|  |
| --- |
|  |
|  |



# Title: Connecting to Oracle Database from Python

# Author: Pritha

# Project: TDI

• Perform data processing operations here

• Create an instance of OracleDataExtractor

• Connect to Oracle database

• Execute the query to extract data

• Process the extracted data

• Print the processed data

• Disconnect from Oracle database

## Code Snippets

#Title: Connecting to Oracle Database from Python

#Author: Pritha

#Project: TDI

import cx\_Oracle

class OracleDataExtractor:

def \_\_init\_\_(self, username, password, host, port, service\_name):

self.username = username

self.password = password

self.host = host

self.port = port

self.service\_name = service\_name

self.connection = None

def connect(self):

"""Establishes a connection to Oracle database"""

dsn = cx\_Oracle.makedsn(self.host, self.port, service\_name=self.service\_name)

self.connection = cx\_Oracle.connect(self.username, self.password, dsn)

def extract\_data(self, query):

"""Executes the provided query and returns the result"""

cursor = self.connection.cursor()

cursor.execute(query)

result = cursor.fetchall()

cursor.close()

return result

def disconnect(self):

"""Closes the connection to Oracle database"""

self.connection.close()

def process\_data(data):

"""Processes the extracted data"""

# Perform data processing operations here

processed\_data = [item.upper() for item in data]

return processed\_data

# Create an instance of OracleDataExtractor

extractor = OracleDataExtractor(

username='your\_username',

password='your\_password',

host='your\_host',

port='your\_port',

service\_name='your\_service\_name'

)

try:

# Connect to Oracle database

extractor.connect()

# Execute the query to extract data

query = 'SELECT \* FROM your\_table'

extracted\_data = extractor.extract\_data(query)

# Process the extracted data

processed\_data = process\_data(extracted\_data)

# Print the processed data

for item in processed\_data:

print(item)

finally:

# Disconnect from Oracle database

extractor.disconnect()