{Python Dictionaries}

Python Collection Data Types

thislist = ["apple", "banana", "cherry"]

thistuple = ("apple", "banana", "cherry", "orange“)

{DICTIONARY} – collection which is unordered, changeable, and indexed.

What is a {Dictionary}

- Similar to a list, but uses **{key:value}** pairs.
- Items in dictionaries are **unordered** collection of items.
- Place items in **{curly braces}**

Empty {Dictionary}

items = { }

released = { }

personalInfo = { }

Dictionaries – {Key:Value} Pairs

EXAMPLE:

personalInfo = {“Name”:”Molly”, “Age”:18}

1st Item are **Keys**

2nd Item are **Values**

Dictionaries – {Key:Value} Pairs

EXAMPLE:

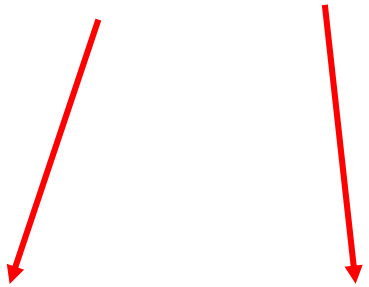
myCombo= {12345: “Luggage Combo”, 42: “The Answer”}

1st Item are **Keys**

2nd Item are **Values**

Dictionaries – {Key:Value} Pairs

EXAMPLE:



phonebook = {“Savannah”:”476-3321”, “Nate”:”351-7743”}

1st Item are **Keys**

2nd Item are **Values**

Create and Print a {Dictionary}

Save As: dictionary.py

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

print(thisDictionary)                    #What is the Output?
```

Create and Print a {Dictionary} – {Multi-line}

#CREATES the dictionary

```
thisDictionary = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
print(thisDictionary)
```

#What is the Output?

EXAMPLE: Dictionaries – {Multi-line}

```
released = {
  “iPhone": “2007",
  “iPhone 3G": “2008",
  “iPhone 3GS": “2009”,
  “iPhone 4": “2010",
  “iPhone 4S": “2011",
  “iPhone 5": “2012”,
}
print(released)
```

EXAMPLE: Dictionaries – {Multi-line}

```
MLB_team = {
  “Colorado": “Rockies",
  “Boston": “Red Sox",
  “Minnesota": “Twins”,
  “Milwaukee": “Brewers",
  “Seattle": “Mariners",
}
print(MLB_team)
```

Accessing Elements – use keys enclosed in [brackets]

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

aVar = **thisDictionary**["model"] **#What is OUTPUT?**

print(aVar)

Accessing Elements – use keys enclosed in [brackets]

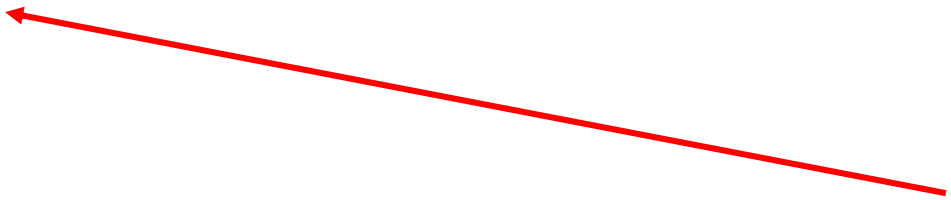
```
thisDictionary = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
```

```
print(thisDictionary[“model”])
```

*same as previous slide, except accessing and printing is in the same print line

CHANGING Elements – use = assignment operator

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



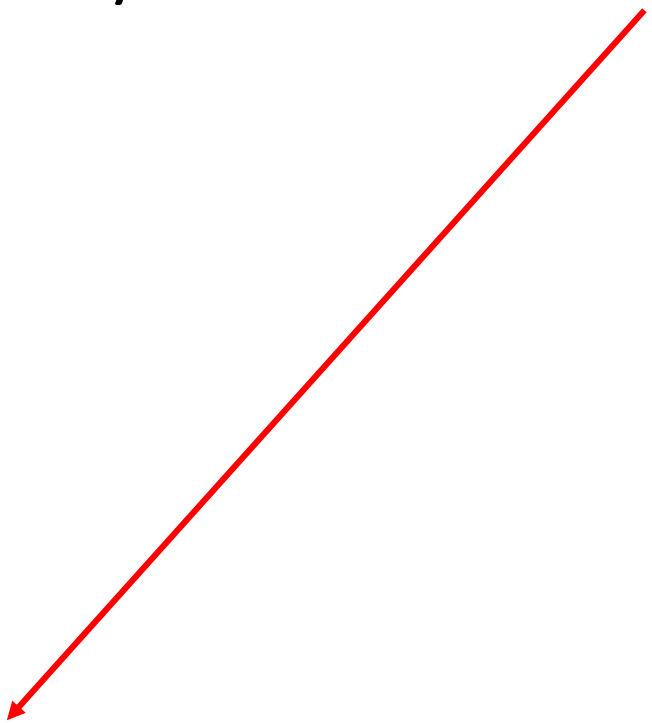
thisDictionary["year"] = 2019

print(**thisDictionary**)

#What is OUTPUT?

ADDING Element to Dictionary = use new key:value

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



thisDictionary[“color”] = “red”

print(**thisDictionary**)

#What is OUTPUT?

REMOVE Element in Dictionary – use pop( ) method

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

thisDictionary.pop(“model”)

```
print(thisDictionary)
```

****use pop( ) method to remove an item with a specified key**

#What is OUTPUT?

DELETE Element in Dictionary – use del keyword

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

**use del method to remove an item with a specified key

```
del thisDictionary["model"]
 print(thisDictionary)
```

#What is OUTPUT?

DELETE entire Dictionary

```
thisDictionary = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
del thisDictionary
 print(thisDictionary)
```

#What is OUTPUT?

Check If Item Exists in Dictionary – in keyword

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

```
if "model" in thisDictionary:
    print("Yes, 'model' is one of the keys in the thisDictionary dictionary")
```

Check If Item Exists in Dictionary – in keyword

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

****OUTPUT – will be TRUE**

```
print("brand" in thisDictionary)
```

Use a for Loop Through a Dictionary – use key

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

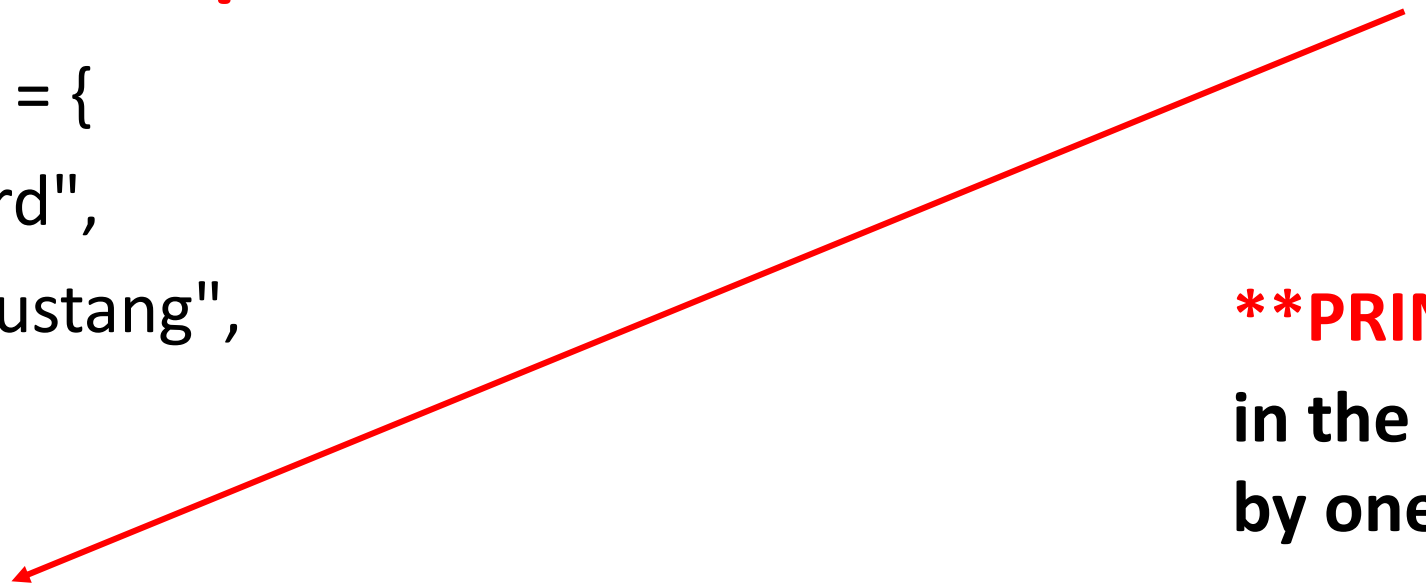
```
for anyVar in thisDictionary:
  print(anyVar)
```

**PRINTS all key names in the Dictionary one by one

#What is output?

Use a for Loop Through a Dictionary – use value

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



```
for anyVar in thisDictionary:
  print(thisDictionary[anyVar])
```

**PRINTS all values in the Dictionary one by one

#What is output?

Use a for Loop Through a Dictionary – key/value

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

Prints all key/values in the Dictionary one by one

#What is output?

```
for anyVar in thisDictionary:
  print(anyVar, thisDictionary[anyVar])
```

Dictionary Methods

Dictionary Methods

Python has a set of built-in methods that you can use on dictionaries.

Method	Description
clear()	Removes all the elements from the dictionary
copy()	Returns a copy of the dictionary
fromkeys()	Returns a dictionary with the specified keys and values
get()	Returns the value of the specified key
items()	Returns a list containing the a tuple for each key value pair
keys()	Returns a list containing the dictionary's keys
pop()	Removes the element with the specified key
popitem()	Removes the last inserted key-value pair
setdefault()	Returns the value of the specified key. If the key does not exist: insert the key, with the specified value
update()	Updates the dictionary with the specified key-value pairs
values()	Returns a list of all the values in the dictionary

Built-in Functions with Dictionary

all( )	****Search the Internet for more examples on how to use these functions.**
any( )	
len( )	
sorted( )	