**

```

1- Muestra el nombre de cada empresa junto con su respectiva localidad.
2- Muestra cargos que empiecen por una subcadena especificada.
3- Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado
4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.
5- Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.
6- Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.
7- Salir

```

Selecciona una opción: 2

Introduce la subcadena para buscar los cargos: In  
 Cargos encontrados que comienzan con 'In':  
 Informático  
 Ingeniero

```

1- Muestra el nombre de cada empresa junto con su respectiva localidad.
2- Muestra cargos que empiecen por una subcadena especificada.
3- Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado
4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.
5- Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.
6- Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.
7- Salir

```

Selecciona una opción:

**Ejercicio3:Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado.**

**def mostrar\_area\_trabajo\_empleado(db, nombre\_empleado):**

**try:**

**cursor = db.cursor()**

**sql = """SELECT Nombre, Descripción**

**FROM areatrabajo**

**WHERE ID\_AreaTrabajo = (**

**SELECT ID\_AreaTrabajo**

**FROM areatrabajo**

**WHERE Nombre = %s**

**)"""**

```
cursor.execute(sql, (nombre_empleado,))
```

```
resultado = cursor.fetchone()
```

```
if resultado:
```

```
    print(f"Área de trabajo de {nombre_empleado}:")
```

```
    print(f"Nombre: {resultado[0]}\nDescripción: {resultado[1]}")
```

```
else:
```

```
    print(f"No se encontró información del área de trabajo para {nombre_empleado}")
```

```
except Exception as e:
```

```
    print(f"Ocurrió un error: {e}")
```

```
1- Muestra el nombre de cada empresa junto con su respectiva localidad.
```

```
2- Muestra cargos que empiecen por una subcadena especificada.
```

```
3- Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado
```

```
4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.
```

```
5- Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.
```

```
6- Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.
```

```
7- Salir
```

Selecciona una opción: 3

Introduce el nombre del empleado: Alejandro

Área de trabajo de Alejandro:

Nombre: Alejandro

Descripción: Área de desarrollo de software

```
1- Muestra el nombre de cada empresa junto con su respectiva localidad.
```

```
2- Muestra cargos que empiecen por una subcadena especificada.
```

```
3- Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado
```

```
4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.
```

```
5- Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.
```

```
6- Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.
```

```
7- Salir
```

Selecciona una opción:

**Ejercicio4: Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.**

**def agregar\_empresa(db, cif, nombre, direccion, localidad):**

**cursor = db.cursor()**

**sql = "INSERT INTO empresa (cif, nombre, direccion, localidad) VALUES (%s, %s, %s, %s)" # Corregido el nombre de la tabla a 'empresa'**

**val = (cif, nombre, direccion, localidad)**

**cursor.execute(sql, val)**

**db.commit()**

**print(cursor.rowcount, "empresa insertada.")**

**def actualizar\_tabla\_empresa(db):**

**cursor = db.cursor()**

**sql = "SELECT \* FROM empresa"**

**try:**

**cursor.execute(sql)**

**registros = cursor.fetchall()**

**print("TABLA ACTUALIZADA EMPRESA:")**

**for result in registros:**

**print("CIF:", result[0])**

**print("Nombre:", result[1])**

**print("Direccion:", result[2])**

**print("Localidad:", result[3])**

**print("-----")**

**except:**

**print("Se ha producido un error al mostrar la tabla.")**

- 1- Muestra el nombre de cada empresa junto con su respectiva localidad.
- 2- Muestra cargos que empiecen por una subcadena especificada.
- 3- Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado
- 4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.
- 5- Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.
- 6- Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.
- 7- Salir

```

Selecciona una opción: 4
Introduce el cif de la empresa: K55443322
Introduce el nombre de la empresa: ALDI
Introduce la dirección de la empresa: Calle Real
Introduce la localidad de la empresa: Huelva
1 empresa insertada.
TABLA ACTUALIZADA EMPRESA:
CIF: A12345678
Nombre: Mercadona
Dirección: Calle Sevilla
Localidad: Utrera
-----
CIF: B46066361
Nombre: Ford
Dirección: Calle Arenal
Localidad: Sevilla
-----
CIF: C28328508
Nombre: UPS
Dirección: Calle Pureza
Localidad: Cádiz
-----
CIF: D65005357
Nombre: Iberia
Dirección: Calle Castilla
Localidad: Córdoba
-----
CIF: F46078986
Nombre: Consum
Dirección: Calle Tetuan
Localidad: Almería
-----
CIF: G09266298
Nombre: FC Barcelona
Dirección: Calle Sierpes
Localidad: Jaén
-----
CIF: W0049001A
Nombre: ARAG SE
Dirección: Calle Góles
Localidad: Huelva
-----
CIF: H11223344
Nombre: Burger King
Dirección: Calle Rioja
Localidad: Málaga
-----
CIF: I55667788
Nombre: McDonalds
Dirección: Calle Velázquez
Localidad: Granada
-----
CIF: J99887766
Nombre: Abengoa
Dirección: Avenida Portugal
Localidad: Sevilla
-----
CIF: K55443322
Nombre: ALDI
Dirección: Calle Real
Localidad: Huelva
-----

```

**Ejercicio5: Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.**

**def eliminar\_personas\_por\_prefijo(db, prefijo):**

**try:**

**cursor = db.cursor()**

**sql = "DELETE FROM persona\_de\_contacto WHERE Cargo LIKE %s"**

**cursor.execute(sql, (prefijo + '%',))**

**filas\_afectadas = cursor.rowcount**

**print(f"Se han eliminado {filas\_afectadas} registros donde el nombre comienza con '{prefijo}'.")**

**db.commit()**

**except:**



```
    print("Se ha producido un error al eliminar los clientes.")
    db.rollback()
def actualizar_tabla_persona_contacto(db):
    cursor = db.cursor()
    sql = "SELECT * FROM persona_de_contacto"
    try:
        cursor.execute(sql)
        registros = cursor.fetchall()
        print("TABLA ACTUALIZADA PERSONA DE CONTACTO:")
        for result in registros:
            print("Codigo_ContactoEmpresa:", result[0])
            print("Cargo:", result[1])
            print("Nombre:", result[2])
            print("Correo_electronico:", result[3])
            print("CIF:", result[4])
            print("-----")
    except:
        print("Se ha producido un error al mostrar la tabla.")
```

- 1- Muestra el nombre de cada empresa junto con su respectiva localidad.
- 2- Muestra cargos que empiecen por una subcadena especificada.
- 3- Pide por teclado un nombre de empleado y muestre la descripción del área de trabajo asociada a ese empleado.
- 4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la información de la tabla empresa.
- 5- Elimina todos los cargos cuyos nombres comienzan con un prefijo específico.
- 6- Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.
- 7- Salir

Selecciona una opción: 5

Introduce el prefijo que quieras eliminar: Pro  
 Se han eliminado 2 registros donde el nombre comienza con 'Pro'.  
 TABLA ACTUALIZADA PERSONA DE CONTACTO:  
 Codigo\_ContactoEmpresa: 111  
 Cargo: Gerente  
 Nombre: Juan  
 Correo\_electronico: juan.ruiz@gmail.com  
 CIF: A12345678

-----  
 Codigo\_ContactoEmpresa: 222  
 Cargo: Desarrollador  
 Nombre: Daniel  
 Correo\_electronico: daniel.diane@gmail.com  
 CIF: B46066361

-----  
 Codigo\_ContactoEmpresa: 333  
 Cargo: Coordinador  
 Nombre: Mario  
 Correo\_electronico: mario.ruiz@gmail.com  
 CIF: C28328508

-----  
 Codigo\_ContactoEmpresa: 777  
 Cargo: Analista  
 Nombre: Angela  
 Correo\_electronico: angela.gonz@gmail.com  
 CIF: W0049001A

-----  
 Codigo\_ContactoEmpresa: 999  
 Cargo: Arquitecta  
 Nombre: Paula  
 Correo\_electronico: paula.prieto@gmail.com  
 CIF: I55667788

-----  
 Codigo\_ContactoEmpresa: 1000  
 Cargo: Jefa  
 Nombre: Lucia  
 Correo\_electronico: lucia.romero@gmail.com  
 CIF: J99887766

-----  
 Codigo\_ContactoEmpresa: 444  
 Cargo: Informatico  
 Nombre: Cristian  
 Correo\_electronico: cristian.pinillos@gmail.com  
 CIF: D65005357

-----  
 Codigo\_ContactoEmpresa: 666  
 Cargo: Ingeniero  
 Nombre: Pablo  
 Correo\_electronico: pablo.rodri@gmail.com  
 CIF: G09266298

**Ejercicio6: Actualiza la información de la descripción de la tabla área de trabajo solicitando al usuario el código de área de trabajo y el nombre que desea actualizar.**

**def actualizar\_area\_trabajo(db, ID\_AreaTrabajo, campo, nuevo\_valor):**

**sql = "UPDATE areatrabajo SET Nombre = %s WHERE ID\_AreaTrabajo = %s"**

**try:**

**cursor = db.cursor()**

**cursor.execute(sql, (nuevo\_valor, ID\_AreaTrabajo))**

**db.commit()**

**print("La información ha sido actualizada correctamente.")**

**except Exception as e:**

**db.rollback()**

**print("No se ha podido actualizar la información:", e)**

```
def actualizar_tabla_area_trabajo(db):
```

```
    cursor = db.cursor()
```

```
    sql = "SELECT * FROM areatrabajo"
```

```
    try:
```

```
        cursor.execute(sql)
```

```
        registros = cursor.fetchall()
```

```
        print("TABLA ACTUALIZADA PERSONA DE CONTACTO:")
```

```
        for result in registros:
```

```
            print("ID_AreaTrabajo:", result[0])
```

```
            print("Nombre:", result[1])
```

```
            print("Descripcion:", result[2])
```

```
            print("-----")
```

```
    except:
```

```
        print("Se ha producido un error al mostrar la tabla.")
```

- 1- Muestra el nombre de cada empresa junto con su respectiva localidad.
- 2- Muestra cargos que empiecen por una subcadena especificada.
- 3- Pide por teclado un nombre de empleado y muestre la descripcion del area de trabajo asociada a ese empleado
- 4- Pide por teclado los datos de una nueva empresa. Luego, inserta los datos en la tabla empresa y muestra una tabla actualizada con toda la informacion de la tabla empresa.
- 5- Elimina todos los cargos cuyos nombres comienzan con un prefijo especifico.
- 6- Actualiza la informacion de la descripcion de la tabla area de trabajo solicitando al usuario el codigo de area de trabajo y el nombre que desea actualizar.
- 7- Salir

```

Selecciona una opción: 6
Introduce la ID del Area de Trabajo que desea actualizar: 111
Introduce el campo que desea actualizar (Nombre, Descripcion): Nombre
Introduce el nuevo valor para el campo seleccionado: Miguel
La información ha sido actualizada correctamente.
TABLA ACTUALIZADA PERSONA DE CONTACTO:
ID_AreaTrabajo: 222
Nombre: Raul
Descripcion: Recursos Humanos
-----
ID_AreaTrabajo: 333
Nombre: Daniel
Descripcion: Ventas
-----
ID_AreaTrabajo: 444
Nombre: Pablo
Descripcion: Soporte Tecnico
-----
ID_AreaTrabajo: 555
Nombre: Fran
Descripcion: Marketing
-----
ID_AreaTrabajo: 666
Nombre: Manuel
Descripcion: Investigacion y Desarrollo
-----
ID_AreaTrabajo: 777
Nombre: Gonzalo
Descripcion: Operaciones
-----
ID_AreaTrabajo: 888
Nombre: Mario
Descripcion: Calidad
-----
ID_AreaTrabajo: 999
Nombre: Cristian
Descripcion: Finanzas
-----
ID_AreaTrabajo: 1000
Nombre: Juan
Descripcion: Proyectos Especiales
-----
ID_AreaTrabajo: 111
Nombre: Miguel
Descripcion: Area de desarrollo de software
-----

```