List of dictionary methods:

- clear()
- copy()
- fromkeys(iterable, value=None)
- get(key, default=None)
- items()
- keys()
- pop(key, default=None)
- popitem()
- setdefault(key, default=None)
- update(other_dict)
- values()

Explanation with code

- clear(): This method removes all items from the dictionary.

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Clear the dictionary
my_dict.clear()

# Output: {}
print(my_dict)
```

- copy(): This method returns a shallow copy of the dictionary. The new dictionary will contain the same keys and values as the original dictionary.

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Make a copy of the dictionary
new_dict = my_dict.copy()
new_dict["apple"] = 3  # without changing the orignal dict

# Output: {'apple': 3, 'banana': 3, 'orange': 1}
print(new_dict)
```

- get(key, default=None): This method returns the value for a given key. If the key is not found in the dictionary, it returns the default value (which defaults to None).

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Get the value for the key 'apple'
# Output: 2
print(my_dict.get('apple'))

# Get the value for the key 'pear'
# Output: None
print(my_dict.get('pear'))

# Get the value for the key 'pear', with a default value of 0
# Output: 0
print(my_dict.get('pear', 0))
```

- items(): This method returns a list of key-value pairs in the dictionary as tuples.

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Get the key-value pairs as a list of tuples
# Output: [('apple', 2), ('banana', 3), ('orange', 1)]
print(my_dict.items())
```

- keys(): This method returns a list of all the keys in the dictionary.

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Get the keys as a list
# Output: ['apple', 'banana', 'orange']
print(my_dict.keys())
```

- pop(key, default=None): This method removes the key-value pair for a given key and returns the value. If the key is not found in the dictionary, it returns the default value (which defaults to None).

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Remove the key-value pair for the key 'banana' and return the value
# Output: 3
print(my_dict.pop('banana'))
```

```
# Remove the key-value pair for the key 'pear' and return the default value
# Output: 0
print(my_dict.pop('pear', 0))
```

- popitem(): This method removes and returns an arbitrary key-value pair from the dictionary. If the dictionary is empty, it raises a KeyError.

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Remove and return an arbitrary key-value pair from the dictionary
# Output: ('orange', 1)
print(my_dict.popitem())
```

- setdefault(key, default=None): This method returns the value for a given key. If the key is not found in the dictionary, it adds the key with the default value (which defaults to None) and returns the default value.

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Get the value for the key 'apple'
# Output: 2
print(my_dict.setdefault('apple'))

# Get the value for the key 'pear', which doesn't exist in the dictionary, c
# Output: 0
print(my_dict.setdefault('pear', 0))

# The dictionary now contains the key-value pair for 'pear'
# Output: {'apple': 2, 'banana': 3, 'orange': 1, 'pear': 0}
print(my_dict)
```

- update(other_dict): This method updates the dictionary with the key-value pairs from another dictionary. If a key exists in both dictionaries, the value in the original dictionary is replaced with the value from the other dictionary.

```
# Create two dictionaries
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}
```

```
other_dict = {'pear': 4, 'kiwi': 2}

# Update my_dict with the key-value pairs from other_dict
my_dict.update(other_dict)

# The updated dictionary contains all the key-value pairs
# Output: {'apple': 2, 'banana': 3, 'orange': 1, 'pear': 4, 'kiwi': 2}
print(my_dict)
```

- values(): This method returns a list of all the values in the dictionary.

```
# Create a dictionary
my_dict = {'apple': 2, 'banana': 3, 'orange': 1}

# Get the values as a list
# Output: [2, 3, 1]
print(my_dict.values())
```