

Universidad Estatal a Distancia
Escuela de ciencias exactas y naturales

Cátedra de Ingeniería de software

Proyecto programado N°3

BASE DE DATOS

Código: 00826

Alumno (a): Cristian Josué Monge Fernández.

Ced:604510914

Centro Universitario: Quepos.

I Cuatrimestre, 2022

Contenidos

| | |
|---|---|
| Introducción | 3 |
| Función llamada "fnc_get_reino" | 4 |
| Procedimiento llamado "sp_inserta_registro" | 5 |
| Conclusión | 6 |
| Bibliografía..... | 7 |

Introducción

En el Presente trabajo veremos la forma en cómo se generan scripts para obtener ciertos parámetros que se solicitaran en el proyecto programado, veremos en la que se crea un función para recibir como parámetro de entrada el nombre común de un ser vivo y así devolver el reino al cual pertenece, también crearemos un procedimiento en el cual se reciben ciertos parámetros para así crear un nuevo registro en las tablas con los datos existentes, este procedimiento también sera capaz de comprobar si esta información ya existía previamente, de lo contrario lo creara.

1. Crear una función llamada "fnc_get_reino" la cual reciba como parámetro de entrada el nombre común de un ser vivo, y devuelva como resultado el reino al cual pertenece.

```
CREATE FUNCTION [dbo].[fnc_get_reino] (@nombreComun VARCHAR(30))
RETURNS VARCHAR(30)
AS
BEGIN
    RETURN (
        SELECT idReino
        FROM reino
        INNER JOIN phylum ON reino.idReino = phylum.reino_idReino
        INNER JOIN clase ON phylum.idPhylum = clase.Phylum_idPhylum
        INNER JOIN orden ON clase.idClase = orden.Clase_idClase
        INNER JOIN familia ON orden.idOrden = familia.Orden_idOrden
        INNER JOIN genero ON familia.idFamilia = genero.Familia_idFamilia
        INNER JOIN especie ON genero.idGenero = especie.Genero_idGenero
        WHERE especie.nombre_comun = @nombreComun
    )
END
GO
```

2. Genere un procedimiento llamado “sp_inserta_registro”, el cual reciba los siguientes parámetros: nombre común, nombre científico, especie, género, familia, orden, clase, phylum y reino, e inmediatamente cree un nuevo registro en las tablas de la base de datos, con los datos recibidos. El procedimiento debe ser capaz de determinar que si ya hay información existente en la base de datos, solo debe utilizarla, de lo contrario deberá crear lo que sea nuevo.

```
CREATE PROCEDURE [dbo].[sp_inserta_registro]
-- Params
(
@nombreComun varchar(30), @especie varchar(30),
@genero varchar(30), @familia varchar(30),@orden varchar(30), @clase varchar(30),
@phylum varchar(30), @reino varchar(30))
AS
BEGIN

    SET NOCOUNT ON;

    -- Como todos los campos de la mayoria de tablas con primary keys, la base de datos
    evita los duplicados

    INSERT INTO [dbo].[reino] ([idReino]) VALUES(@reino)
    INSERT INTO [dbo].[phylum]([idPhylum],[reino_idReino]) VALUES(@phylum,@reino)
    INSERT INTO [dbo].[clase]([idClase],[Phylum_idPhylum]) VALUES(@clase,@phylum)
    INSERT INTO [dbo].[orden]([idOrden],[Clase_idClase]) VALUES(@orden,@clase)
    INSERT INTO [dbo].[familia]([idFamilia],[Orden_idOrden]) VALUES(@familia,@orden)
    INSERT INTO [dbo].[genero]([idGenero],[Familia_idFamilia]) VALUES(@genero,@familia)
    INSERT INTO
[dbo].[especie]([idEspecie],[Genero_idGenero],[nombre_cientifico],[nombre_comun])
VALUES(@especie,@genero,@especie+' '+@genero,@nombreComun)
END
GO
```

Conclusión

En el pasado trabajo vimos la forma en cómo se generaron scripts para obtener ciertos parámetros que se solicitaban en el proyecto programado, vimos en la que se crea un función para recibir como parámetro de entrada el nombre común de un ser vivo y así devolver el reino al cual pertenece, también creamos un procedimiento en el cual se recibían ciertos parámetros para así crear un nuevo registro en las tablas con los datos existentes, este procedimiento también fue capaz de comprobar si esta información ya existe previamente, de lo contrario lo creo.

Bibliografía

N/A