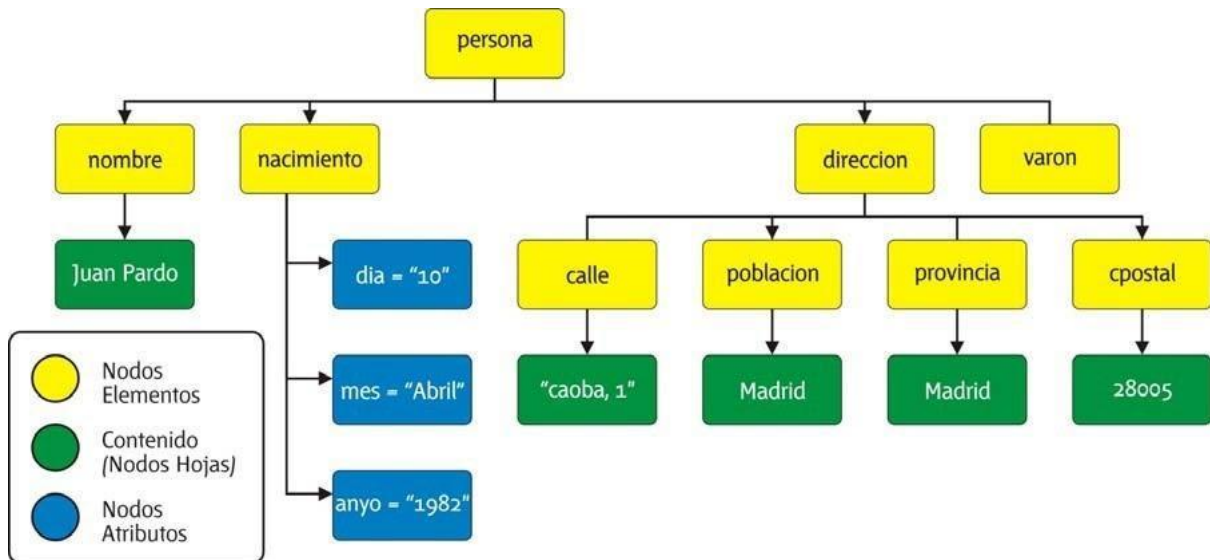


Ejercicio 1

Hacer una JSON que pueda contener una lista de personas, cada persona está formada por los siguientes elementos. **Hacer un ejemplo con 2 personas**



```
[
{
  "persona": {
    "nombre": "Enrique Sainz",
    "nacimiento": {
      "dia": "24",
      "mes": "Diciembre",
      "año": "1993"
    },
    "direccion": {
      "calle": "Califato de Cordoba, 1",
      "poblacion": "Alcazar de San Juan",
      "provincia": "Ciudad Real",
      "cpostal": "13600"
    },
    "varon": true
  }
}
```

```
}  
  
},  
  
{  
  
  "persona": {  
  
    "nombre": "Ana López",  
  
    "nacimiento": {  
  
      "dia": "25",  
  
      "mes": "Marzo",  
  
      "anyo": "1990"  
  
    },  
  
    "direccion": {  
  
      "calle": "Sol, 20",  
  
      "poblacion": "Sevilla",  
  
      "provincia": "Sevilla",  
  
      "cpostal": "41001"  
  
    },  
  
    "varon": false  
  
  }  
  
}  
  
]
```

To format and validate your JSON, just copy + paste it below:

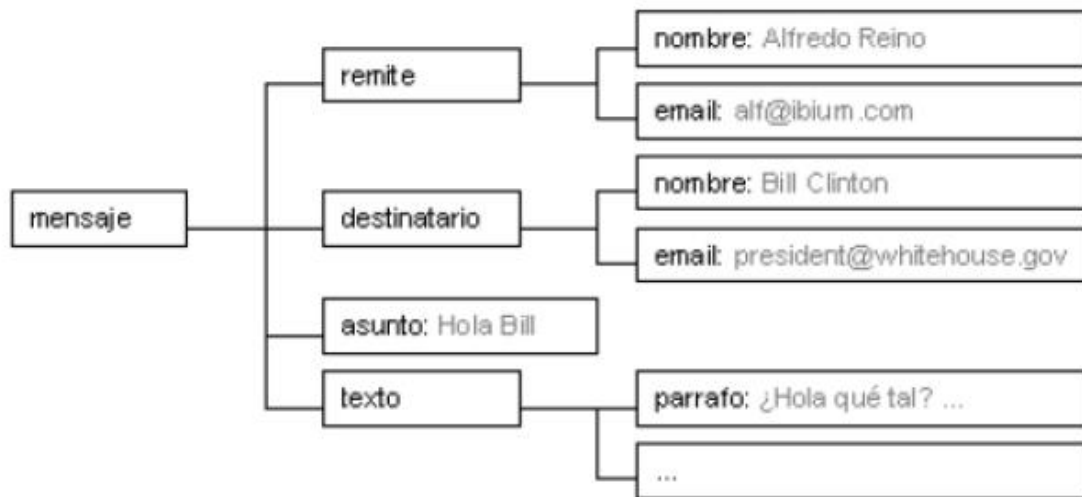
```
1 {  
2 {  
3   "persona": {  
4     "nombre": "Ana López",  
5     "nacimiento": {  
6       "dia": "25",  
7       "mes": "Marzo",  
8       "anyo": "1990"  
9     },  
10    "direccion": {  
11      "calle": "Sol, 20",  
12      "poblacion": "Sevilla",  
13      "provincia": "Sevilla",  
14      "cpostal": "41001"  
15    },  
16    "varon": false  
17  },  
18 },  
19 {  
20   "persona": {  
21     "nombre": "Ana López",  
22     "nacimiento": {  
23       "dia": "25",  
24       "mes": "Marzo",  
25       "anyo": "1990"  
26     },  
27     "direccion": {  
28       "calle": "Sol, 20",  
29       "poblacion": "Sevilla",  
30       "provincia": "Sevilla",  
31       "cpostal": "41001"  
32     },  
33     "varon": false  
34   },  
35 },  
36 }  
37 }  
38 }
```

[Validate JSON](#) [Clear](#) [Compress](#)

JSON is valid!

Ejercicio 2

Una aplicación de mensajería quiere contener cada mensaje en un JSON. Cada mensaje tiene estos campos



Haz un JSON con un mensaje de ejemplo

```
{  
  "mensaje": {  
    "remite": {  
      "nombre": "Alfredo Reino",  
      "email": "alf@ibium.com"  
    },  
    "destinatario": {  
      "nombre": "Bill Clinton",  
      "email": "president@whitehouse.gov"  
    },  
    "asunto": "Hola Bill",  
    "texto": [  
      { "parrafo": "¿Hola qué tal? ..." },  
      { "parrafo": "Espero tu respuesta." }  
    ]  
  }  
}
```

```

1+ {
2+   "mensaje": {
3+     "remite": {
4+       "nombre": "Vicente Rossi",
5+       "email": "vic@adultra.com"
6+     },
7+     "destinatario": {
8+       "nombre": "Juan Villalba",
9+       "email": "juv@adultra.com.ar"
10+    },
11+    "asunto": "Hola Juan!",
12+    "texto": [
13+      {
14+        "parrafo": "¡Hola me llamo ...!"
15+      },
16+      {
17+        "parrafo": "Espero te encuentres bien"
18+      }
19+    ]
20+  }
21+ }

```

Validate JSON

Clear

Compress

JSON is valid!

Ejercicio 3 (Pedidos)

Genera el JSON necesario para una app de compras que debe enviar la información de **cada pedido** al servidor en un archivo JSON. Para cada pedido

se debe generar un archivo JSON que contendrá:

- Un número de pedido, la fecha (AAAA-MM-DD) y la hora del pedido (hh:mm:ss)
- Un elemento usuario que contendrá, el nombre de usuario, el nombre completo y la dirección de envío.
- La dirección de envío a su vez es un elemento que está compuesto por la calle, la población y el código postal.
- A continuación irá una lista de artículos. Para cada artículo se incluirá el código de artículo, una descripción y el precio.

Haz un ejemplo de archivo para **un pedido de tres artículos**

```
{
  "pedido": {
    "numero_pedido": "PED001",
    "fecha": "2025-05-04",
    "hora": "19:40:00",
    "usuario": {
      "nombre_usuario": "cliente123",
      "nombre_completo": "Carlos Perez",
      "direccion_envio": {
        "calle": "Avda. Constitución, 5",
        "poblacion": "Alcázar de San Juan",
        "codigo_postal": "13600"
      }
    }
  }
}
```

```

    }
  },
  "articulos": [
    {
      "codigo_articulo": "ART001",
      "descripcion": "Teclado inalámbrico",
      "precio": 45.99
    },
    {
      "codigo_articulo": "ART002",
      "descripcion": "Ratón óptico",
      "precio": 25.50
    },
    {
      "codigo_articulo": "ART003",
      "descripcion": "Monitor 24 pulgadas",
      "precio": 189.95
    }
  ]
}

```

```

1 {
2   "pedido": {
3     "numero_pedido": "123456",
4     "fecha": "2023-10-26",
5     "hora": "10:30 AM",
6     "usuario": {
7       "nombre_usuario": "johndoe123",
8       "nombre_completo": "John Doe",
9       "direccion_envio": {
10        "calle": "Calle Constitución 45",
11        "poblacion": "Mérida de Iruya",
12        "codigo_postal": "10000"
13      }
14    },
15    "articulos": [
16      {
17        "codigo_articulo": "ART001",
18        "descripcion": "Teclado inalámbrico",
19        "precio": 45.99
20      },
21      {
22        "codigo_articulo": "ART002",
23        "descripcion": "Ratón óptico",
24        "precio": 25.50
25      },
26      {
27        "codigo_articulo": "ART003",
28        "descripcion": "Monitor 24 pulgadas",
29        "precio": 189.95
30      }
31    ]
32  }
33 }

```

Validate JSON Clear Compress

JSON is valid!

Ejercicio 4 (Concesionarios)

Se desea modelar en JSON la información relativa a distintos concesionarios de vehículos de una marca. La lista de concesionarios debe tener al menos uno, no puede estar vacía. Deberá poder recoger la información de contacto de cada concesionario (nombre, dirección, localidad y uno o más teléfonos) así como la lista de los vehículos que tiene a la venta. La lista de vehículos puede estar vacía, o contener 1 o varios vehículos diferentes. Cada vehículo se identifica

mediante su matrícula y pueden ser del tipo “turismo” o “furgoneta”, siendo el

valor por defecto “turismo”. Del vehículo se recoge el modelo, cilindrada, tipo de combustible (sólo puede ser Gasolina o Diesel), color, y puede contener un elemento “asegurado” que es un booleano.

Hacer un ejemplo con 2 concesionarios con 2 y 3 vehículos respectivamente

```
{
  "concesionarios": [
    {
      "nombre": "AutoMancha",
      "direccion": "Calle Industria, 10",
      "localidad": "Ciudad Real",
      "telefonos": ["926111222", "600111222"],
      "vehiculos": [
        {
          "matricula": "1234 ABC",
          "tipo": "turismo",
          "modelo": "Ford Mondeo",
          "cilindrada": 1998,
          "combustible": "Gasolina",
          "color": "Rojo",
          "asegurado": true
        },
        {
          "matricula": "5678 DEF",
          "tipo": "furgoneta",
          "modelo": "Berlingo",
          "cilindrada": 2499,
          "combustible": "Diesel",
          "color": "Blanco"
        }
      ]
    }
  ]
}
```

```
},
{
  "nombre": "MotorQuijote",
  "direccion": "Polígono La Estrella, Parcela 5",
  "localidad": "Toledo",
  "telefonos": ["925333444"],
  "vehiculos": [
    {
      "matricula": "9020 CDZ",
      "modelo": "Velsatis",
      "cilindrada": 1598,
      "combustible": "Gasolina",
      "color": "Azul",
      "asegurado": false
    },
    {
      "matricula": "5454 KJF",
      "tipo": "turismo",
      "modelo": "Audi a4",
      "cilindrada": 2199,
      "combustible": "Diesel",
      "color": "Gris"
    },
    {
      "matricula": "7886 HHY",
      "modelo": "Audi A6",
      "cilindrada": 1998,
      "combustible": "Gasolina",
      "color": "Negro",
      "asegurado": true
    }
  ]
}
```

```
1- {
2-   "concesionarios": [
3-     {
4-       "nombre": "Videoclub",
5-       "direccion": "Calle de la Independencia 10",
6-       "localidad": "Madrid",
7-       "telefonos": [
8-         "912345678",
9-         "912345679"
10-      ],
11-       "vehiculos": [
12-         {
13-           "matricula": "1234 ABC",
14-           "tipo": "carro",
15-           "modelo": "Ford Focus",
16-           "cilindrada": 1600,
17-           "combustible": "gasolina",
18-           "color": "rojo",
19-           "asegurado": true
20-         },
21-         {
22-           "matricula": "5678 DEF",
23-           "tipo": "camioneta",
24-           "modelo": "Nissan X-Trail",
25-           "cilindrada": 2000,
26-           "combustible": "gasolina",
27-           "color": "negro"
28-         }
29-       ]
30-     }
31-   ]
32- }
```

Validate JSON Clear Compress

JSON is valid!

Ejemplo 5 (Videoclub)

Una cadena de videoclubs quiere emplear una base de datos para almacenar información referente a las facturas que se hacen a los clientes. Esta información es la siguiente:

- En un mismo documento se puede guardar información de varias facturas.
- Cada factura está formada por dos tipos de información: datos de cliente y datos del ticket de factura propiamente dichos.
- De los datos del cliente se desea guardar: su nombre, su primer y segundo apellidos, DNI y teléfono (uno o varios). Además, como características del cliente, se desea conocer el identificador de cliente.
- En cada factura habrá alquileres, compras o los dos. Incluye la forma de pago y el importe total.
- Los alquileres se realizan de películas. El alquiler de películas lleva asociada una fecha de devolución que es común a todas las películas alquiladas en la misma factura.
- De cada película se quiere conocer su título, género, duración y los nombres y apellidos de dos actores que participan en ella. Existen dos atributos que definen a las películas: idPelícula y valoración.
- Además, opcionalmente se puede guardar para cada película el nombre de un archivo con la imagen de la carátula en formato jpg.
- Con respecto a las compras, puedes ser de DVDs o cintas de video pero no los dos a la vez.
- De los DVDs interesa el título del DVD, la fecha de salida al mercado y si viene o no con extras (aunque es opcional)
- De las cintas de video se guardará el título, el formato (VHS, por ejemplo) y si está rebobinada o no.

Hacer un ejemplo con dos facturas, una para un cliente por el alquiler de dos películas y otro de la compra de un DVD


```
{
  "facturas": [
    {
      "factura": {
        "cliente": {
          "id_cliente": "CLI001",
          "nombre": "Pedro",
          "apellido1": "García",
          "apellido2": "Martínez",
          "DNI": "11111111A",
          "telefonos": ["911223344", "666778899"]
        },
        "ticket": {
          "alquileres": {
            "fecha_devolucion": "2025-05-11",
            "películas": [
              {
                "película": {
                  "idPelícula": "PEL001",
                  "valoracion": "4.5",
                  "titulo": "El Gran Hotel",
                  "genero": "Comedia",
                  "duracion": "120 min",
                  "actores": [
                    {"nombre": "Ricardo", "apellidos": "Darín"},
                    {"nombre": "Belén", "apellidos": "Rueda"}
                  ],
                  "caratula": "gran_hotel.jpg"
                }
              }
            ],
            "película": {
              "idPelícula": "PEL002",
              "valoracion": "5.0",
              "titulo": "Misión Imposible",
              "genero": "Acción",
            }
          }
        }
      }
    }
  ]
}
```

```
    "duracion": "145 min",
    "actores": [
      {"nombre": "Tom", "apellidos": "Cruise"},
      {"nombre": "Rebecca", "apellidos": "Ferguson"}
    ]
  }
}
],
"forma_pago": "Tarjeta",
"importe_total": 8.50
}
},
{
  "factura": {
    "cliente": {
      "id_cliente": "CLI002",
      "nombre": "Luisa",
      "apellido1": "Fernández",
      "apellido2": "Sánchez",
      "DNI": "22222222B",
      "telefonos": ["934455667"]
    },
    "ticket": {
      "compras": {
        "dvds": [
          {
            "dvd": {
              "titulo": "Matrix Revolutions",
              "fecha_salida": "2003-11-05",
              "extras": true
            }
          }
        ]
      }
    ]
  },
}
```

```
    "forma_pago": "Efectivo",
    "importe_total": 15.95
  }
}
}
]
```

```
1+ {
2+   "facturas": [
3+     {
4+       "factura": {
5+         "cliente": {
6+           "id_cliente": 100000,
7+           "nombre": "Juan",
8+           "apellido1": "Garcia",
9+           "apellido2": "Gonzalez",
10+          "dni": "12345678",
11+          "telefonos": [
12+            "987654321",
13+            "876543210"
14+          ],
15+        },
16+        "ticket": {
17+          "squilexes": {
18+            "fecha_devolucion": "2020-10-10",
19+            "peliculas": [
20+              {
21+                "pelicula": {
22+                  "id_pelicula": 100000,
23+                  "valoracion": 9.5,
24+                  "titulo": "El gran hotel",
25+                  "genero": "Drama",
26+                  "duracion": "120 min",
27+                  "actores": [
28+                    "Juan",
29+                    "Garcia",
30+                    "Gonzalez"
31+                  ]
32+                }
33+              }
34+            ]
35+          }
36+        }
37+      }
38+    ]
39+  }
```

Validate JSONClearCompress

JSON is valid!