



# **“PAGINA WEB PARA ATENCION AL CLIENTE EN UNA DULCERIA”**

---

**PROYECTO DE TALLER DE BASE DE DATOS**

**INGENIERIA EN SISTEMAS COMPUTACIONALES**

**PRESENTA:**

**ISIDRO ARREDONDO CAMARENA  
EDGAR CORTÉS RESÉNDIZ**

**MARTINEZ GONZALEZ LEONARDO**

---

**JIQUILPAN, MICHOACÁN, SEPTIEMBRE DE 2024**

## Índice

Link de GitHub con las colecciones y la inserción de datos .....	3
Validación de Datos.....	3
Compras .....	3
Dulces .....	4
Elaboración.....	5
Fabricas .....	6
Materias Primas .....	8
Pedidos .....	9
Proveedores .....	10
Registros .....	11
Repartos .....	12
Sucursales.....	13
Trabajadores .....	15
Usuarios.....	17
Ventas.....	19
Modelado de datos .....	21
Proveedores-Compras.....	21
Materias Primas-Elaboración-Fabricas-Dulces-Trabajadores .....	23
Sucursales-trabajadores-registros-pedidos-repartos .....	25
Pedidos-dulces-usuarios .....	27
Información en colecciones .....	29
Compras .....	29
Dulces .....	29
Elaboración.....	30
Fabricas .....	31
Materias Primas .....	32
Pedidos .....	33
Proveedores .....	34
Registros.....	35
Repartos .....	35
Sucursales.....	36
Trabajadores .....	38

Usuarios.....	39
Ventas.....	40
Consultas generadoras de JSON inserción de datos .....	42
Compras .....	42
Dulces .....	42
Elaboración.....	43
Fabricas .....	43
Materias Primas .....	44
Pedidos .....	44
Proveedores .....	45
Registros.....	45
Repartos .....	45
Sucursales.....	46
Trabajadores .....	46
Usuarios.....	47
Ventas.....	47

# Link de GitHub con las colecciones y la inserción de datos

<https://github.com/sidral24/Dulceria>

## Validación de Datos

### **Compras**

//Usar la base de datos

use("Dulceria")

//Crear la conexion de usuarios

```
db.createCollection("compras", {
  validator: {
    $jsonSchema: {
      bsonType: 'object',
      title: 'Validacion del objeto Compras',
      required: ['id_mp', 'id_compra', 'id_proveedor', 'lote', 'cantidad', 'precio'],
      properties: {
        id_compra: { bsonType: 'int' },
        id_proveedor: { bsonType: 'int' },
        id_mp: { bsonType: 'int' },
        fecha_compra: { bsonType: 'date' },
        lote: { bsonType: 'string' },
        cantidad: { bsonType: 'string' },
        precio: { bsonType: 'int' },
        descuento: { bsonType: 'int' }
      }
    }
  }
})
```

```
}  
})
```

```
db.compras.createIndex(  
  { id_compra: 1 },  
  { unique: true }  
)
```

```
db.compras.drop()
```

## ***Dulces***

//Usar la base de datos

```
use("Dulceria")
```

//Crear la conexion de usuarios

```
db.createCollection("dulces", {  
  validator: {  
    $jsonSchema: {  
      bsonType: 'object',  
      title: 'Validacion del objeto Dulces',  
      required: ['id_dulce', 'nombre', 'precio_uni', 'lote'],  
      properties: {  
        id_dulce: { bsonType: 'int' },  
        nombre: { bsonType: 'string' },  
        precio_uni: { bsonType: 'int' },  
        stock: { bsonType: 'int' },  
        lote: { bsonType: 'string' },  
        fecha_cad: { bsonType: 'date' },
```

```

        reorden: { bsonType: 'int' },
        tipo: { enum: ['CON AZUCAR', 'SIN AZUCAR'] }
    }
}
}
})

```

```

db.dulces.createIndex(
    { id_dulce: 1 },
    { unique: true }
)

```

```

db.dulces.drop()

```

## ***Elaboración***

//Usar la base de datos

```

use("Dulceria")

```

//Crear la conexion de usuarios

```

db.createCollection("elaboracion", {
    validator: {
        $jsonSchema: {
            bsonType: 'object',
            title: 'Validacion del objeto Dulces',
            required: ['id_elaboracion', 'id_fabrica', 'id_trabajdor', 'id_mp', 'id_dulce',
'cantidad'],
            properties: {
                id_elaboracion: { bsonType: 'int' },

```

```

        id_fabrica: { bsonType: 'int' },
        id_trabajador: { bsonType: 'int' },
        id_mp: { bsonType: 'int' },
        id_dulce: { bsonType: 'int' },
        cantidad: { bsonType: 'int' }
    }
}
})

```

```

db.elaboracion.createIndex(
    { id_elaboracion: 1 },
    { unique: true }
)

```

```

db.elaboracion.drop()

```

## ***Fabricas***

//Usar la base de datos

```

use("Dulceria")

```

//Crear la conexion de usuarios

```

db.createCollection("Fabricas", {
    validator: {
        $jsonSchema: {
            bsonType: 'object',
            title: 'Validacion del objeto Dulces',

```

```

    required: ['id_fabrica', 'nom_responsable', 'estado', 'municipio', 'domicilio',
'colonia'],
    properties: {
      id_fabrica: { bsonType: 'int' },
      telefono: { bsonType: 'string' },
      nom_responsable: {
        bsonType: "object",
        required: ['nombre', 'apellido_parteno'],
        properties: {
          nombre: { bsonType: 'string' },
          apellido_paterno: { bsonType: 'string' },
          apellido_materno: { bsonType: 'string' }
        }
      },
      Direccion: {
        bsonType: "object",
        required: ['Calle', 'Numero', 'Colonia', 'cp'],
        properties: {
          Calle: { bsonType: 'string' },
          Numero: { bsonType: 'string' },
          Colonia: { bsonType: 'string' },
          Municipio: { bsonType: 'string' },
          Estado: { bsonType: 'string' },
          cp: { bsonType: 'int' }
        }
      }
    }
  }
}
}
}
}
}

```



```
}}
```

```
db.dulces.createIndex(  
  { nom_responsable: 1 },  
  { unique: true }  
)
```

```
db.dulces.drop()
```

### ***Materias Primas***

//Usar la base de datos

```
use("Dulceria")
```

//Crear la conexion de usuarios

```
db.createCollection("materias_primas", {  
  validator: {  
    $jsonSchema: {  
      bsonType: 'object',  
      title: 'Validacion del objeto Materias Primas',  
      required: ['id_mp', 'nombre', 'lote', 'fecha_cad', 'precio'],  
      properties: {  
        id_mp: { bsonType: 'int' },  
        nombre: { bsonType: 'string' },  
        tipo: { bsonType: 'string' },  
        lote: { bsonType: 'string' },  
        existencia: { bsonType: 'string' },  
        fecha_cad: { bsonType: 'date' },  
        precio: { bsonType: 'int' }
```

```

        }
    }
}
}))

db.materias_primas.createIndex(
    { id_mp: 1, nombre: 1 },
    { unique: true }
)

db.materias_primas.drop()

```

## ***Pedidos***

//Usar la base de datos

```
use("Dulceria")
```

//Crear la conexion de usuarios

```

db.createCollection("pedidos", {
    validator: {
        $jsonSchema: {
            bsonType: 'object',
            title: 'Validacion del objeto Compras',
            required: ['id_pedido', 'id_usuario', 'id_sucursal', 'id_dulce', 'cantidad'],
            properties: {
                id_pedido: { bsonType: 'int' },
                id_usuario: { bsonType: 'int' },
                id_sucursal: { bsonType: 'int' },
                id_dulce: { bsonType: 'int' },
            }
        }
    }
})

```

```

        cantidad: { bsonType: 'int' },
        fecha_pedido: { bsonType: 'date' },

    }

}

})

```

```

db.pedidos.createIndex(
    { id_pedido: 1 },
    { unique: true }
)

```

```

db.compras.drop()

```

## ***Proveedores***

```

use("Dulceria")

```

```

//crear la coleccion de usuarios

```

```

db.createCollection("Proveedores", {
    validator: {
        $jsonSchema: {
            bsonType: "object",
            title: 'validacion del objeto usuarios',
            required: ['id_proveedor', 'nombre', 'rfc', 'direccion'],
            properties: {
                id_proveedor: { bsonType: 'int' },
                nombre: { bsonType: 'string' },

```

```

        rfc: { bsonType: 'string' },
        direccion: { bsonType: "string" },
        telefono: {
            bsonType: 'array',
            minItems: 1, //Especificar el minimo requerido
            uniqueItems: true, //No telefonos repetidos
            items: { bsonType: 'string' }
        }
    }
}
}
})
db.Proveedores.createIndex(
    { id_proveedor: 1 },
    { unique: true }
)

```

## ***Registros***

//Usar la base de datos

```
use("Dulceria")
```

//Crear la conexion de usuarios

```

db.createCollection("registros", {
    validator: {
        $jsonSchema: {
            bsonType: 'object',
            title: 'Validacion del objeto Registros',

```

```

        required: ['id_registro', 'id_sucursal', 'id_dulce', 'precio_uni',
'fecha_recepcion'],
        properties: {
            id_registro: { bsonType: 'int' },
            id_sucursal: { bsonType: 'int' },
            id_dulce: { bsonType: 'int' },
            cantidad: { bsonType: 'int' },
            fecha_recepcion: { bsonType: 'date' },
            precio_uni: { bsonType: 'int' }

        }
    }
}
}))

```

```

db.registros.createIndex(
    { id_registro: 1 },
    { unique: true }
)

```

## ***Repartos***

```

use("Dulceria")

```

```

//Crear la conexion de usuarios

```

```

db.createCollection("repartos", {
    validator: {
        $jsonSchema: {

```

```

    bsonType: 'object',
    title: 'Validacion del objeto repartos',
    required: ['id_fabrica', 'id_repartos', 'id_sucursal', 'id_trabajador', 'id_dulce'],
    properties: {
      id_repartos: { bsonType: 'int' },
      id_fabrica: { bsonType: 'int' },
      id_sucursal: { bsonType: 'int' },
      id_trabajador: { bsonType: 'int' },
      id_dulce: { bsonType: 'int' },
      cantidad: { bsonType: 'int' },
      fecha_reparto: { bsonType: 'date' }
    }
  }
}
})

```

```

db.repartos.createIndex(
  { id_repartos: 1 },
  { unique: true }
)

```

```

db.repartos.drop()

```

## ***Sucursales***

```

use("Dulceria")

```

```

//Crear la conexion de usuarios

```

```

db.createCollection("sucursales", {

```

```

validator: {
  $jsonSchema: {
    bsonType: 'object',
    title: 'Validacion del objeto sucursales',
    required: ['id_sucursal', 'telefono'],
    properties: {
      id_sucursal: { bsonType: 'int' },
      nom_responsable: {
        bsonType: "object",
        required: ['nombre', 'apellido_paterno'],
        properties: {
          nombre: { bsonType: 'string' },
          apellido_paterno: { bsonType: 'string' },
          apellido_materno: { bsonType: 'string' }
        }
      },
      telefono: {
        bsonType: 'array',
        minItems: 1, //Especificar el minimo requerido
        uniqueItems: true, //No telefonos repetidos
        items: { bsonType: 'string' }
      },
      direccion: {
        bsonType: "object",
        title: 'Validación del objeto Address',
        required: ['estado', 'municipio', 'colonia', 'calle', 'numero', 'cp'],
        properties: {
          calle: { bsonType: 'string' },

```

```

        numero: { bsonType: 'string' },
        colonia: { bsonType: 'string' },
        municipio: { bsonType: 'string' },
        estado: { bsonType: 'string' },
        cp: { bsonType: 'int' },
    }
},
}
}
}
})

```

```

db.repartos.createIndex(
    { id_sucursal: 1 },
    { unique: true }
)

```

```

db.repartos.drop()

```

## ***Trabajadores***

```

use("Dulceria")

```

```

//crear la coleccion de usuarios

```

```

db.createCollection("trabajadores", {
    validator: {
        $jsonSchema: {
            bsonType: "object",
            title: 'validacion del objeto usuarios',

```



```

required: ['id_trabajador', 'id_sucursal', 'nombre', 'rfc'],
properties: {
  id_trabajador: { bsonType: 'int' },
  id_sucursal: { bsonType: 'int' },
  nombre: {
    bsonType: "object",
    title: 'Validación del objeto Address',
    required: ['apellido_paterno', 'nombres'],
    properties: {
      apellido_paterno: { bsonType: 'string' },
      apellido_materno: { bsonType: 'string' },
      nombres: { bsonType: 'string' },
    }
  },
  fechaNac: { bsonType: 'date' },
  fechaIngreso: { bsonType: 'date' },
  rfc: { bsonType: 'string' },
  tipo: { bsonType: 'string' },
  telefono: {
    bsonType: 'array',
    minItems: 1, //Especificar el minimo requerido
    uniqueItems: true, //No telefonos repetidos
    items: { bsonType: 'string' }
  }
}
}
}
})

```

```
db.Trabajadores.createIndex(  
  {  
    id_sucursal: 1,  
    id_trabajador: 1  
  },  
  { unique: true }  
)
```

## ***Usuarios***

```
//USAR LA BASE DE DATOS  
use("Dulceria")
```

```
//crear la coleccion de usuarios  
db.createCollection("usuarios", {  
  validator: {  
    $jsonSchema: {  
      bsonType: "object",  
      title: 'validacion del objeto usuarios',  
      required: ['email', 'id_usuario'],  
      properties: {  
        id_usuario: {  
          bsonType: 'int'  
        },  
        nombre: {  
          bsonType: "object",  
          title: 'Validación del objeto nombre',
```

```

    required: ['apellido_paterno', 'nombres'],
    properties: {
      apellido_paterno: { bsonType: 'string' },
      apellido_materno: { bsonType: 'string' },
      nombres: { bsonType: 'string' },
    }
  },
  email: {
    bsonType: 'string',
    pattern: "^[A-Za-z\\d.-]{1,}@[A-Za-z\\d.-]{1,}[.][A-Za-z\\d.-]{1,}$"
  },
  telefono: {
    bsonType: 'array',
    minItems: 1, //Especificar el minimo requerido
    uniqueItems: true, //No telefonos repetidos
    items: { bsonType: 'string' }
  }
}
}
})

```

```

db.usuarios.createIndex(
  {
    email: 1,
    idUsuario: 1
  },
  { unique: true }
)

```

)

## ***Ventas***

use("Dulceria")

//Crear la conexion de usuarios

```
db.createCollection("ventas", {  
  validator: {  
    $jsonSchema: {  
      bsonType: 'object',  
      title: 'Validacion del objeto Ventas',  
      required: ['id_venta', 'id_usuario', 'id_sucursal', 'id_dulce', 'cantidad'],  
      properties: {  
        id_venta: { bsonType: 'int' },  
        id_usuario: { bsonType: 'int' },  
        id_sucursal: { bsonType: 'int' },  
        id_dulce: { bsonType: 'int' },  
        cantidad: { bsonType: 'int' },  
        fecha: { bsonType: 'date' },  
        precio_uni: { bsonType: 'int' },  
        descuento: { bsonType: 'int' }  
      }  
    }  
  }  
})
```

db.ventas.createIndex(

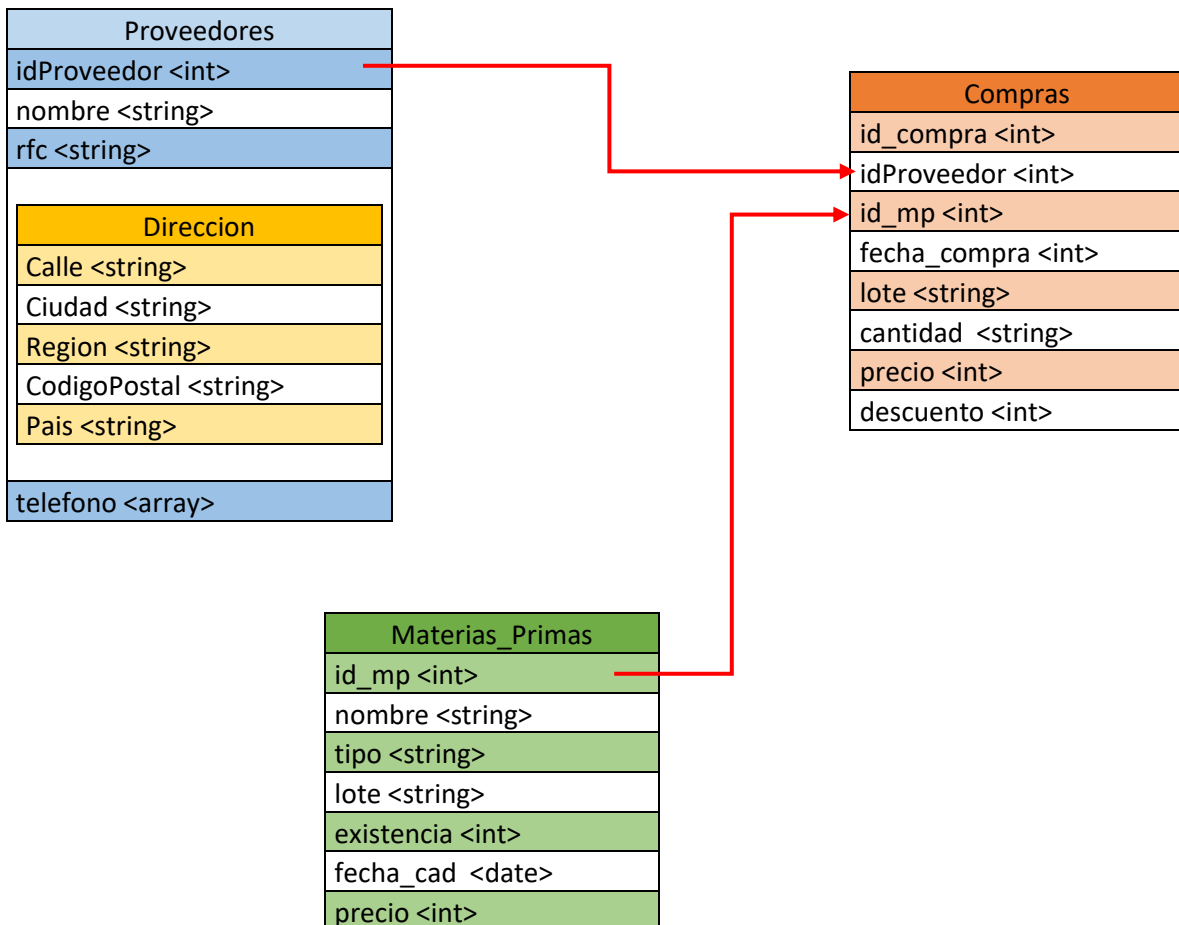
```
{ id_venta: 1 },  
  { unique: true }  
)
```

```
db.ventas.drop()
```

## Modelado de datos

### ***Proveedores-Compras***

Se hizo una relación de N:N en cada una de estas tablas, esto con la finalidad de que como los datos son independientes y si existe un cambio constante en los registros de estos mismos, es necesario que las tablas sean referenciadas de esta manera para así poder mantener un flujo de datos mas eficiente sin alterar tanto las colecciones y que sea menos eficiente.

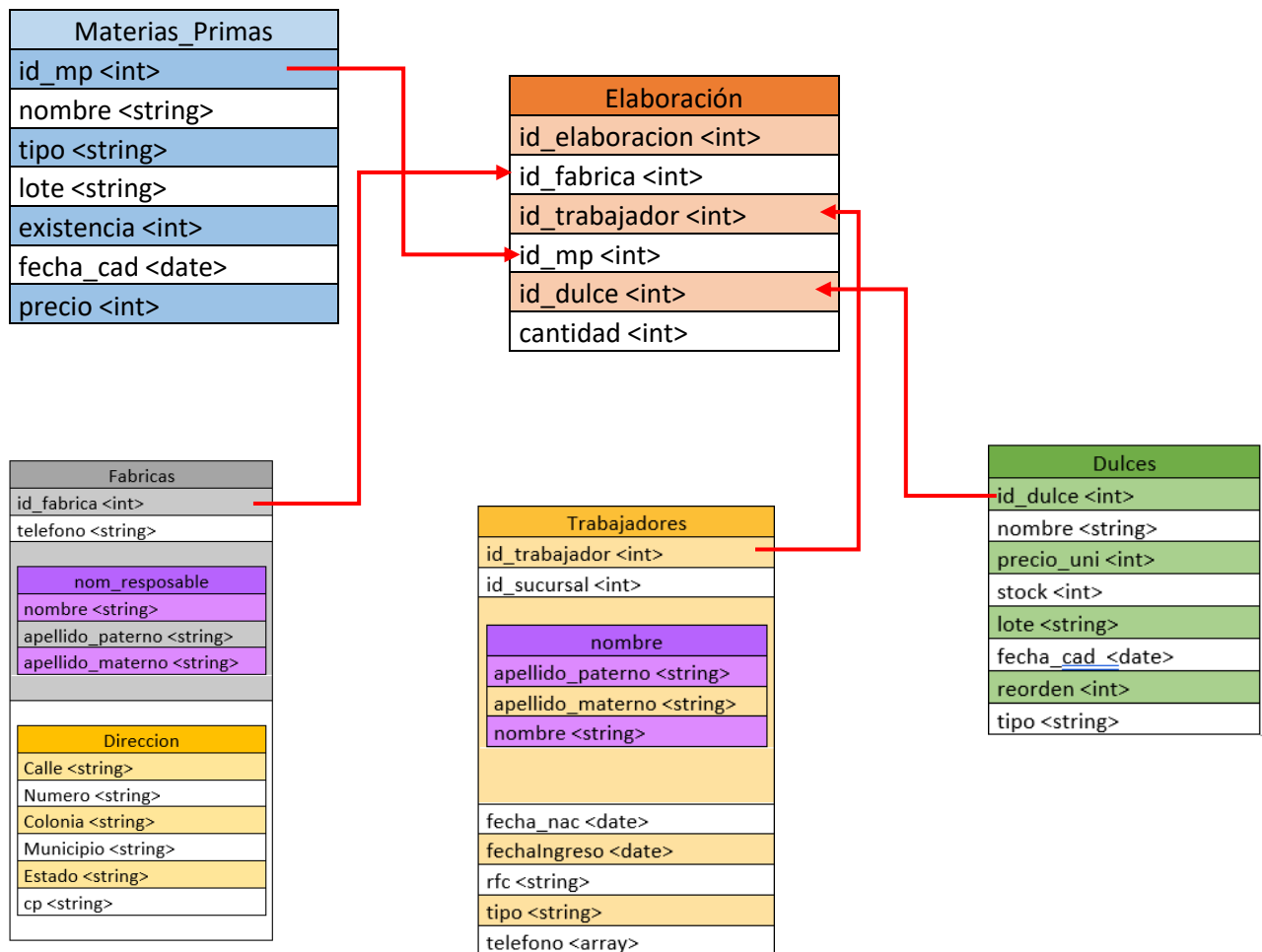


Por la parte de la tabla de proveedores decidimos embeber la dirección, ya que en este la dirección de los proveedores es muy difícil de que cambie frecuentemente, así que de esta manera nos evitamos de crear una colección exclusivamente para esta sección, además de que en las consultas necesitamos encontrar la dirección de manera mas eficiente así que de esta manera podemos adquirir esta información sin tiempo de búsqueda tan extensos.

Proveedores						
idProveedor <int>						
nombre <string>						
rfc <string>						
<table> <tr> <th>Direccion</th></tr> <tr> <td>Calle &lt;string&gt;</td></tr> <tr> <td>Ciudad &lt;string&gt;</td></tr> <tr> <td>Region &lt;string&gt;</td></tr> <tr> <td>CodigoPostal &lt;string&gt;</td></tr> <tr> <td>Pais &lt;string&gt;</td></tr> </table>	Direccion	Calle <string>	Ciudad <string>	Region <string>	CodigoPostal <string>	Pais <string>
Direccion						
Calle <string>						
Ciudad <string>						
Region <string>						
CodigoPostal <string>						
Pais <string>						
telefono <array>						

## Materias Primas-Elaboración-Fabrics-Dulces-Trabajadores

Para en esta sección de relaciones podemos observar que de la misma manera que la anterior podemos apreciar que las relaciones entre tablas se siguen manteniendo ya que en este apartado los datos son muy susceptibles a cambios, así que de esta manera nos evitamos tiempos de actualización de las colecciones que tienen otras colecciones embebidas, además de que por la parte de las relaciones podemos ver que la que lleva más carga de información y de cambios es la tabla de elaboración y esto demuestra que no podemos embeber aquí las tablas.



Además de que podemos ver que dos tablas tienen embebida unos campos, esto lo hicimos porque por ejemplo en los campos que son de nombre es necesario poder buscar por apellidos o por nombre y de esta manera podemos encontrarlos de una manera más eficiente, además de que es un campo que es necesario que este embebido ya que podemos obtener este bloque de información más rápido que si



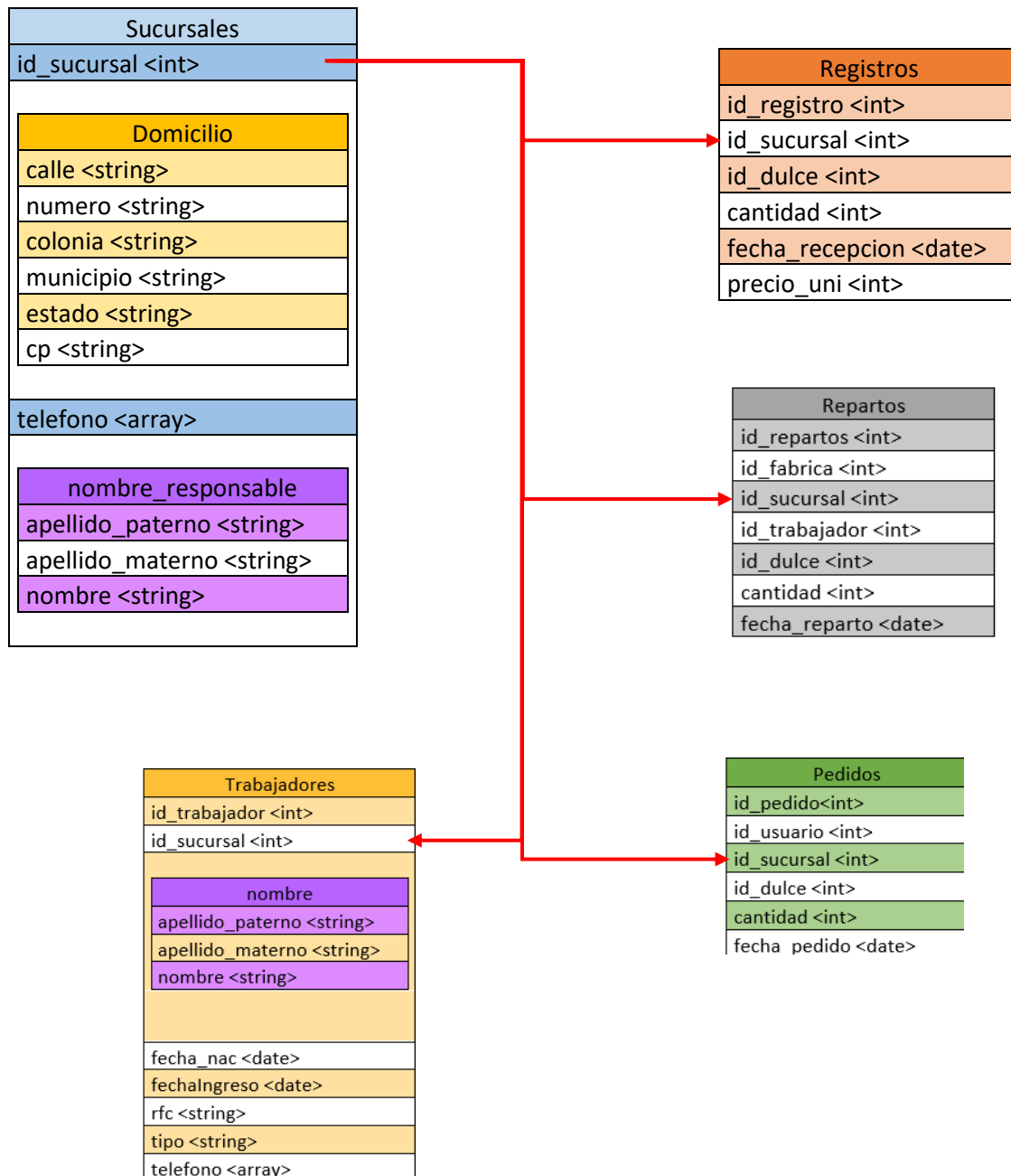
fuera por separado o en un solo campo. Por otro lado, podemos ver el dato embebido de “Dirección” que este nos ayuda a conocer donde tenemos las fábricas, este lo pusimos de igual manera embebido, porque consideramos que los datos de dirección en este aspecto de fabricas o de establecimientos no se consideran muy susceptibles a cambios y esto nos ayuda a una mejor integridad de los datos.

Fabricas
id_fabrica <int>
telefono <string>
nom_resposable
nombre <string>
apellido_paterno <string>
apellido_materno <string>
Direccion
Calle <string>
Numero <string>
Colonia <string>
Municipio <string>
Estado <string>
cp <string>

Trabajadores
id_trabajador <int>
id_sucursal <int>
nombre
apellido_paterno <string>
apellido_materno <string>
nombre <string>
fecha_nac <date>
fechaIngreso <date>
rfc <string>
tipo <string>
telefono <array>

## Sucursales-trabajadores-registros-pedidos-repartos

En esta sección de la misma manera que en las anteriores podemos observar que las relaciones las hicimos de manera relacional, esto creando una mejor optimización y eficiencia en la actualización o incluso en el grabado de datos.



Por otro lado, podemos ver que tenemos dos tablas que tienen datos embebidos esto lo hicimos de esta manera, porque consideramos que esos datos son difícilmente de que se modifiquen en un corto periodo de tiempo o que estén en constante cambio.

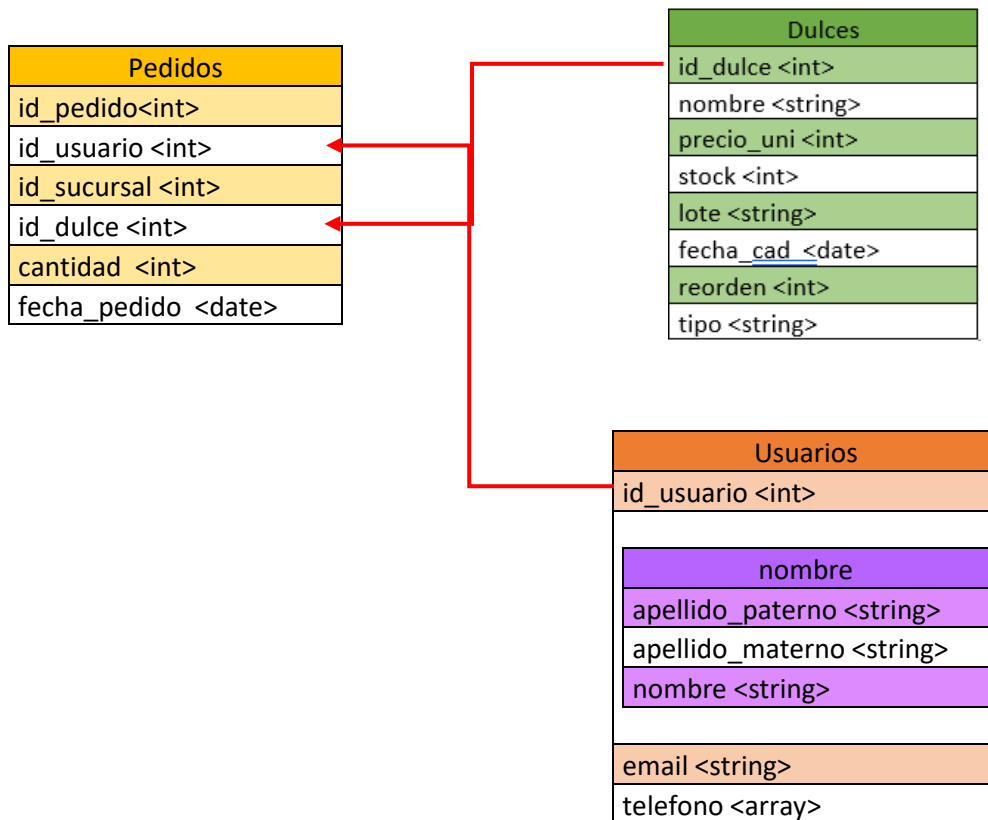
Sucursales							
id_sucursal <int>							
<table> <tr> <th>Domicilio</th></tr> <tr> <td>calle &lt;string&gt;</td></tr> <tr> <td>numero &lt;string&gt;</td></tr> <tr> <td>colonia &lt;string&gt;</td></tr> <tr> <td>municipio &lt;string&gt;</td></tr> <tr> <td>estado &lt;string&gt;</td></tr> <tr> <td>cp &lt;string&gt;</td></tr> </table>	Domicilio	calle <string>	numero <string>	colonia <string>	municipio <string>	estado <string>	cp <string>
Domicilio							
calle <string>							
numero <string>							
colonia <string>							
municipio <string>							
estado <string>							
cp <string>							
telefono <array>							
<table> <tr> <td>nombre_responsable</td></tr> <tr> <td>apellido_paterno &lt;string&gt;</td></tr> <tr> <td>apellido_materno &lt;string&gt;</td></tr> <tr> <td>nombre &lt;string&gt;</td></tr> </table>	nombre_responsable	apellido_paterno <string>	apellido_materno <string>	nombre <string>			
nombre_responsable							
apellido_paterno <string>							
apellido_materno <string>							
nombre <string>							

Trabajadores				
id_trabajador <int>				
id_sucursal <int>				
<table> <tr> <td>nombre</td></tr> <tr> <td>apellido_paterno &lt;string&gt;</td></tr> <tr> <td>apellido_materno &lt;string&gt;</td></tr> <tr> <td>nombre &lt;string&gt;</td></tr> </table>	nombre	apellido_paterno <string>	apellido_materno <string>	nombre <string>
nombre				
apellido_paterno <string>				
apellido_materno <string>				
nombre <string>				
fecha_nac <date>				
fechaIngreso <date>				
rfc <string>				
tipo <string>				
telefono <array>				

## ***Pedidos-dulces-usuarios***

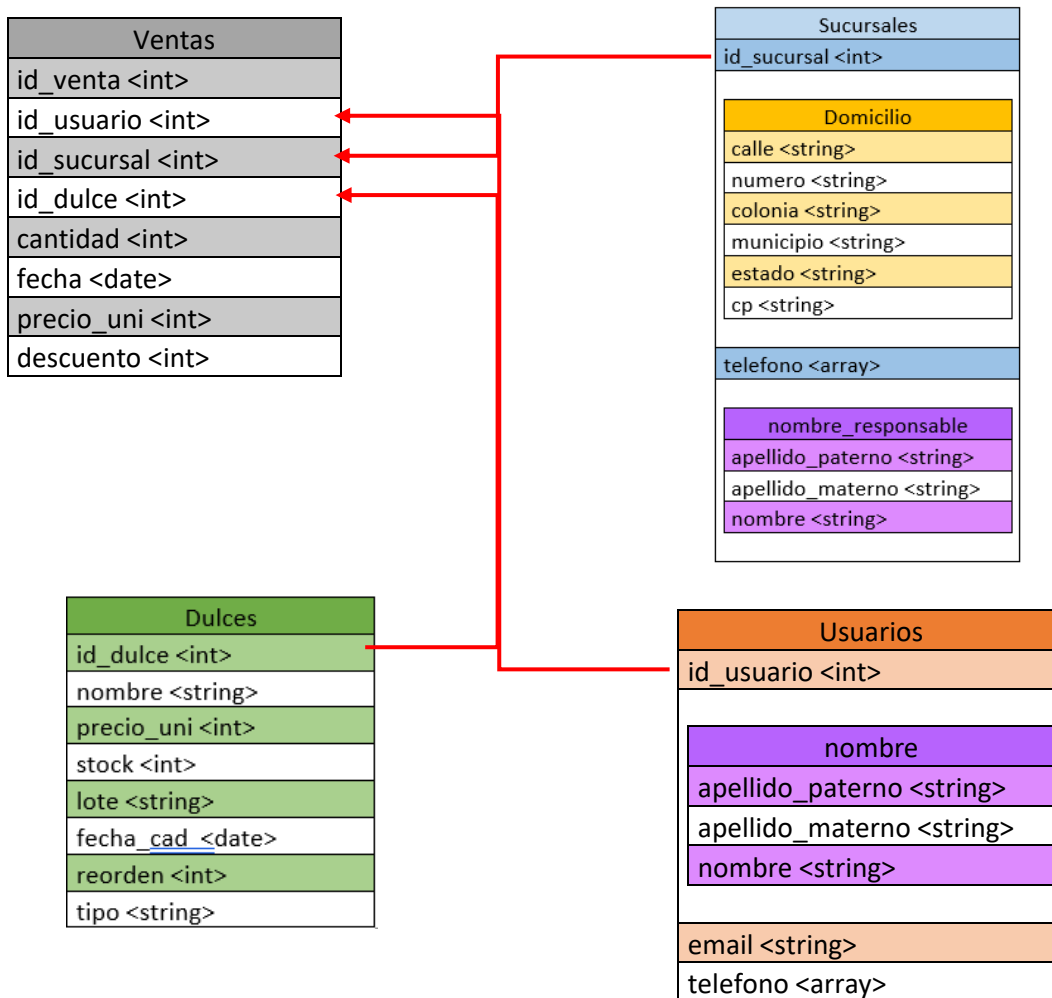
Para estas tablas la hicimos relacional porque es mejor para los datos ya que estos pueden estar constante cambio, así que decidimos dejarlas relacionadas.

Además de que en la tabla usuarios embebimos los datos del nombre, esto lo realizamos así porque el nombre de uno de nuestros clientes no puede cambiar con frecuencia, aunque si puede cambiar por si hubo algún error en la inserción de sus datos, pero esto no puede ser muy común.



## Ventas-Dulces-Sucursales-Usuarios

Para estas relaciones podemos observar que la tabla de ventas es un pivote o una tabla que guarda todas las claves de las demás tablas, esto ya que la tabla de ventas almacena todos los datos generados de una compra del usuario, esto generando una venta hacia nuestro sistema así que en esta tabla se prefirió guardar una estructura relacional ya que los datos están en constante movimiento y esto produce que si existe alguna corrección en los datos podamos modificarlos sin mayor problema.



## Información en colecciones

### **Compras**

```
use("Dulceria")
```

```
//insertar datos
```

```
db.compras.insertMany(  
  [  
    {  
      "id_compra": 1,  
      "id_proveedor": 1,  
      "id_mp": 1,  
      "fecha_compra": new Date("2024-01-15"),  
      "lote": "L1234",  
      "cantidad": "500 kg",  
      "precio": 10000,  
      "descuento": 5  
    },.....  
  ]  
)
```

### **Dulces**

```
use("Dulceria")
```

```
//insertar datos
```

```
db.dulces.insertMany(  
  [  
    {  
      "id_dulce": 1,  
      "nombre": "Ate de Membrillo",  
    },  
  ]  
)
```

```
"precio_uni": 50.0000,  
"stock": 100,  
"lote": "Lote001",  
"fecha_cad": new Date("2025-12-31"),  
"reorden": 20,  
"tipo": "CON AZUCAR"  
},.....
```

### ***Elaboración***

use("Dulceria")

//insertar datos

db.elaboracion.insertMany(  
[

```
{  
  "id_elaboracion": 1,  
  "id_fabrica": 1,  
  "id_trabajador": 1,  
  "id_mp": 1,  
  "id_dulce": 1,  
  "cantidad": 200
```

```
},
```

```
{  
  "id_elaboracion": 2,  
  "id_fabrica": 1,  
  "id_trabajador": 2,  
  "id_mp": 2,  
  "id_dulce": 2,
```

```
"cantidad": 150
},.....
```

### ***Fabricas***

```
use("Dulceria")
```

```
//insertar datos
```

```
db.fabricas.insertMany(
[
  {
    "id_fabrica": 1,
    "telefono": "5551234567",
    "nom_responsable": "Juan Pérez",
    "domicilio": {
      "estado": "Ciudad de México",
      "municipio": "Iztapalapa",
      "colonia": "Col. Dulce",
      "domicilio": "Calle Caramelo 123",
      "cp": 9876
    }
  },
  {
    "id_fabrica": 2,
    "telefono": "5557654321",
    "nom_responsable": "María Gómez",
    "domicilio": {
      "estado": "Jalisco",
      "municipio": "Guadalajara",
```



```
        "colonia": "Col. Golosina",
        "domicilio": "Avenida Azúcar 456",
        "cp": 44100
    }
},.....
```

### ***Materias Primas***

use("Dulceria")

//insertar datos

```
db.materias_primas.insertMany(
[
    {
        "id_mp": 1,
        "nombre": "Azúcar Refinada",
        "tipo": "Edulcorante",
        "lote": "L1234",
        "existencia": "1000 kg",
        "fecha_cad": new Date("2025-12-31"),
        "precio": 20
    },
    {
        "id_mp": 2,
        "nombre": "Cacao en Polvo",
        "tipo": "Saborizante",
        "lote": "L5678",
        "existencia": "500 kg",
        "fecha_cad": new Date("2024-11-15"),
```

```
    "precio": 45
  },....
```

## ***Pedidos***

```
use("Dulceria")
```

```
//insertar datos
```

```
db.pedidos.insertMany(
```

```
  [
    {
      "id_pedido": 2,
      "id_usuario": 1,
      "id_sucursal": 1,
      "id_dulce": 1,
      "cantidad": 5,
      "fecha_pedido": new Date("2024-10-01")
    },
    {
      "id_pedido": 3,
      "id_usuario": 2,
      "id_sucursal": 1,
      "id_dulce": 2,
      "cantidad": 10,
      "fecha_pedido": new Date("2024-10-02")
    },....
```

## ***Proveedores***

```
use("Dulceria")
```

```
//insertar datos
```

```
db.Proveedores.insertMany(
```

```
[
```

```
{
```

```
  "id_proveedor": 1,
```

```
  "nombre": "Azúcares del Valle",
```

```
  "rfc": "AVR123456789",
```

```
  "direccion": "Calle Caramelo 123, Ciudad Dulce",
```

```
  "telefono": [
```

```
    "555-123-4567"
```

```
  ]
```

```
},
```

```
{
```

```
  "id_proveedor": 2,
```

```
  "nombre": "Esencias y Sabores S.A.",
```

```
  "rfc": "ESS234567890",
```

```
  "direccion": "Av. Vainilla 234, Colonia Sabrosa",
```

```
  "telefono": [
```

```
    "555-234-5678"
```

```
  ]
```

```
},.....
```

## **Registros**

```
use("Dulceria")
```

```
//insertar datos
```

```
db.registros.insertMany(
```

```
[  
  {  
    "id_registro": 1,  
    "id_sucursal": 1,  
    "id_dulce": 1,  
    "cantidad": 50,  
    "fecha_recepcion": new Date("2024-10-01"),  
    "precio_uni": 45.0000  
  },  
  {  
    "id_registro": 2,  
    "id_sucursal": 1,  
    "id_dulce": 2,  
    "cantidad": 30,  
    "fecha_recepcion": new Date("2024-10-02"),  
    "precio_uni": 55.0000  
  },.....  
]
```

## **Repartos**

```
use("Dulceria")
```

```
//insertar datos
```

```
db.repartos.insertMany(
```

```
[
  {
    "id_repartos": 2,
    "id_fabrica": 1,
    "id_sucursal": 1,
    "id_trabajador": 1,
    "id_dulce": 1,
    "cantidad": 100,
    "fecha_reparto": new Date("2024-10-01")
  },
  {
    "id_repartos": 3,
    "id_fabrica": 1,
    "id_sucursal": 2,
    "id_trabajador": 2,
    "id_dulce": 2,
    "cantidad": 150,
    "fecha_reparto": new Date("2024-10-02")
  },.....
]
```

## ***Sucursales***

```
use("Dulceria")
```

```
//insertar datos
```

```
db.sucursales.insertMany(
```

```
[
  {
    "id_sucursal": 1,
```

```
"nom_responsable": "Pedro Sánchez",
"telefono": [
  "5551234567"
],
"direccion": {
  "estado": "Ciudad de México",
  "municipio": "Coyoacán",
  "colonia": "Col. Alegre",
  "domicilio": "Avenida Azucarera 101",
  "cp": 4360
}
},
{
  "id_sucursal": 2,
  "nom_responsable": "María López",
  "telefono": [
    "5552345678"
  ],
  "direccion": {
    "estado": "Jalisco",
    "municipio": "Guadalajara",
    "colonia": "Col. Dulce",
    "domicilio": "Calle de la Chocolatería 202",
    "cp": 44100
  }
},.....
```

## **Trabajadores**

```
use("Dulceria")
```

```
//insertar datos
```

```
db.trabajadores.insertMany(
```

```
[
  {
    "id_trabajador": 1,
    "id_sucursal": 1,
    "nombre": {
      "apellido_paterno": "Sánchez",
      "apellido_materno": "Gómez",
      "nombres": "Pedro"
    },
    "fecha_nacimiento": "1990-05-15",
    "fecha_ingreso": "2022-01-10",
    "rfc": "SAGP900515HDFMPR01",
    "tipo": "Permanente",
    "telefono": [
      "5551234567"
    ]
  },
  {
    "id_trabajador": 2,
    "id_sucursal": 1,
    "nombre": {
      "apellido_paterno": "López",
      "apellido_materno": "Martínez",
      "nombres": "María"
    }
  }
]
```

```

    },
    "fecha_nacimiento": "1985-08-22",
    "fecha_ingreso": "2021-03-15",
    "rfc": "LOMM850822HDFMTR02",
    "tipo": "Permanente",
    "telefono": [
        "5552345678"
    ]
},.....

```

## ***Usuarios***

```
use("Dulceria")
```

```
//insertar datos
```

```

db.usuarios.insertMany(
    [
        {
            "id_usuario": 1,
            "nombre": {
                "apellido_paterno": "Sánchez",
                "apellido_materno": "Gómez",
                "nombres": "Pedro"
            },
            "email": "pedro.sanchez@example.com",
            "telefono": [
                "5551234567"
            ]
        },
    ]
),

```



```

{
  "id_usuario": 2,
  "nombre": {
    "apellido_paterno": "López",
    "apellido_materno": "Martínez",
    "nombres": "María"
  },
  "email": "maria.lopez@example.com",
  "telefono": [
    "5552345678"
  ]
},.....

```

## **Ventas**

```
use("Dulceria")
```

```
//insertar datos
```

```

db.ventas.insertMany(
[
  {
    "id_venta": 1,
    "id_usuario": 1,
    "id_sucursal": 1,
    "id_dulce": 1,
    "cantidad": 2,
    "fecha": new Date("2024-10-01"),
    "precio_uni": 50.0000,
    "descuento": 5
  }
]
)

```

```
},  
{  
  "id_venta": 2,  
  "id_usuario": 2,  
  "id_sucursal": 1,  
  "id_dulce": 2,  
  "cantidad": 5,  
  "fecha": new Date("2024-10-02"),  
  "precio_uni": 60.0000,  
  "descuento": 10  
},.....
```

## Consultas generadoras de JSON inserción de datos

### ***Compras***

```
SELECT id_compra,  
       id_proveedor,  
       id_mp,  
       fecha_compra,  
       lote,  
       cantidad,  
       precio,  
       descuento  
FROM compras  
FOR JSON PATH
```

### ***Dulces***

```
SELECT id_dulce,  
       nombre,  
       precio_uni,  
       stock,  
       lote,  
       fecha_cad,  
       reorden,  
       tipo  
FROM dulces  
FOR JSON PATH
```

## ***Elaboración***

```
SELECT id_elaboracion,  
       id_fabrica,  
       id_trabajador,  
       id_mp,  
       id_dulce,  
       cantidad  
FROM elaboracion  
FOR JSON PATH
```

## ***Fabricas***

```
SELECT id_fabrica  
       ,telefono  
       ,nom_responsable  
       ,estado as [domicilio.estado]  
       ,municipio as [domicilio.municipio]  
       ,colonia as [domicilio.colonia]  
       ,domicilio as [domicilio.domicilio]  
       ,cp as [domicilio.cp]  
FROM fabricas  
FOR JSON PATH
```

## ***Materias Primas***

```
SELECT id_mp
      ,nombre
      ,tipo
      ,lote
      ,existencia
      ,fecha_cad
      ,precio
FROM materias_primas
FOR JSON PATH
```

## ***Pedidos***

```
SELECT id_pedido
      ,id_usuario
      ,id_sucursal
      ,id_dulce
      ,cantidad
      ,fecha_pedido
FROM pedidos
FOR JSON PATH
```

## ***Proveedores***

```
SELECT id_proveedor
      ,nombre
      ,rfc
      ,direccion
      ,JSON_ARRAY(telefono) AS telefono
FROM proveedores
FOR JSON PATH
```

## ***Registros***

```
SELECT id_registro
      ,id_sucursal
      ,id_dulce
      ,cantidad
      ,fecha_recepcion
      ,precio_uni
FROM registros
FOR JSON PATH
```

## ***Repartos***

```
SELECT id_repartos
      ,id_fabrica
      ,id_sucursal
      ,id_trabajador
```

```
,id_dulce
,cantidad
,fecha_reparto
FROM repartos
FOR JSON PATH
```

### ***Sucursales***

```
SELECT id_sucursal
      ,nom_responsable
      ,JSON_ARRAY(telefono) as telefono
      ,estado as [direccion.estado]
      ,municipio as [direccion.municipio]
      ,colonia as [direccion.colonia]
      ,domicilio as [direccion.domicilio]
      ,cp as [direccion.cp]
FROM sucursales
for JSON PATH
```

### ***Trabajadores***

```
SELECT id_trabajador
      ,id_sucursal
      ,apellido_paterno as [nombre.apellido_paterno]
      ,apellido_materno as [nombre.apellido_materno]
      ,nombres as [nombre.nombres]
      ,fecha_nacimiento
```

```
,fecha_ingreso
,rfc
,tipo
,JSON_ARRAY(telefono) as telefono
FROM trabajadores
for JSON PATH
```

## ***Usuarios***

```
SELECT id_usuario
,apellido_paterno as [nombre.apellido_paterno]
,apellido_materno as [nombre.apellido_materno]
,nombres as [nombre.nombres]
,email
,JSON_ARRAY(telefono) as telefono
FROM usuarios
FOR JSON PATH
```

## ***Ventas***

```
SELECT id_venta
,id_usuario
,id_sucursal
,id_dulce
,cantidad
,fecha
,precio_uni
,descuento
```



FROM ventas  
FOR JSON PATH