Min-Max with Key function

functions | communitycreator | min() | max()

Two forms of min() and max() functions

Syntax

- non-iterables

```
min(arg1, arg2, *args, key)


max(arg1, arg2, *args, key)
```

- iterables

```
min(iterable, *iterables, key, default)


max(iterable, *iterables, key, default)
```

Example: non-iterables

```
# Example on non iterables

min_result = min(4, -5, 23, 5, 10, -10)
print("The minimum number is:", min_result)

max_result = max(4, -5, 23, 5, 10, -10)
print("The maximum number is:", max_result)
```

Examples: iterables

1. List:

```
# Example on a list

number = [3, 2, 8, 5, 10, 6]

smallest_number = min(number)
print("The smallest number is:", smallest_number)

largest_number = max(number)
print("The largest number is:", largest_number)
```

2. Using built-in function `len` as key:

```
# Example using key

languages = ["Python", "C Programming", "Java", "JavaScript"]

smallest_string = min(languages, key = len);
print("The smallest string is:", smallest_string)

largest_string = max(languages, key = len);
print("The largest string is:", largest_string)
```

3. Using user-defined function as key on list:

```
def findMax(num):
    rem = 0
    while(num):
        rem = num%10
        return rem

print('Number with max remainder is:', max(11,48,33,17,19, key=findMax))


num = [11,48,33,17]
print('Number with min remainder is:', min(num, key=findMax))
```

4. Using user-defined function as key on tuple:

```
# Find out who is the youngest and eldest student
def myFunc(e):
    return e[1] # return age

L = [('Sam', 35),('Tom', 25),('Bob', 30)]

min_x = min(L, key=myFunc)
print("Youngest student is: ", min_x)

max_x = max(L, key=myFunc)
print("eldest student is:",max_x)


```

5. Using user-defined function as key on dictionary:

```
# Example of min() on Dictionaries

square = {2: 4, 3: 9, -1: 1, -2: 4}

# the smallest key
key1 = min(square)
print("The smallest key:", key1)

# the key whose value is the smallest
key2 = min(square, key = lambda k: square[k])

print("The key with the smallest value:", key2)

# getting the smallest value
print("The smallest value:", square[key2])
```

```
# Example of max() on Dictionaries

square = {2: 4, -3: 9, -1: 1, -2: 4}

# the largest key
key1 = max(square)
print("The largest key:", key1)

# the key whose value is the largest
key2 = max(square, key = lambda k: square[k])

print("The key with the largest value:", key2)

# getting the largest value
print("The largest value:", square[key2])
```

Contributor: RAJAMANI