NESTED DICTIONARIES

You can **nest dictionaries** as values of another dictionary

- The nested dictionary is referred to as an *inner* dictionary (the other is an *outer* dictionary)

```
item_history = {
    2019: {"skis": [249.99, 10, "in stock"], "snowboard": [219.99, 0, "sold out"]},
    2020: {"skis": [259.99, 10, "in stock"], "snowboard": [229.99, 0, "sold out"]},
    2021: {"skis": [269.99, 10, "in stock"], "snowboard": [239.99, 0, "sold out"]},
}

item_history
```

```
{2019: {'skis': [249.99, 10, 'in stock'],
  'snowboard': [219.99, 0, 'sold out']},
 2020: {'skis': [259.99, 10, 'in stock'],
  'snowboard': [229.99, 0, 'sold out']},
 2021: {'skis': [269.99, 10, 'in stock'],
  'snowboard': [239.99, 0, 'sold out']}}
```

The outer dictionary here has years as keys, and inner dictionaries as values

The inner dictionaries have items as keys, and lists with item attributes as values

```
item_history[2020]
```

```
{'skis': [259.99, 10, 'in stock'], 'snowboard': [229.99, 0, 'sold out']}
```

To access an inner dictionary, reference the outer dictionary key

```
item_history[2020]['skis']
```

```
[259.99, 10, 'in stock']
```

To access the values of an inner dictionary, reference the outer dictionary key, then the inner dictionary key of interest

ASSIGNMENT: NESTED DICTIONARIES

NEW MESSAGE

February 10, 2022

From: **Jerry Slush** *(IT Manager)*

Subject: **European Dictionary Updates**

Hi again!

We decided to restructure the dictionary you created earlier into a nested dictionary with each attribute represented by a key for fast lookup. Can you:

1. Verify the price on item 10009
2. Update the sizes for item 10009 to European sizing (they are stored in a list)
3. Create a dictionary based on the new one that includes item name as keys and sizes as values

modified_european_dictionary.ipynb

Reply | Forward

Results Preview

```
175.99
```

```
euro_data[10009]
```

```
{'name': 'Ski Boots',
 'price': 175.99,
 'category': 'hardware',
 'sizes': [37, 38, 39.5, 40.5, 41.5, 43.5, 44.5, 46.5]}
```

```
product_sizes
```

```
{'Coffee': ['250mL'],
 'Beanie': ['Child', 'Adult'],
 'Gloves': ['Child', 'Adult'],
 'Sweatshirt': ['XS', 'S', 'M', 'L', 'XL', 'XXL'],
 'Helmet': ['Child', 'Adult'],
 'Snow Pants': ['XS', 'S', 'M', 'L', 'XL', 'XXL'],
 'Coat': ['S', 'M', 'L'],
 'Ski Poles': ['S', 'M', 'L'],
 'Ski Boots': [37, 38, 39.5, 40.5, 41.5, 43.5, 44.5, 46.5
```

ASSIGNMENT: NESTED DICTIONARIES

NEW MESSAGE

February 10, 2022

From: **Jerry Slush** *(IT Manager)*

Subject: **European Dictionary Updates**

Hi again!

We decided to restructure the dictionary you created earlier into a nested dictionary with each attribute represented by a key for fast lookup. Can you:

1. Verify the price on item 10009
2. Update the sizes for item 10009 to European sizing (they are stored in a list)
3. Create a dictionary based on the new one that includes item name as keys and sizes as values

modified_european_dictionary.ipynb

Reply | Forward

Solution Code

```
euro_data[10009]['price']
```

```
175.99
```

```
boot_sizes = [37, 38, 39.5, 40.5, 41.5, 43.5, 44.5, 46.5]

euro_data[10009]['sizes'] = boot_sizes
```

```
product_sizes = {}
for product_details in euro_data.values():
    product_sizes[product_details['name']] = product_details['sizes']

product_sizes
```

```
{'Coffee': ['250mL'],
 'Beanie': ['Child', 'Adult'],
 'Gloves': ['Child', 'Adult'],
 'Sweatshirt': ['XS', 'S', 'M', 'L', 'XL', 'XXL'],
 'Helmet': ['Child', 'Adult'],
 'Snow Pants': ['XS', 'S', 'M', 'L', 'XL', 'XXL'],
 'Coat': ['S', 'M', 'L'],
 'Ski Poles': ['S', 'M', 'L'],
 'Ski Boots': [37, 38, 39.5, 40.5, 41.5, 43.5, 44.5, 46.5]}
```