



SECRETARÍA DE
INNOVACIÓN

Agenda

Sesión 13/18

1. Punto 1: primary key
2. Punto 2: autoincrementables
3. Punto 3: cláusula UNIQUE
4. Punto 4: cláusula WHERE
5. Punto 5: consultas con variables

PROGRAMACIÓN CON PYTHON

PRIMARY KEY

Bases de datos

Claves principales

En una BD relacionar se debe asignar un identificador único en cada tabla. Es un campos clave que representa cada registro de forma independiente.

```
import sqlite3

conexion=sqlite3.connect("AdministracionAlumnos")

cursor = conexion.cursor()

cursor.execute("""
    CREATE TABLE CURSOS (
        codigo_curso VARCHAR(4) PRIMARY KEY,
        nombre_curso VARCHAR(50),
        cupos_curso INTEGER,
        horas_curso INTEGER)
    "" #instrucción en varias líneas
    )
```

```
loscursos=[
    ("CI01","Python básico",50,36), #CI01 campo clave
    ("CI02","Java",150,30),
    ("CI03","C#",250,50),
    ("CI04","Python intermedio",50,50),
    ("CI05","Python avanzado",75,100)
]

cursor.executemany("INSERT INTO CURSOS VALUES (?,?,?,?)", loscursos)

conexion.commit()

conexion.close()
```



```
import sqlite3
```

```
conexion=sqlite3.connect("AdministracionAlumnos")
```

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

```
#Lo insertar porque no existe
```

```
cursor.execute("INSERT INTO CURSOS VALUES ('CI06','PHP',60,100)")
```

```
conexion.commit()
```

```
conexion.close()
```



```
import sqlite3
```

```
conexion=sqlite3.connect("AdministracionAlumnos")
```

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

```
#No lo inserta porque ya existe
```

```
cursor.execute("INSERT INTO CURSOS VALUES ('CI04','JS',40,50)")
```

```
conexion.commit()
```

```
conexion.close()
```

```
#Error UNIQUE en consola
```

PROGRAMACIÓN CON PYTHON

AUTO INCREMENTABLES


```
import sqlite3
```

```
conexion=sqlite3.connect("AdministracionAlumnos")
```

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

```
cursor.execute("""
```

```
    CREATE TABLE CURSOS (
```

```
    id_curso INTEGER PRIMARY KEY AUTOINCREMENT,
```

```
    nombre_curso VARCHAR(50),
```

```
    cupos_curso INTEGER,
```

```
    horas_curso INTEGER)
```

```
""
```

```
)
```

```
loscursos=[
```

```
    ("Python básico",50,36), #eliminanos el dato del campo 1
```

```
    ("Java",150,30),
```

```
    ("C#",250,50),
```

```
    ("Python intermedio",50,50),
```

```
    ("Python avanzado",75,100)
```

```
]
```

#Sustituir el primer valor de VALUES por NULL

```
cursor.executemany("INSERT INTO CURSOS VALUES (NULL,?,?,?)", loscursos)
```

```
conexion.commit()
```

```
conexion.close()
```

```
import sqlite3  
conexion=sqlite3.connect("AdministracionAlumnos")  
cursor = conexion.cursor()
```

```
loscursos=[  
    ("Python básico",50,36),  
    ("Java",150,30),  
    ("C#",250,50),  
    ("Python intermedio",50,50),  
    ("Python avanzado",75,100)  
]
```

```
cursor.executemany("INSERT INTO CURSOS VALUES (NULL,?,?,?)", loscursos)
```

```
conexion.commit()  
conexion.close()]
```

PROGRAMACIÓN CON PYTHON

CLÁUSULA UNIQUE

Bases de datos

Impide que la información de un campo se repita en la BD

```
import sqlite3

conexion=sqlite3.connect("AdministracionAlumnos")

cursor = conexion.cursor()

cursor.execute("""

    CREATE TABLE CURSOS (

        id_curso INTEGER PRIMARY KEY AUTOINCREMENT,

        codigo_curso VARCHAR(10) UNIQUE,

        nombre_curso VARCHAR(50),

        cupos_curso INTEGER,

        horas_curso INTEGER)

""")
```

```
los cursos=[

    ("CODCUR001","Python básico",50,36),

    ("CODCUR002","Java",150,30),

    ("CODCUR003","C#",250,50),

    ("CODCUR004","Python intermedio",50,50),

    ("CODCUR005","Python avanzado",75,100)

]
```

```
cursor.executemany("INSERT INTO CURSOS VALUES (NULL,?,?,?,?,?)", los cursos)
```

```
conexion.commit()
```

```
conexion.close()
```

```
import sqlite3
conexion=sqlite3.connect("AdministracionAlumnos")
cursor = conexion.cursor()

los cursos=[
    ("CODCUR006","Python básico",50,36),
    ("CODCUR007","Java",150,30),
    ("CODCUR008","C#",250,50),
    ("CODCUR001","Python intermedio",50,50), #error UNIQUE
    ("CODCUR009","Python avanzado",75,100)
]

cursor.executemany("INSERT INTO CURSOS VALUES (NULL,?,?,?,?,?)", los cursos)

conexion.commit()
conexion.close()
```

PROGRAMACIÓN CON PYTHON

CLÁUSULA WHERE

```
import sqlite3
```

```
conexion=sqlite3.connect("AdministracionAlumnos")
```

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

```
cursor.execute("SELECT * FROM CURSOS WHERE nombre_curso = 'Java'") #case sensitive
```

```
loscursos=cursor.fetchall()
```

```
print(loscursos)
```

```
conexion.commit()
```

```
conexion.close()
```



```
import sqlite3
conexion=sqlite3.connect("AdministracionAlumnos")
cursor = conexion.cursor()

cursor.execute("UPDATE CURSOS SET cupos_curso=87 WHERE nombre_curso = 'Java'")

cursor.execute("SELECT * FROM CURSOS WHERE nombre_curso = 'Java'")
loscursos=cursor.fetchall()
print(loscursos)

conexion.commit()
conexion.close()
```

```
import sqlite3
conexion=sqlite3.connect("AdministracionAlumnos")
cursor = conexion.cursor()

cursor.execute("DELETE FROM CURSOS WHERE nombre_curso = 'Java'")

cursor.execute("SELECT * FROM CURSOS WHERE nombre_curso = 'Java'")
loscursos=cursor.fetchall()
print(loscursos)

conexion.commit()
conexion.close()
```

PROGRAMACIÓN CON PYTHON

CONSULTAS CON VARIABLES

```
import sqlite3
conexion=sqlite3.connect("AdministracionAlumnos")
cursor = conexion.cursor()

elcodigo = input("Digite el código del curso que desea consultar: "),

cursor.execute("SELECT * FROM CURSOS WHERE codigo_curso = ?",elcodigo)
loscursos=cursor.fetchall()
print(loscursos)

conexion.commit()
conexion.close()
```

RESUMEN DE SESIÓN



SECRETARÍA DE
INNOVACIÓN