Exercise 1: Convert the following dictionary into JSON format

```
data = {"key1" : "value1", "key2" : "value2"}
```

Expected Output:

```
data = {"key1" : "value1", "key2" : "value2"}
```

.

Exercise 2: Access the value of key2 from the following JSON

```
import json

sampleJson = """{"key1": "value1", "key2": "value2"}"""
# write code to print the value of key2
```

Expected Output:

```
value2
```

.

Question 3: PrettyPrint following JSON data

PrettyPrint following JSON data with **indent level 2** and key-value **separators** should be `(",", " = ")`.

```
sampleJson = {"key1": "value1", "key2": "value2"}
```

Expected Output:

```
{
```

```
  "key1" = "value2",

  "key2" = "value2",

  "key3" = "value3"

}
```

.

Exercise 4: Sort JSON keys in and write them into a file

Sort following JSON data alphabetical order of keys

```
sampleJson = {"id" : 1, "name" : "value2", "age" : 29}
```

Expected Output:

```
{

    "age": 29,

    "id": 1,

    "name": "value2"

}
```

.

Question 5: Access the nested key ‘salary’ from the following JSON

```
import json

sampleJson = """{
   "company":{
      "employee":{
         "name":"emma",
```

```
          "payble":{
             "salary":7000,
             "bonus":800
          }
       }
    }
}"""

# write code to print the value of salary
```

Expected Output:

```
7000
```

.

Exercise 6: Convert the following Vehicle Object into JSON

```
import json

class Vehicle:
    def __init__(self, name, engine, price):
        self.name = name
        self.engine = engine
        self.price = price

vehicle = Vehicle("Toyota Rav4", "2.5L", 32000)

# Convert it into JSON format
```

Expected Output:

```
{

    "name": "Toyota Rav4",

    "engine": "2.5L",

    "price": 32000

}
```

.

Exercise 7: Convert the following JSON into Vehicle Object

```
{ "name": "Toyota Rav4", "engine": "2.5L", "price": 32000 }
```

For example, we should able to access Vehicle Object using the dot operator like this.

```
vehicleObj.name, vehicleObj.engine, vehicleObj.price
```

.

Question 8: Check whether following json is valid or invalid. If Invalid correct it

```
{
   "company":{
      "employee":{
         "name":"emma",
         "payble":{
            "salary":7000
            "bonus":800
         }
      }
   }
}
```

.

Exercise 9: Parse the following JSON to get all the values of a key ‘name’ within an array

```
[
   {
      "id":1,
      "name":"name1",
      "color":[
         "red",
         "green"
      ]
   },
   {
      "id":2,
      "name":"name2",
      "color":[
         "pink",
         "yellow"
      ]
   }
]
```

Expected Output:

`["name1", "name2"]`