Hands on exercises on dictionary

*'''*

*Exercise 1: Create a Dictionary*

*1. Create a dictionary called `person` with the following key-value pairs:*

*- Name: "Alice"*

*- Age: 25*

*- City: "New York"*

*2. Print the dictionary.*

*'''*

person = {

"Name" : "Alice",

"Age" : 25,

"City" : "New York"

}

print(person)

''' Exercise 2: Access Dictionary Elements

C.

2. Access the value of the `"City"` key and print it.'''

print(person["City"])

'''Exercise 3: Add and Modify Elements

1. Add a new key-value pair to the `person` dictionary: `"email": "alice@example.com"`.

2. Change the value of the `"Age"` key to 26.

3. Print the modified dictionary.'''

person["email"] = "alice@example.com"

person["Age"] = 26

print(person)

''' Exercise 4: Remove Elements

1. Remove the `"City"` key from the `person` dictionary.

2. Print the dictionary after removing the key.'''

person.pop("City")

print(person)

'''Exercise 5: Check if a Key Exists

1. Check if the key `"email"` exists in the `person` dictionary. Print a message based on the result.

2. Check if the key `"phone"` exists in the dictionary. Print a message based on the result.'''

if "email" in person:

print("The key 'email' exists in the dictionary.")

else:

print("The key 'email' does not exist in the dictionary.")

if "phone" in person:

print("The key 'phone' exists in the dictionary.")

else:

print("The key 'phone' does not exist in the dictionary.")

'''Exercise 6: 5: Check if a Key Exists

1. Check if the key `"email"` exists in the `person` dictionary. Print a message based on the result.

2. Check if the key `"phone"` exists in the dictionary. Print a message based on the result.'''

if "email" in person:

print("The key 'email' exists in the dictionary.")

else:

print("The key 'email' does not exist in the dictionary.")

if "phone" in person:

print("The key 'phone' exists in the dictionary.")

else:

print("The key 'phone' does not exist in the dictionary.")

''' Exercise 7: Nested Dictionary

1. Create a dictionary called `employees` where the keys are employee IDs (`101`, `102`, `103`) and the values are dictionaries containing employee details (like name and job title). Example structure:

```python

employees = {

101: {"name": "Bob", "job": "Engineer"},

102: {"name": "Sue", "job": "Designer"},

103: {"name": "Tom", "job": "Manager"}

}

```

2. Print the details of employee with ID `102`.

3. Add a new employee with ID `104`, name `"Linda"`, and job `"HR"`.

4. Print the updated dictionary.'''

employees = {

101: {"name": "Bob", "job": "Engineer"},

102: {"name": "Sue", "job": "Designer"},

103: {"name": "Tom", "job": "Manager"}

}

print(employees[102])

employees[104] = {"name": "Linda", "job": "HR"}

print(employees)

''' Exercise 8: Dictionary Comprehension

1. Create a dictionary comprehension that generates a dictionary where the keys are numbers from 1 to 5 and the values are the squares of the keys.

2. Print the generated dictionary.'''

squares = {num: num\*\*2 for num in range(1, 6)}

print(squares)

'''Exercise 9: Merge Two Dictionaries

1. Create two dictionaries:

```python

dict1 = {"a": 1, "b": 2}

dict2 = {"c": 3, "d": 4}

```

2. Merge `dict2` into `dict1` and print the result.'''

dict1 = {"a": 1, "b": 2}

dict2 = {"c": 3, "d": 4}

for key, value in dict2.items():

dict1[key] = value

print(dict1)

''' Exercise 10: Default Dictionary Values

1. Create a dictionary that maps letters to numbers: `{"a": 1, "b": 2, "c": 3}`.

2. Use the `get()` method to retrieve the value of key `"b"`.

3. Use the `get()` method to try to retrieve the value of a non-existing key `"d"`, but provide a default value of `0` if the key is not found.'''

letters = {"a": 1, "b": 2, "c": 3}

print(letters["b"])

print(letters.get("d", 0))

''' Exercise 11: Dictionary from Two Lists

1. Given two lists:

```python

keys = ["name", "age", "city"]

values = ["Eve", 29, "San Francisco"]

```

2. Create a dictionary by pairing corresponding elements from the `keys` and `values` lists.

3. Print the resulting dictionary.'''

keys = ["name", "age", "city"]

values = ["Eve", 29, "San Francisco"]

result = {keys[i]: values[i] for i in range(len(keys))}

print(result)

'''Exercise 12: Count Occurrences of Words

1. Write a Python program that takes a sentence as input and returns a dictionary that counts the occurrences of each word in the sentence.

```python

sentence = "the quick brown fox jumps over the lazy dog the fox"

```

2. Print the dictionary showing word counts.'''

sentence = "the quick brown fox jumps over the lazy dog the fox"

word\_counts = {}

for word in sentence.split():

word\_counts[word] = word\_counts.get(word, 0) + 1

print(word\_counts)