**PYTHON ASSESSMENT -30.9.2022**

I. Write Python Code for the following

1. A while loop that asks the user for a number, and prints a countdown from that number to zero.

def count\_down(number):

    if number == 0:

        return

    else:

        print(number)

        count\_down(number - 1)

if \_\_name\_\_ == "\_\_main\_\_":

    input\_num = int(input("Enter a number: "))

    print("\nCounting down from:", input\_num)

    count\_down(input\_num)

    print("Finished counting down from:", input\_num)

1. To count the numbers of characters in the string and store them in a dictionary data structure Write a program to use split and join methods in the string and trace a birthday of a person with a dictionary data structure

def man\_person\_birthday(birth\_str):

    num\_of\_chars = len(birth\_str)

    print("Number of characters:", num\_of\_chars)

    for ch in birth\_str:

        print("Character:", ch)

if \_\_name\_\_ == "\_\_main\_\_":

    man\_person\_birthday('1997-01-28')

II. Python – DB Connectivity program

Create three tables:

1. Movies

2. Reviewers

3. Ratings

1. Movies - contains general information about movies and has the following attributes:

* id
* title
* release\_year
* genre
* collection\_in\_mil

2. Reviewers contains information about people who posted reviews or ratings and has the following attributes:

* id
* first\_name
* last\_name

3. Ratings contains information about ratings that have been posted and has the following attributes:

* movie\_id (foreign key)
* reviewer\_id (foreign key)
* rating

import pyodbc

def create():

    conn = pyodbc.connect(

        'Driver={SQL Server};'

        'Server=DESKTOP-VOT1DO6\MSSQLSERVER\_NEW;'

        'Database=HelloWorld;'

        'Trusted\_Connection=yes;'

    )

    cursor = conn.cursor()

    cursor.execute("""

        CREATE TABLE Movies(

            id INT IDENTITY(1,1) PRIMARY KEY,

            title VARCHAR(255) NOT NULL,

            release\_year VARCHAR(255) NOT NULL,

            genre VARCHAR(255) NOT NULL,

            collection\_in\_mil SMALLMONEY

        )

        CREATE TABLE Reviewers(\

            id INT IDENTITY(1,1) PRIMARY KEY,\

            first\_name VARCHAR(255) NOT NULL,\

            last\_name VARCHAR(255) NOT NULL\

        )

        CREATE TABLE Ratings(\

            movie\_id INT,\

            reviewer\_id INT,\

            rating INT NOT NULL,\

            FOREIGN KEY(movie\_id) REFERENCES Movies(id),\

            FOREIGN KEY(reviewer\_id) REFERENCES Reviewers(id)\

        )"""

    )

    conn.commit()

def insert():

    conn = pyodbc.connect(

        'Driver={SQL Server};'

        'Server=DESKTOP-VOT1DO6\MSSQLSERVER\_NEW;'

        'Database=HelloWorld;'

        'Trusted\_Connection=yes;'

    )

    cursor = conn.cursor()

    cursor.execute("""

        INSERT INTO Movies(title, release\_year, genre, collection\_in\_mil)

        VALUES

            ('movie1', '2000', 'horrr', 1000),

            ('movie2', '2000', 'action', 2500),

            ('movie3', '2000', 'adventure', 1500),

            ('movie4', '2000', 'horror', 7500),

            ('movie5', '2000', 'sci-fi', 9500);

        INSERT INTO Reviewers(first\_name, last\_name)

        VALUES

            ('john, 'snow'),

            ('name1, 'nile'),

            ('shay, 'brown'),

            ('brian, 'black'),

            ('jay, 'walker');

        INSERT INTO Movies(movie\_id, reviewer\_id, rating)

        VALUES

            (1, 1, 9),

            (1, 2, 10),

            (2, 1, 4),

            (1, 3, 3),

            (1, 4, 4);

        )"""

    )

    conn.commit()

def highest\_collection():

    pass

def highest\_rating():

    pass

def low\_rating():

    pass

if \_\_name\_\_ == "\_\_main\_\_":

    # create()

    insert()

    # highest\_collection()

    # highest\_rating()

    # low\_rating()

Questions :

* Connect DB by using python connectivity
* Insert 5 rows in all three tables
* Find the highest collection
* Find the highest rating
* Find the count of reviewers who gave low Rating