

# JSON EN ORACLE

---

# JSON en ORACLE

---

- ❑ A partir de la versión Oracle Database 12.1.0.2 es posible trabajar con el formato JSON dentro de nuestra base de datos relacional Oracle.
- ❑ A partir de la 21c se hay incorporado un tipo nativo JSON dentro de la Base de Datos.
- ❑ En versiones anteriores podemos guardar JSON en VARCHAR2, CLOB o BLOB
- ❑ Tenemos una serie de funciones y características para trabajar con JSON

# JSON en ORACLE

---

❑ Desde de la 12c y antes de la 21c

```
CREATE TABLE productos (  
  codigo INT,  
  nombre VARCHAR2(200),  
  datos BLOB  
);
```

```
CREATE TABLE productos (  
  codigo INT,  
  nombre VARCHAR2(200),  
  datos CLOB  
);
```

```
CREATE TABLE productos (  
  codigo INT,  
  nombre VARCHAR2(200),  
  datos VARCHAR2(4000)  
);
```

# JSON en ORACLE

---

❑ Desde de la 21c

```
CREATE TABLE productos (  
    codigo INT,  
    nombre VARCHAR2(200),  
    datos JSON  
);
```

# JSON en ORACLE

---

- ❑ SQL. Formas de acceder a los datos JSON
  - ❑ Hay múltiples funciones que podemos usar:
    - ❑ json\_value
    - ❑ json\_query
    - ❑ json\_table
    - ❑ Json\_exists
    - ❑ is json / is not json
    - ❑ json\_transform
  - ❑ Con la notación por punto

```
SELECT prod.productos.datos FROM productos prod;
```

# JSON en ORACLE

---

## ☐ PL/SQL.

- ☐ Se usan en general las mismas funciones que en SQL
- ☐ También se pueden usar objetos PL/SQL para gestionar los datos JSON:
  - ☐ JSON\_ELEMENT\_T : Super Tipo del que heredan algunos de los otros objetos no se suele utilizar directamente
  - ☐ JSON\_OBJECT\_T : un objeto que representa un documento JSON.
  - ☐ JSON\_ARRAY\_T : Un objeto que representa un array JSON.
  - ☐ JSON\_SCALAR\_T : un valor escalar asociado a una clave. puede ser un una una cadena, un número,boolean or NULL.
  - ☐ JSON\_KEY\_LIST : Un array de claves.

# JSON en ORACLE

---

## □ Ejemplos

**-- Trabajar con un objeto JSON**

```
json1 := JSON_OBJECT_T('{“codigo_emple”:9999}');
```

```
json1.put('nombre', 'pepe');
```

```
Json1.remove(“nombre”);
```

```
json1.put('telefonos', JSON_ARRAY_T(['1111111","22222222","3333333"]));
```

**-- Trabajar con un array JSON**

```
array1 := JSON_ARRAY_T(['rojo', " amarillo", " verde"]);
```

```
array1.append('azul');
```