

# Universidad Tecnológica de Panamá

Centro Regional De Veraguas



Facultad de Ingeniería de Sistemas  
Computacionales

Curso: Base de Datos II

Profesor: Carlos Herrera

---

LABORATORIO 4

---

Estudiante

Arland Barrera

2024

# Contenido

---

<b>1</b>	<b>Desarrollo</b>	<b>4</b>
1.1	Problemas resueltos . . . . .	4
1.1.1	Problema resuelto 1 . . . . .	4
1.1.2	Problema resuelto 2 . . . . .	4
1.1.3	Problema resuelto 3 . . . . .	5
1.1.4	Problema resuelto 4 . . . . .	5
1.1.5	Problema resuelto 5 . . . . .	6
1.1.6	Problema resuelto 6 . . . . .	6
1.1.7	Problema resuelto 7 . . . . .	7
1.1.8	Problema resuelto 8 . . . . .	7
1.1.9	Problema resuelto 9 . . . . .	8
1.1.10	Problema resuelto 10 . . . . .	8
1.1.11	Problema resuelto 11 . . . . .	9
1.2	Problemas propuestos . . . . .	10
1.2.1	Problema propuesto 1 . . . . .	10
1.2.2	Problema propuesto 2 . . . . .	11
1.2.3	Problema propuesto 3 . . . . .	12
1.2.4	Problema propuesto 4 . . . . .	13

# Lista de figuras

---

1.1	Problema resuelto 1 . . . . .	4
1.2	Problema resuelto 2 . . . . .	4
1.3	Problema resuelto 3 . . . . .	5
1.4	Problema resuelto 4 . . . . .	5
1.5	Problema resuelto 5 . . . . .	6
1.6	Problema resuelto 6 . . . . .	6
1.7	Problema resuelto 7 . . . . .	7
1.8	Problema resuelto 8 . . . . .	7
1.9	Problema resuelto 9 . . . . .	8
1.10	Problema resuelto 10 . . . . .	8
1.11	Problema resuelto 11 . . . . .	9
1.12	Problema propuesto 1 . . . . .	10
1.13	Problema propuesto 1 ejecución . . . . .	10
1.14	Problema propuesto 2 . . . . .	11
1.15	Problema propuesto 2 ejecución . . . . .	11
1.16	Problema propuesto 3 . . . . .	12
1.17	Problema propuesto 3 ejecución . . . . .	12
1.18	Problema propuesto 4 . . . . .	13
1.19	Problema propuesto 4 ejecución . . . . .	13

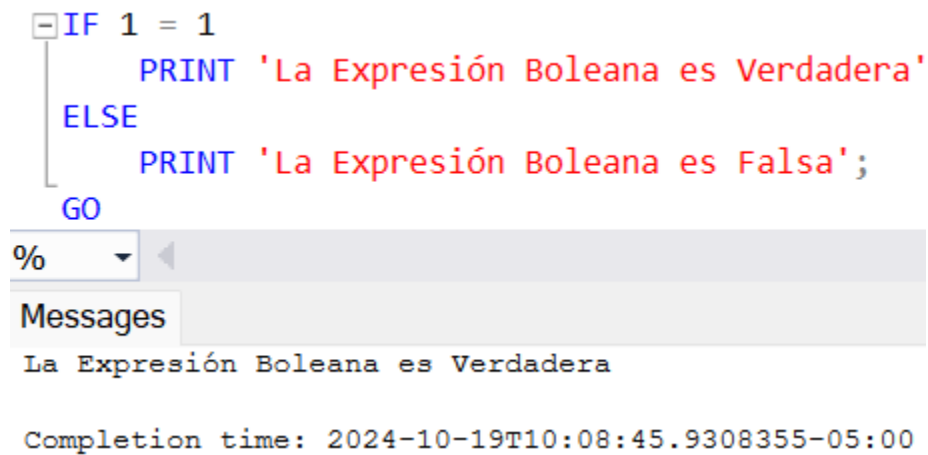
---

# Desarrollo

---

## 1.1 Problemas resueltos

### 1.1.1 Problema resuelto 1



```
IF 1 = 1
    PRINT 'La Expresión Boleana es Verdadera'
ELSE
    PRINT 'La Expresión Boleana es Falsa';
GO
```

%

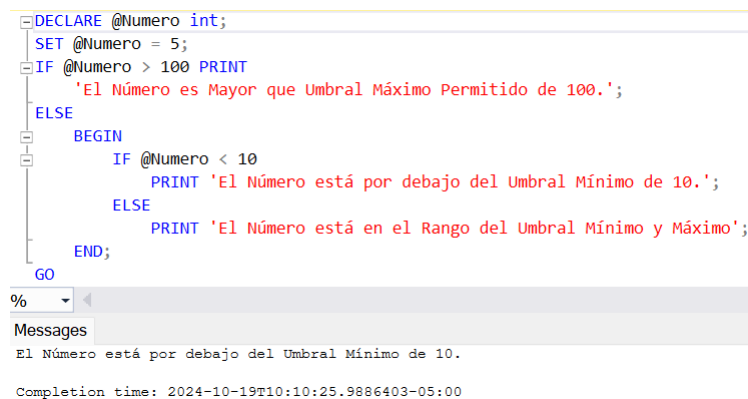
Messages

La Expresión Boleana es Verdadera

Completion time: 2024-10-19T10:08:45.9308355-05:00

Figura 1.1: Problema resuelto 1

### 1.1.2 Problema resuelto 2



```
DECLARE @Numero int;
SET @Numero = 5;
IF @Numero > 100 PRINT
    'El Número es Mayor que Umbral Máximo Permitido de 100.';
ELSE
    BEGIN
        IF @Numero < 10
            PRINT 'El Número está por debajo del Umbral Mínimo de 10.';
        ELSE
            PRINT 'El Número está en el Rango del Umbral Mínimo y Máximo';
    END;
GO
```

%

Messages

El Número está por debajo del Umbral Mínimo de 10.

Completion time: 2024-10-19T10:10:25.9886403-05:00

Figura 1.2: Problema resuelto 2

### 1.1.3 Problema resuelto 3

```
Declare @Nombre varchar(20), @Apellido varchar(20)
SET @Nombre='Juan'
SET @Apellido ='Molinar'
IF (@Nombre <> '' and @Apellido <> '')
BEGIN
    PRINT 'Nuevo Registrado Validado para insertar'
END
ELSE
BEGIN
    PRINT 'Se requiere por lo menos el nombre y apellido del Estudiante'
END
GO
```

Messages

Nuevo Registrado Validado para insertar

Completion time: 2024-10-19T10:11:22.0505954-05:00

Figura 1.3: Problema resuelto 3

### 1.1.4 Problema resuelto 4

```
DECLARE @Id Int, @Descripcion varchar(50), @Cantidad int, @Venta money,
@Costo money;
SET @Id = 100008;
SET @Descripcion ='Bocina inalambrica';
SET @Cantidad = 25; SET @Venta = 50.50;
SET @Costo = 40.75;
IF EXISTS(SELECT * FROM TB_Producto WHERE Codigo_Producto = @Id)
BEGIN
    PRINT 'Ya existe un registro con ese codigo'
END
ELSE
BEGIN
    INSERT INTO TB_Producto(Codigo_Producto, Descripcion_Producto,
Cantidad_Inventario, Precio_Venta, Precio_Costo)
VALUES(@Id, @Descripcion, @Cantidad, @Venta, @costo);
END
GO
```

Messages

(1 row affected)

Completion time: 2024-10-19T10:12:15.3646275-05:00

Figura 1.4: Problema resuelto 4

### 1.1.5 Problema resuelto 5

```
DECLARE @Calificacion INT, @NotaLetra VARCHAR(1)
SET @Calificacion=50
SET @NotaLetra= (CASE
    WHEN @Calificacion >= 90 THEN 'A'
    WHEN @Calificacion >= 81 AND @Calificacion < 90 THEN 'B'
    WHEN @Calificacion >= 71 AND @Calificacion < 80 THEN 'C'
    WHEN @Calificacion >= 61 AND @Calificacion < 70 THEN 'D'
    ELSE 'F'
END)
SELECT 'Su Evaluación correspondiente es: ' + @NotaLetra [Nota del Estudiante]
GO
```

results Messages

Nota del Estudiante
Su Evaluación correspondiente es: F

Figura 1.5: Problema resuelto 5

### 1.1.6 Problema resuelto 6

```
SELECT Codigo_Producto AS Codigo, Descripcion_Producto AS Descripcion,
Precio_Venta AS Precio,
CASE
    WHEN Precio_Venta < 50 THEN 'El producto es accesible'
    WHEN Precio_Venta >= 50 AND Precio_Venta < 200 THEN 'El producto es medianamente accesible'
    ELSE 'El producto es inaccesible'
END
AS Mensaje FROM TB_Producto;
```

results Messages

Codigo	Descripcion	Precio	Mensaje
100008	Bocina inalambrica	50.50	El producto es medianamente accesible

Figura 1.6: Problema resuelto 6

## 1.1.7 Problema resuelto 7

```
DECLARE @Precio MONEY, @Nombre varchar (50), @Producto INT, @Mensaje varchar(50)
SET @Producto = 100004
SELECT @Precio = Precio_Venta, @Nombre = Descripcion_Producto FROM TB_Producto
WHERE Codigo_Producto = @Producto
SET @Mensaje = (CASE
    WHEN @Precio < 50 THEN 'Producto es accesible'
    WHEN @Precio >= 50 AND @Precio < 200 THEN 'Producto es medianamente accesible'
    ELSE 'Producto es inaccesible'
END )
--Imprimir la consulta
print 'El producto : ' + @Nombre + ', el : ' + @Mensaje
GO
```

%

Messages

Completion time: 2024-10-19T10:15:35.9938389-05:00

Figura 1.7: Problema resuelto 7

## 1.1.8 Problema resuelto 8

```
DECLARE @Bandera INT
SET @Bandera = 1
WHILE (@Bandera <=5)
BEGIN
    PRINT @Bandera
    SET @Bandera = @Bandera + 1
    IF @Bandera = 4
    BEGIN
        PRINT 'Alcanzado el Umbral: ' + CAST(@Bandera AS VARCHAR(2))
        BREAK;
    END
END
GO
```

%

Messages

1  
2  
3  
Alcanzado el Umbral: 4

Completion time: 2024-10-19T10:16:42.1815192-05:00

Figura 1.8: Problema resuelto 8

## 1.1.9 Problema resuelto 9

```
DECLARE @IdRecurso INT;
SET @IdRecurso = 0;
WHILE @IdRecurso <= 10
BEGIN
    PRINT 'www.utp.ac.pa/BDII/Recurso' + CAST(@IdRecurso AS VARCHAR(2))
    SET @IdRecurso = @IdRecurso + 1
END
PRINT ' '
PRINT '-----'
PRINT 'Cantidad de Recursos Disponibles para BDII: ' + CAST(@IdRecurso AS VARCHAR(2));
GO
```

%

Messages

www.utp.ac.pa/BDII/Recurso0  
www.utp.ac.pa/BDII/Recurso1  
www.utp.ac.pa/BDII/Recurso2  
www.utp.ac.pa/BDII/Recurso3  
www.utp.ac.pa/BDII/Recurso4  
www.utp.ac.pa/BDII/Recurso5  
www.utp.ac.pa/BDII/Recurso6  
www.utp.ac.pa/BDII/Recurso7  
www.utp.ac.pa/BDII/Recurso8  
www.utp.ac.pa/BDII/Recurso9  
www.utp.ac.pa/BDII/Recurso10

-----

Cantidad de Recursos Disponibles para BDII: 11

Completion time: 2024-10-19T10:17:32.7021432-05:00

Figura 1.9: Problema resuelto 9

## 1.1.10 Problema resuelto 10

```
DECLARE @Cadena CHAR(15), @LCadena INT, @Indice INT, @Caracter CHAR(1), @Contar INT
SET @Cadena = 'Base de Datos'
SET @Contar= 0
SET @Indice = 1
SET @LCadena= LEN(@Cadena)
WHILE @Indice<= @LCadena
BEGIN
    SET @Caracter = SUBSTRING(@Cadena, @Indice, 1)
    PRINT @Caracter
    SET @Indice= @Indice+ 1
END
GO
```

%

Messages

B  
a  
s  
e  
  
d  
e  
  
D  
a  
t  
o  
s

Completion time: 2024-10-19T10:18:33.9389220-05:00

Figura 1.10: Problema resuelto 10



### 1.1.11 Problema resuelto 11

```
WHILE (SELECT AVG(Precio_Venta) FROM TB_Producto) < 500
BEGIN
    UPDATE TB_PRODUCTO SET Precio_Venta = Precio_Venta * 2
    IF (SELECT MAX(Precio_Venta) FROM TB_Producto) > 1000
        BREAK
    ELSE
        CONTINUE
END
GO
```

%

Messages

(1 row affected)

(1 row affected)

(1 row affected)

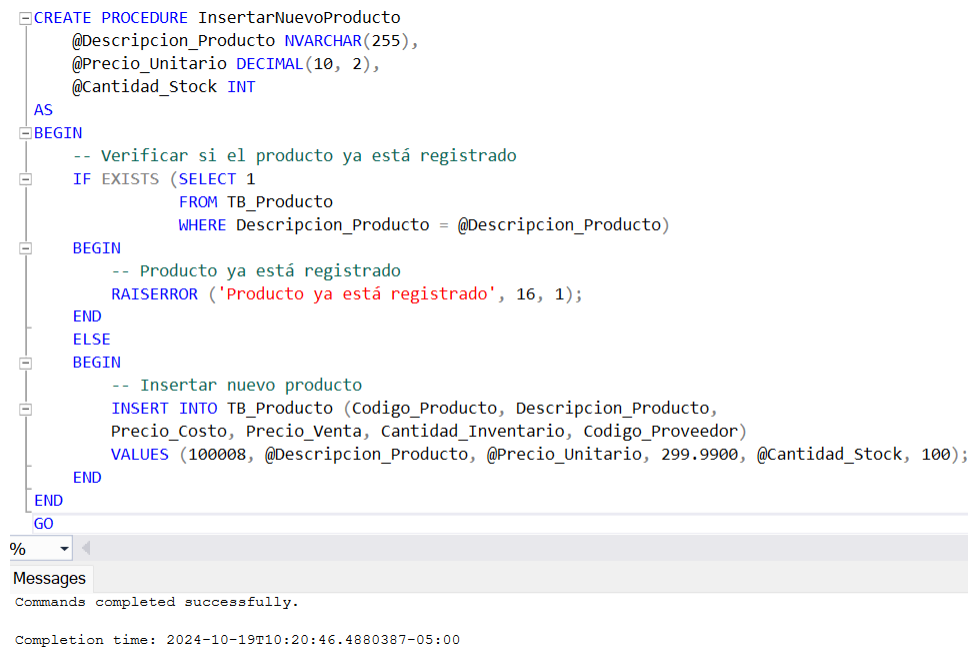
(1 row affected)

Completion time: 2024-10-19T10:19:19.6754372-05:00

Figura 1.11: Problema resuelto 11

## 1.2 Problemas propuestos

### 1.2.1 Problema propuesto 1



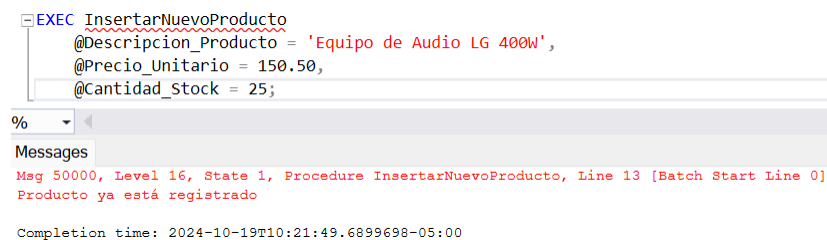
```
CREATE PROCEDURE InsertarNuevoProducto
    @Descripcion_Producto NVARCHAR(255),
    @Precio_Unitario DECIMAL(10, 2),
    @Cantidad_Stock INT
AS
BEGIN
    -- Verificar si el producto ya está registrado
    IF EXISTS (SELECT 1
               FROM TB_Producto
               WHERE Descripcion_Producto = @Descripcion_Producto)
    BEGIN
        -- Producto ya está registrado
        RAISERROR ('Producto ya está registrado', 16, 1);
    END
    ELSE
    BEGIN
        -- Insertar nuevo producto
        INSERT INTO TB_Producto (Codigo_Producto, Descripcion_Producto,
                                Precio_Costo, Precio_Venta, Cantidad_Inventario, Codigo_Proveedor)
        VALUES (100008, @Descripcion_Producto, @Precio_Unitario, 299.9900, @Cantidad_Stock, 100);
    END
END
GO
```

Messages

Commands completed successfully.

Completion time: 2024-10-19T10:20:46.4880387-05:00

Figura 1.12: Problema propuesto 1



```
EXEC InsertarNuevoProducto
    @Descripcion_Producto = 'Equipo de Audio LG 400W',
    @Precio_Unitario = 150.50,
    @Cantidad_Stock = 25;
```

Messages

Msg 50000, Level 16, State 1, Procedure InsertarNuevoProducto, Line 13 [Batch Start Line 0]  
Producto ya está registrado

Completion time: 2024-10-19T10:21:49.6899698-05:00

Figura 1.13: Problema propuesto 1 ejecución

## 1.2.2 Problema propuesto 2

```
CREATE PROCEDURE GenerarCartaCandidato
    @NombrePersona VARCHAR(50),
    @AnoNacimiento INT
AS
BEGIN
    DECLARE @Edad INT,
            @FechaActual DATE = GETDATE(),
            @DiaActual NVARCHAR(50),
            @NumeroDia INT,
            @MesActual NVARCHAR(50),
            @AnoActual INT,
            @Mensaje NVARCHAR(MAX);

    -- Calcular la edad del candidato
    SET @Edad = YEAR(@FechaActual) - @AnoNacimiento;

    -- Obtener los componentes de la fecha
    SET @DiaActual = DATENAME(dw, @FechaActual); -- Nombre del día
    SET @NumeroDia = DAY(@FechaActual); -- Número del día
    SET @MesActual = 'de ' + DATENAME(month, @FechaActual); -- Nombre del mes
    SET @AnoActual = YEAR(@FechaActual); -- Año actual

    -- Encabezado de la carta
    SET @Mensaje = 'San Antonio. Atalava. ' + @DiaActual + ' '

    %
    Messages
    Commands completed successfully.

    Completion time: 2024-10-19T10:23:02.7925102-05:00
```

Figura 1.14: Problema propuesto 2

```
EXEC GenerarCartaCandidato 'Fulano de Tal', 2005;

%
Messages
San Antonio, Atalaya, Saturday 19 de de October 2024

Apreciado Candidato: Fulano de Tal, según las normas de la empresa, se requiere personal mayor de 18 años.
Según los datos de su hoja de vida, usted tiene 19 años.

Por favor presentarse a la oficina de la empresa a partir de mañana a las 08:00 a.m. para tomar posesión del cargo en oposición.

Sin otro particular.
Atentamente,
Director de Recursos Humanos
Empresas XYZ Inc.

Completion time: 2024-10-19T10:23:36.8433065-05:00
```

Figura 1.15: Problema propuesto 2 ejecución

### 1.2.3 Problema propuesto 3

```
CREATE FUNCTION CategorizarProveedor (  
    @MontoAcumulado DECIMAL(18, 2),  
    @MontoUmbra1 DECIMAL(18, 2)  
)  
RETURNS NVARCHAR(50)  
AS  
BEGIN  
    DECLARE @Categoria NVARCHAR(50);  
  
    IF @MontoAcumulado > @MontoUmbra1  
        SET @Categoria = 'Proveedor VIP';  
    ELSE IF @MontoAcumulado <= @MontoUmbra1 AND @MontoAcumulado > (@MontoUmbra1 * 0.30)  
        SET @Categoria = 'Proveedor Frecuente';  
    ELSE  
        SET @Categoria = 'Proveedor Regular';  
  
    RETURN @Categoria;  
END;
```

%

Messages

Commands completed successfully.

Completion time: 2024-10-19T10:24:31.4581923-05:00

Figura 1.16: Problema propuesto 3

```
SELECT dbo.CategorizarProveedor(100, 100000) AS CategoriaProveedor,  
       dbo.CategorizarProveedor(100000, 100000) AS CategoriaProveedor,  
       dbo.CategorizarProveedor(100001, 100000) AS CategoriaProveedor  
GO
```

Results Messages

CategoriaProveedor	CategoriaProveedor	CategoriaProveedor
Proveedor Regular	Proveedor Frecuente	Proveedor VIP

Figura 1.17: Problema propuesto 3 ejecución

## 1.2.4 Problema propuesto 4

```
USE DB_EmpresaXYZ
GO
CREATE PROCEDURE ReservarCodigosProductos
AS
BEGIN
    DECLARE @UltimoID INT, @NuevoID INT, @Contador INT;

    -- Obtener el último ID de producto registrado
    SELECT @UltimoID = MAX(Codigo_Producto) FROM TB_Producto;

    -- Inicializar contador
    SET @Contador = 1;

    -- Repetir hasta insertar 5 nuevos productos
    WHILE @Contador <= 5
    BEGIN
        -- Calcular el nuevo código de producto
        SET @NuevoID = @UltimoID + @Contador;

        -- Insertar el nuevo producto con valores por defecto
        INSERT INTO TB_Producto (Codigo_Producto, Descripcion_Producto, Cantidad_Inventario, Precio_Venta, Precio_Costo, Codigo_Proveedor)
        VALUES (@NuevoID, 'Nuevo Producto Por Definir', 0, 0, 0, 100);

        -- Incrementar el contador
        SET @Contador = @Contador + 1;
    END
END
```

Messages  
Commands completed successfully.

Figura 1.18: Problema propuesto 4

```
USE DB_EmpresaXYZ
GO
EXEC ReservarCodigosProductos
SELECT * FROM TB_Producto
GO
```

Codigo_Producto	Descripcion_Producto	Cantidad_Inventario	Precio_Venta	Precio_Costo	Codigo_Proveedor
100001	Equipo de Audio LG 400W	30	320.75	275.00	100
100002	Equipo de Audio LG 700W	25	650.75	520.00	103
100003	Consola Wii V.2012"	60	199.99	140.25	102
100004	NoteBook ACER 14" 2GHZ, 2GB RAM	15	399.99	315.68	103
100005	NoteBook ACER 14" 2GHZ, 4GB RAM	10	599.99	525.30	104
100006	Memoria USB 16GB Kingston	100	32.25	24.65	100
100007	Memoria USB 32GB Kingston	75	60.05	51.90	103
100008	Nuevo Producto Por Definir	0	0.00	0.00	100
100009	Nuevo Producto Por Definir	0	0.00	0.00	100
100010	Nuevo Producto Por Definir	0	0.00	0.00	100
100011	Nuevo Producto Por Definir	0	0.00	0.00	100
100012	Nuevo Producto Por Definir	0	0.00	0.00	100

Figura 1.19: Problema propuesto 4 ejecución