

Carga Masiva

Teoria:

copy_from (archivo, tabla, sep = '\t', null = '\\ N', tamaño = 8192, columnas = Ninguno)

Lea los datos del archivo de objeto similar a un archivo y los anexe a la tabla denominada tabla.

Parámetros:

archivo: objeto similar a un archivo desde el que leer datos. Debe tener métodos read () y readline ().

tabla: nombre de la tabla en la que copiar los datos.

sep: separador de columnas esperado en el archivo. Por defecto es una pestaña.

null: representación textual de NULL en el archivo. El valor predeterminado es la cadena de dos caracteres \ N.

tamaño: tamaño del búfer utilizado para leer del archivo.

columnas: iterable con el nombre de las columnas a importar. La longitud y los tipos deben coincidir con el contenido del archivo a leer. Si no se especifica, se asume que toda la tabla coincide con la estructura del archivo.

Practica:

```
>>> f = StringIO ("42 \tfoo \n74 \tbar \n")
>>> cur.copy_from (f, 'prueba', columnas = ('num', 'datos'))
>>> cur.execute ("seleccionar * de la prueba donde id> 5;")
>>> cur.fetchall ()
[(6, 42, 'foo'), (7, 74, 'bar')]
```

Codigo:

```
def copiarTemporal(self):
self.salida = ""
conex= psycopg2.connect(host="localhost", database="blockbuster", user="root",
password="P@ssw0rd")
cur = conex.cursor()
with open('/home/ferand20/Escritorio/Api-Blockbuster/BlockbusterData.csv', 'r') as f:
next(f)
cur.copy_from(f, 'temporal', sep=';',null='-')
conex.commit()
cur.close()
conex.close()
```

```
self.salida="Datos Insertados En Temporal"
```