**Challenge-3:**

**Question:** We have a nested object. We would like a function where you pass in the object and a key and get back the value.

The choice of language and implementation is up to you.

**Example Inputs**
object = {“a”:{“b”:{“c”:”d”}}}
key = a/b/c
object = {“x”:{“y”:{“z”:”a”}}}
key = x/y/z
value = a

Answer/Output-1: **Example code snippet of a nested object function in Python that takes an object and a key as input parameter variable.**

Prog. Language : **Python**

## Brief summary:

The below code uses the Python programming language to generate a nested object function which have the inputs of objects and key.

The code uses **get_value_from_nested_object** function which takes an object and a key as input. It splits the key using the / delimiter to get individual keys. Then, it iterates through the keys, accessing each level of the nested object until it reaches the desired value. If a key doesn't exist or a value is not subscriptable, it returns None.

In the below example usage, we have two objects (object_1 and object_2) and their corresponding keys (key_1 and key_2). The function extracts the values based on the keys, and the results are printed as output.

**Python Code:**

```
def get_value_from_nested_object(obj, key):

    keys = key.split("/")

    value = obj



    try:

        for k in keys:

            value = value[k]

        return value

    except (KeyError, TypeError):

        return None
```

```
# Example usage: object-1
object_1 = {"a": {"b": {"c": "d"}}}
key_1 = "a/b/c"

result_1 = get_value_from_nested_object(object_1, key_1)
print(result_1)  # Output: d

# Example usage: object-2
object_2 = {"x": {"y": {"z": "a"}}}
key_2 = "x/y/z"

result_2 = get_value_from_nested_object(object_2, key_2)
print(result_2)  # Output: a
```