texta-1

February 3, 2025

```python
[4]: # Python List Inbuilt Functions and Slicing

# Creating a sample list
my_list = [10, 20, 30, 40, 50, 60, 70, 80, 90]

# 1. append(): Add an item to the end of the list
my_list.append(100)
print("After append:", my_list)

# 2. extend(): Extend the list by appending elements from another iterable
my_list.extend([110, 120])
print("After extend:", my_list)

# 3. insert(): Insert an item at a specific position
my_list.insert(2, 25)  # Insert 25 at index 2
print("After insert:", my_list)

# 4. remove(): Remove the first occurrence of an item
my_list.remove(30)
print("After remove:", my_list)

# 5. pop(): Remove and return an item at a given position (default is the last␣
  ↪item)
popped_item = my_list.pop()
print("Popped item:", popped_item)
print("After pop:", my_list)

# 6. index(): Return the index of the first occurrence of an item
index_of_40 = my_list.index(40)
print("Index of 40:", index_of_40)

# 7. count(): Count the occurrences of an item
count_of_20 = my_list.count(20)
print("Count of 20:", count_of_20)

# 8. sort(): Sort the list in ascending order (can use reverse=True for␣
  ↪descending)
```

```python
my_list.sort()
print("After sort:", my_list)

# 9. reverse(): Reverse the elements of the list
my_list.reverse()
print("After reverse:", my_list)

# 10. copy(): Create a shallow copy of the list
list_copy = my_list.copy()
print("Copied list:", list_copy)

# 11. clear(): Remove all items from the list
my_list.clear()
print("After clear:", my_list)

# Slicing Examples
sliced_list = list_copy[2:6]  # Get elements from index 2 to 5
print("Sliced list (index 2 to 5):", sliced_list)

sliced_step = list_copy[::2]  # Get every second element
print("Sliced list (every second element):", sliced_step)

reversed_list = list_copy[::-1]  # Reverse the list using slicing
print("Reversed list (slicing):", reversed_list)

first_three = list_copy[:3]  # First three elements
print("First three elements:", first_three)

last_three = list_copy[-3:]  # Last three elements
print("Last three elements:", last_three)
```

```
After append: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
After extend: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120]
After insert: [10, 20, 25, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120]
After remove: [10, 20, 25, 40, 50, 60, 70, 80, 90, 100, 110, 120]
Popped item: 120
After pop: [10, 20, 25, 40, 50, 60, 70, 80, 90, 100, 110]
Index of 40: 3
Count of 20: 1
After sort: [10, 20, 25, 40, 50, 60, 70, 80, 90, 100, 110]
After reverse: [110, 100, 90, 80, 70, 60, 50, 40, 25, 20, 10]
Copied list: [110, 100, 90, 80, 70, 60, 50, 40, 25, 20, 10]
After clear: []
Sliced list (index 2 to 5): [90, 80, 70, 60]
Sliced list (every second element): [110, 90, 70, 50, 25, 10]
Reversed list (slicing): [10, 20, 25, 40, 50, 60, 70, 80, 90, 100, 110]
First three elements: [110, 100, 90]
```

```
      Last three elements: [25, 20, 10]
```

[5]:
```python
# Python Dictionary Inbuilt Functions and Examples

# Creating a sample dictionary
my_dict = {
    "name": "Ravi",
    "age": 34,
    "job": "Assistant Professor",
    "city": "Delhi"
}

# 1. keys(): Get all keys in the dictionary
keys = my_dict.keys()
print("Keys:", keys)

# 2. values(): Get all values in the dictionary
values = my_dict.values()
print("Values:", values)

# 3. items(): Get all key-value pairs as a list of tuples
items = my_dict.items()
print("Items:", items)

# 4. get(): Get the value for a specific key
age = my_dict.get("age")
print("Value for 'age':", age)

# 5. pop(): Remove an item by key and return its value
job = my_dict.pop("job")
print("Popped 'job':", job)
print("After pop:", my_dict)

# 6. popitem(): Remove and return the last inserted key-value pair
last_item = my_dict.popitem()
print("Popped last item:", last_item)
print("After popitem:", my_dict)

# 7. update(): Update the dictionary with key-value pairs from another
 ↪dictionary
my_dict.update({"country": "India", "city": "Mumbai"})
print("After update:", my_dict)

# 8. setdefault(): Get the value of a key, or insert it with a default value if
 ↪not present
default_value = my_dict.setdefault("hobby", "Reading")
print("Default value added for 'hobby':", default_value)
```

```python
print("After setdefault:", my_dict)

# 9. copy(): Create a shallow copy of the dictionary
dict_copy = my_dict.copy()
print("Copied Dictionary:", dict_copy)

# 10. clear(): Remove all items from the dictionary
my_dict.clear()
print("After clear:", my_dict)
```

```
Keys: dict_keys(['name', 'age', 'job', 'city'])
Values: dict_values(['Ravi', 34, 'Assistant Professor', 'Delhi'])
Items: dict_items([('name', 'Ravi'), ('age', 34), ('job', 'Assistant
Professor'), ('city', 'Delhi')])
Value for 'age': 34
Popped 'job': Assistant Professor
After pop: {'name': 'Ravi', 'age': 34, 'city': 'Delhi'}
Popped last item: ('city', 'Delhi')
After popitem: {'name': 'Ravi', 'age': 34}
After update: {'name': 'Ravi', 'age': 34, 'country': 'India', 'city': 'Mumbai'}
Default value added for 'hobby': Reading
After setdefault: {'name': 'Ravi', 'age': 34, 'country': 'India', 'city':
'Mumbai', 'hobby': 'Reading'}
Copied Dictionary: {'name': 'Ravi', 'age': 34, 'country': 'India', 'city':
'Mumbai', 'hobby': 'Reading'}
After clear: {}
```

[ ]: