

# Lenguaje de Marcas

# JSON

# ¿Qué es JSON?

**JSON** – JavaScript Object Notation

- Formato de texto independiente del lenguaje de programación.
- Se utiliza par el intercambio de datos.
- **No** es un **lenguaje de programación**.

# Elementos

se basa en la estructuración de una colección de **pares** con **nombre** y **valor** que contienen:

- Una *clave*: que corresponde al identificador del contenido.
- Un *valor*: que representa el contenido correspondiente.

```
{  
  "nombre": "Juan",  
  "Apellido": "Pérez",  
}
```

# Sintaxis

- Utiliza pares clave-valor:
  - × clave (**key**): cadenas de texto
  - × valor (**value**): puede ser de diferentes tipos de datos.
- Los pares **clave-valor** se separan por dos puntos (:), y los **elementos** se separan por comas (,).
- Los **objetos** JSON se encierran entre llaves {} y los **arrays** se encierran entre corchetes [].

# Sintaxis

- **Número:**

```
{  
    "edad":30,  
}
```

- **Cadena de texto:**

```
{  
    "nombre":"Juan",  
}
```

- **Booleano:**

```
{  
    "esEstudiante":true,  
}
```

- **Nulo:**

```
{  
    "comentario":null,  
}
```

# Sintaxis

- **Objeto**: conjuntos de datos separados por comas (,) que se identifican por ir entre llaves { }(anidamiento):

```
{  
  "persona": {  
    "nombre": "María",  
    "Apellido": "González",  
  }  
}
```

# Sintaxis

- **Array**: conjunto de objetos, también separados por comas y que se identifica por ir entre corchetes [ ](lista de valores):

```
{  
  "numeros":[1,2,3,4,5],  
}
```

# Sintaxis

- **Anidamiento de objetos y arrays:**

```
{  
  "alumno": {  
    "nombre": "Ana",  
    "notas": [8.5, 9.2, 7, 8],  
  }  
}
```



# Sintaxis

- No admite comentarios y se deben omitir en los archivos JSON válidos.
- Es importante separar los elementos con comas (,) en objetos JSON y en arrays JSON, excepto para el último elemento en la lista.
- Las cadenas de texto siempre deben ir entre comillas dobles ("), y las claves también deben estar entre comillas dobles en un objeto JSON
- JSON es sensible a mayúsculas y minúsculas,
- Los espacios en blanco, como espacios y retornos de carro, son ignorados en JSON.

# Sintaxis

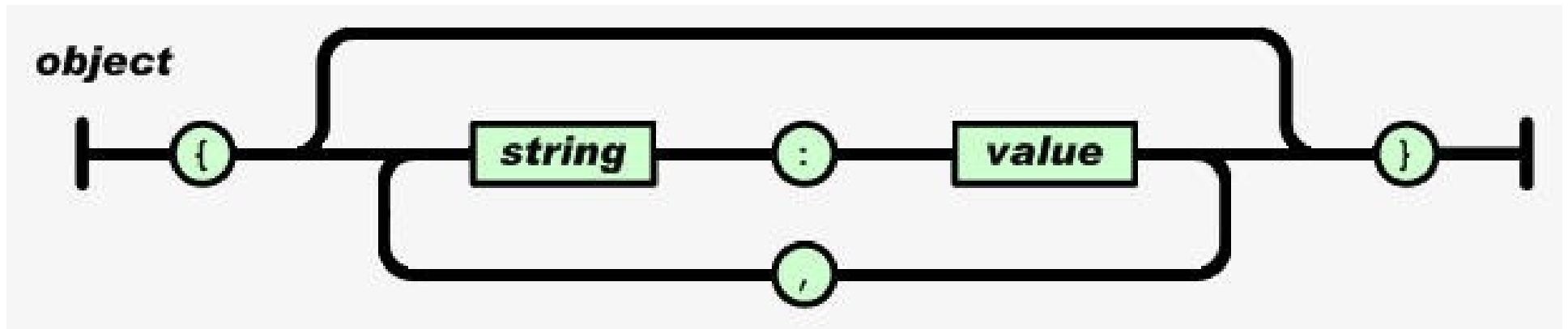
```
{  
  "arrayColores":[  
    {  
      "nombreColor":"rojo",  
      "valorHexadec":"#f00"  
    },  
    {  
      "nombreColor":"verde",  
      "valorHexadec":"#0f0"  
    },  
    {  
      "nombreColor":"azul",  
      "valorHexadec":"#00f"  
    },  
    {  
      "nombreColor":"cyan",  
      "valorHexadec":"#0ff"  
    },  
    {  
      "nombreColor":"magenta",  
      "valorHexadec":"#f0f"  
    },  
    {  
      "nombreColor":"negro",  
      "valorHexadec":"#000"  
    }  
  ]  
}
```

```
{  
  "arrayColores":[{  
    "rojo":"#f00",  
    "verde":"#0f0",  
    "azul":"#00f",  
    "cyan":"#0ff",  
    "magenta":"#f0f",  
    "amarillo":"#ff0",  
    "negro":"#000"  
  }  
]
```

```
{  
  "rojo":"#f00",  
  "verde":"#0f0",  
  "azul":"#00f",  
  "cyan":"#0ff",  
  "magenta":"#f0f",  
  "amarillo":"#ff0",  
  "negro":"#000"  
}
```

# Sintaxis

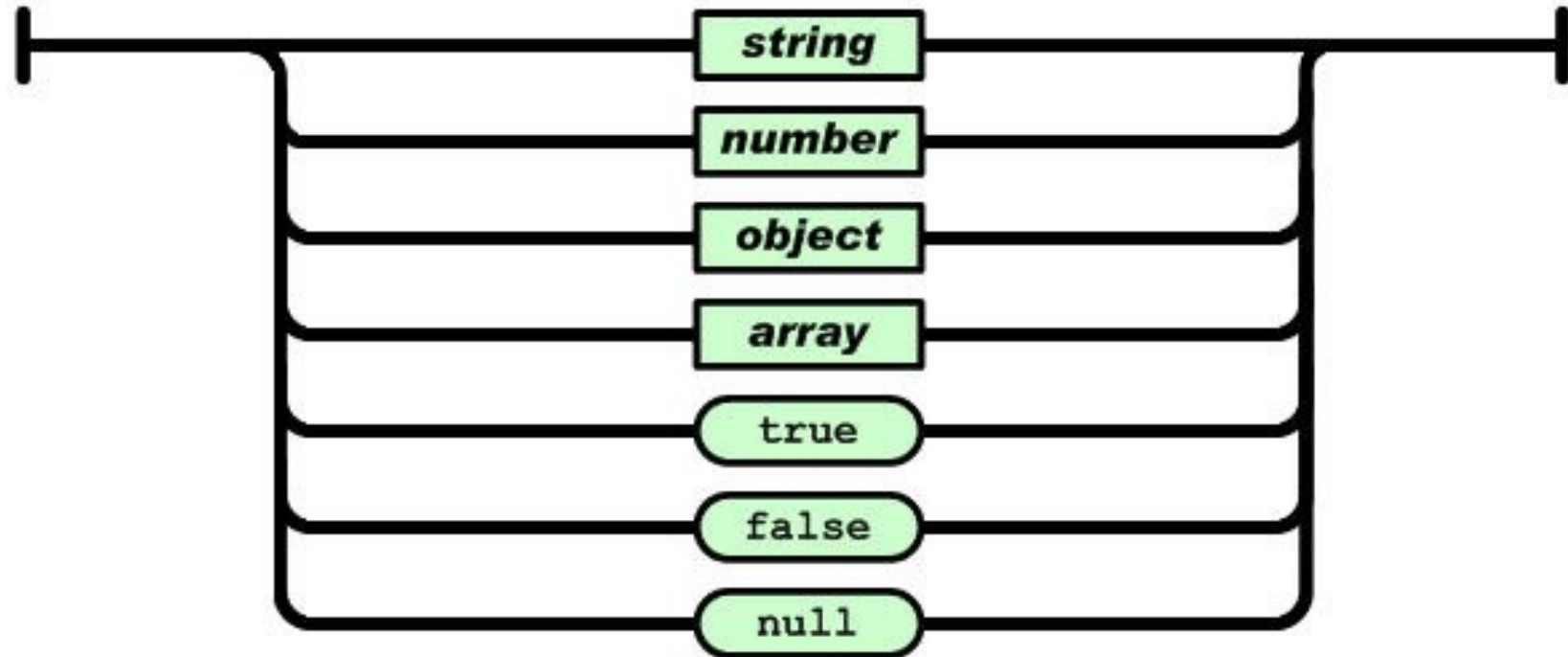
## Objeto JSON



# Sintaxis

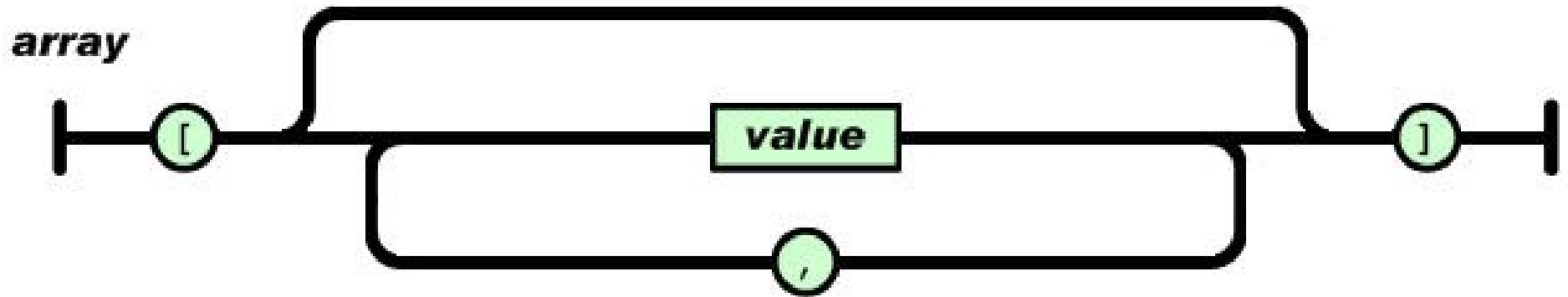
## Valor JSON

*value*



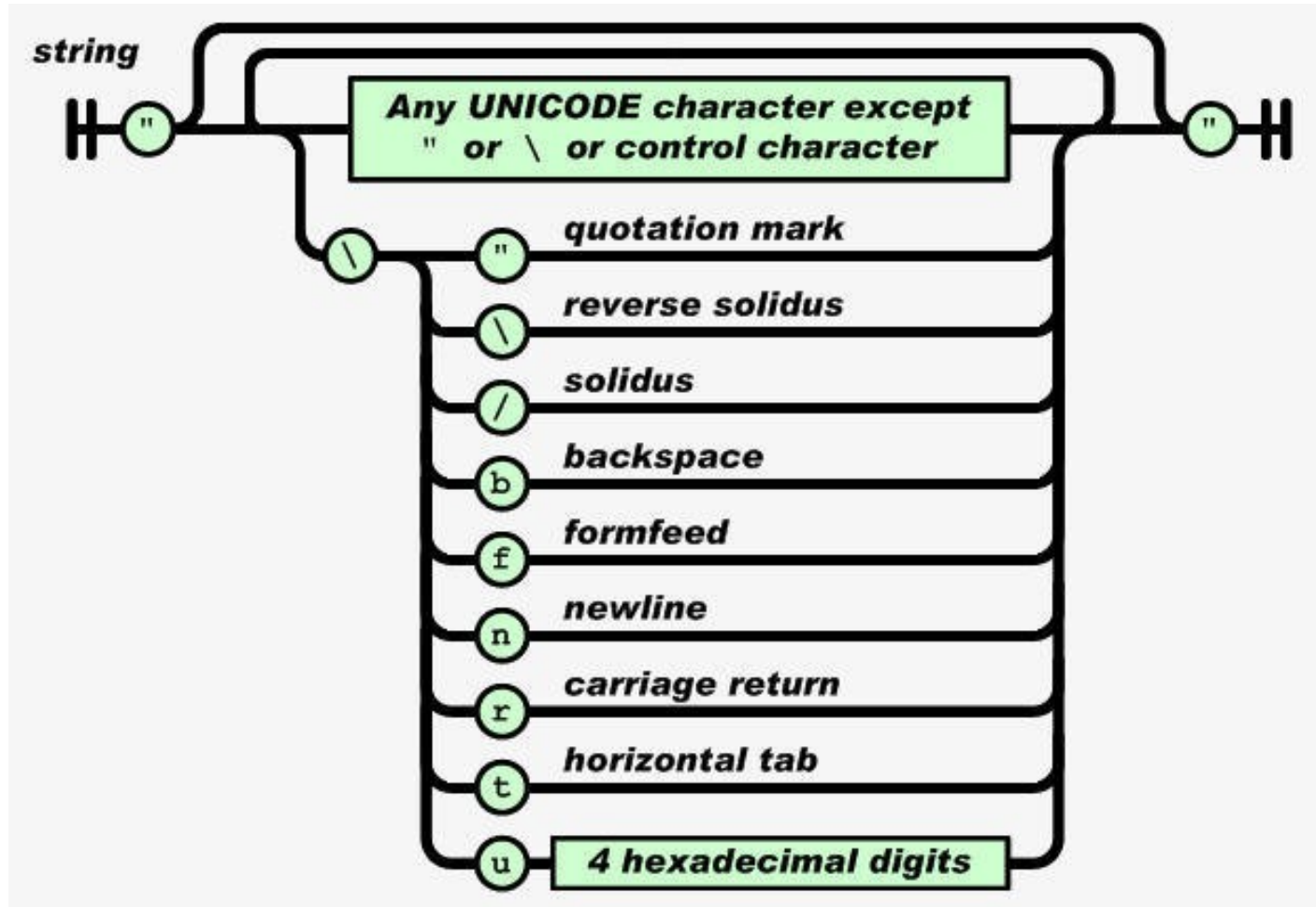
# Sintaxis

## Array JSON



# Sintaxis

## Cadena JSON



## Sintaxis

# Número JSON

