ANSWERS

PYTHON - DATA STRUCTURE

TOPICS – Methods of Dictionary & Sets

Dictionary

A dictionary is used to map or associate things you want to store the keys you need to get them. A dictionary in Python is just like a dictionary in the real world. Python Dictionary are defined into two elements Keys and Values.

Keys will be a single element Values can be a list or list within a list, numbers, etc.

```
dictItem = {
            "cricket": "Virat Kohali",
            "actor": "Sidhart Malhotra",
            "actress": "Kajol",
            "singer": "Armaan Malik",
            "female_singer": "Ishita V."
            }
```

1) type():

```
# type()  # Returns class type

print(type(dictItem))  # <class 'dict'>
```

2) str():

```
# str()  # Return the string

print(str(dictItem))
```

3).len():

```
# len()  # Return count of key entities

print(len(dictItem))  # 5
```

4) get():

```
# get() Returns the value of the specified key

print(dictItem)
print(dictItem.get("singer"))  # Armaan Malik
```

5) del():

```
# del*

del dictItem   # delete the dictionary
print(dictItem)

del dictItem["actor"]    # Deleting a specific key
print(dictItem)
```

6) pop():

```
# pop()*  Removes the element with the specified key

delItem = dictItem.pop("actor")
print(delItem)                  # Sidhart Malhotra
```

7) popitem():

```
# popitem() #  Removes the last inserted key-value pair
dictItem.popitem()
print(dictItem)
```

8) clear():

```
# clear()*  Removes all the elements from the dictionary

dictItem.clear()
print(dictItem)      # {}
```

9) copy():

```
# copy()* Returns a copy of the dictionary

dictItem2 = dictItem.copy()
print(dictItem2)
```

10) keys():

```
# keys()  Returns a list containing the dictionary's keys

print(dictItem.keys())
# dict_keys(['cricket', 'actor', 'actress', 'singer', 'female_singer'])
```

11) values():

```
# values()    Returns a list of all the values in the

print(dictItem.values())
# dict_values(['Virat Kohali', 'Sidhart Malhotra', 'Kajol', 'Armaan Malik',
                'Ishita V.'])
```

12) item():

```
# items() Returns a list containing a tuple for each key value pair
print(dictItem.items())
```

```
# dict_items([('cricket', 'Virat Kohali'), ('actor', 'Sidhart Malhotra'),
      ('actress', 'Kajol'), ('singer', 'Armaan Malik'), ('female_singer',
           'Ishita V.')])
```

13) fromkeys():

```
# fromkeys()  Returns a dictionary with the specified keys and value

dictItem2 = dictItem.fromkeys()
print(dictItem2)
```

14) update():

```
# update()* Updates the dictionary with the specified key-value pairs

# Dictionary with three items
Dictionary1 = {'A': 'Apple', 'B': 'Banana'}
Dictionary2 = {'B': 'Bat'}

# Dictionary before Updation

print("Original Dictionary:")
print(Dictionary1)

# update the value of key 'B'

Dictionary1.update(Dictionary2)
print("Dictionary after updation:")
print(Dictionary1)
```

15) setdefault():

```
# setdefault()  # Returns the value of the specified key.

# #Get the value of the "color" item, if the "color" item does not exist,
# insert "color" with the value "white":
```

```
car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

print(car)
x = car.setdefault("model", "Bronco")

print(x)   # Mustang


car = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}

# If the key does not exist:
# insert the key, with the specified value

x = car.setdefault("color", "white")


print(x)    # white

# {'brand': 'Ford', 'model': 'Mustang', 'year': 1964, 'color': 'white'}
print(car)
```