

Creación de la Bases de Datos y la Tabla:

SQLQuery2.sql - LAPTOP-AFF2SUSG\SQLEXPRESS.master (LAPTOP-AFF2SUSG)\Iván (63)*

```
CREATE DATABASE Criptomonedas;

CREATE TABLE Criptomonedas (
    Nombre VARCHAR(255),
    Ranking INT,
    Símbolo VARCHAR(255),
    Suministro FLOAT,
    [Máximo Suministro] FLOAT,
    [Capitalización de Mercado USD] FLOAT,
    [Volumen USD] FLOAT,
    [Precio USD] FLOAT,
    [Cambio Porcentual 24h] FLOAT,
    [Precio PPV 24h] FLOAT,
    [Fuente de Datos] VARCHAR(255)
);

SELECT * FROM Criptomonedas;
```

100 %

Results Messages

Nombre	Ranking	Símbolo	Suministro	Máximo Suministro	Capitalización de Mercado USD	Volumen USD	Precio USD	Cambio Porcentual 24h	Precio PPV 24h	Fuente de Datos
--------	---------	---------	------------	-------------------	-------------------------------	-------------	------------	-----------------------	----------------	-----------------

Query executed successfully. | LAPTOP-AFF2SUSG\SQLEXPRESS ... | LAPTOP-AFF2SUSG\Iván (63) | master | 00:00:00 | 0 rows

Luego con el siguiente código puede conectarme a la base de datos y copiar los mismos. Además, al finalizar me guarda el resultado en un archivo CSV:

```
# Importo las librerías
import pandas as pd
import pyodbc

# Valores de la configuración de SQL Server
server = 'LAPTOP-AFF2SUSG\SQLEXPRESS'
database = 'Criptomonedas'

# Establezco la cadena de conexión
connection = f'DRIVER=SQL Server;SERVER={server};DATABASE={database};'
cnxn = pyodbc.connect(connection)

# Leer el archivo CSV en un DataFrame de pandas
df = pd.read_csv('C:\\Users\\Iván\\Downloads\\criptos.csv')

# Inserción de datos en la tabla
cursor = cnxn.cursor()
for index, row in df.iterrows():
    cursor.execute('INSERT INTO Criptomonedas (Nombre, Ranking, Símbolo,
Suministro, [Máximo Suministro], [Capitalización de Mercado USD], [Volumen USD],
[Precio USD], [Cambio Porcentual 24h], [Precio PPV 24h], [Fuente de Datos])
values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)',
                row['name'], row['rank'], row['symbol'], row['supply'],
row['maxSupply'], row['marketCapUsd'], row['volumeUsd24Hr'], row['priceUsd'],
row['changePercent24Hr'], row['vwap24Hr'], row['explorer'])
    cnxn.commit()

# Consulta SQL para obtener los datos de la tabla
sql_query = "SELECT * FROM Criptomonedas"

# Carga de datos desde la base de datos
df = pd.read_sql(sql_query, cnxn)

# Cierre de la conexión
cnxn.close()

# Guardar los datos en un archivo CSV
df.to_csv('C:\\Users\\Iván\\Downloads\\Criptomonedas.csv', index=False)
```

SELECT * FROM Criptomonedas;

100 %

Results Messages

	Nombre	Ranking	Símbolo	Suministro	Máximo Suministro	Capitalización de Mercado USD	Volumen USD	Precio USD	Cambio Porcentual 24h	Precio PPV 24h	Fuente de Datos
1	Bitcoin	1	BTC	19532106	21000000	677180111394,758	5173757468,69113	34670,102209908	0,0941413041840563	34612,174375081	https://blockchain.info
2	Ethereum	2	ETH	120268317,073653	0	218512421884,069	2884795134,71648	1816,87436226658	0,906497395600307	1799,38184778092	https://etherscan.io
3	Tether	3	USDT	85202267839,0692	0	85287285196,1404	10888572786,7588	1,00099782974359	0,025704410893472	1,00040251739656	https://www.trustnodes.com/en/coin/tether
4	BNB	4	BNB	166801148	166801148	38222041089,5675	211061055,318595	229,147350290224	-0,545513619900399	229,677836026672	https://etherscan.io
5	XRP	5	XRP	45404028640	100000000000	27667027952,6434	652575816,957041	0,609351830252995	1,29047045463584	0,603907549327026	https://xrpl.org
6	USDC	6	USDC	24483315897,7097	0	24499961483,3309	766917518,20708	1,0006798746416	0,0010353134446601	1,00065136396014	https://etherscan.io
7	Solana	7	SOL	419584565,761129	0	16662691912,0352	705002302,888872	39,7123566302039	0,645751497836975	39,5234504642983	https://explorer.solana.com
8	Cardano	8	ADA	35248610987,114	45000000000	11188371811,8427	218856406,340171	0,317413126319584	3,96413037676942	0,319958049489869	https://cardanoexplorer.com
9	Dogecoin	9	DOGE	141680236383,705	0	9591823601,0372	127742171,544547	0,0677005053482559	0,243707184157852	0,0674942969663877	http://dogechain.info
10	TRON	10	TRX	88766666789,3748	0	8708386530,29284	70710949,8859707	0,0981042416626509	-1,14865350450718	0,0984893490526544	https://tronscan.org
11	Chainlink	11	LINK	556849970,452787	1000000000	6245230673,31992	297901708,138815	11,2152841962832	2,36652858919564	11,0158865330378	https://etherscan.io
12	Polygon	12	MATIC	9244372993,46675	10000000000	6136031618,29586	149781508,909629	0,663758550486048	1,9507234298787	0,657947051775484	https://etherscan.io
13	Polkadot	13	DOT	1292563565,45018	0	5958096841,9687	88867218,9130783	4,60951940873685	0,0277495487893128	4,57588082963279	https://polkadot.subscan.io
14	Wrapped Bitcoin	14	WBTC	163841,87097067	0	5686471256,01066	58207522,3967606	34707,069824835	0,374468617710041	34637,9474166331	https://etherscan.io
15	Multi Collateral DAI	15	DAI	5347888595,74641	0	5349455958,18806	61687885,172743	1,00029308060809	0,0645559906861911	1,00052826978022	https://etherscan.io
16	Bitcoin Cash	16	BCH	71011070,4735800	84000000	5116173708,80742	60000073,1243605	80,2200000007141	1,107141315864591	80,0550005400007	https://blockchain.info