

17.3.2025 10:49:24

map_test.py

Page 1/2

```

1
2 # HSLU / ICS/AI ML : Modul ADS : Algorithmen & Datenstrukturen
3 # Path   : uebung05/al/aufgabe01
4 # Version: Mon Mar 17 10:49:24 CET 2025
5
6 from uebung05.al.aufgabe01.map_impl import MapImpl
7
8
9 if __name__ == '__main__':
10
11     the_map = MapImpl()
12
13     print("map.size()      : " + str(the_map.size()))
14     print("map.is_empty() : " + str(the_map.is_empty()))
15
16     print("\nmap.put(1, \"one\") : " + str(the_map.put(1, "one")))
17     the_map.printMap()
18     print("map.size()      : " + str(the_map.size()))
19     print("map.is_empty() : " + str(the_map.is_empty()))
20
21     the_map.put(2, "two")
22     the_map.put(3, "three 1")
23     the_map.printMap("\n")
24
25     print("\nmap.put(3, \"three 2\") : " + str(the_map.put(3, "three 2")))
26     the_map.printMap()
27
28     print()
29     print("map.get(2) : " + str(the_map.get(2)))
30     print("map.get(4) : " + str(the_map.get(4)))
31
32     print("\nmap.remove(2) : " + str(the_map.remove(2)))
33     the_map.printMap()
34
35     print("\nmap.key_set() : " + str(the_map.key_set()))
36     print("map.values()   : " + str(the_map.values()))
37     print("map.entrySet() : " + str(", ".join(map(str, the_map.entrySet()))))
38
39
40

```

17.3.2025 10:49:24

map_test.py

Page 2/2

```

40
41 """ Session-Log (Note: The order of the entries is irrelevant):
42
43 map.size()      : 0
44 map.is_empty()  : True
45
46 map.put(1, "one") : None
47 Printing map (1 Entries):
48     1: one
49 map.size()      : 1
50 map.is_empty()  : False
51
52 Printing map (3 Entries):
53     1: one
54     2: two
55     3: three 1
56
57 map.put(3, "three 2") : three 1
58 Printing map (3 Entries):
59     1: one
60     2: two
61     3: three 2
62
63 map.get(2) : two
64 map.get(4) : None
65
66 map.remove(2) : two
67 Printing map (2 Entries):
68     1: one
69     3: three 2
70
71 map.key_set() : {1, 3}
72 map.values()  : ['one', 'three 2']
73 map.entrySet() : (1,one),(3,three 2)
74
75 """

```

17.3.2025 10:49:24

map_impl.py

Page 1/1

```

1
2 # HSLU / ICS/AIML : Modul ADS : Algorithmen & Datenstrukturen
3 # Path : uebung05/al/aufgabe01
4 # Version: Mon Mar 17 10:49:24 CET 2025
5
6 import enum
7 from uebung05.al.aufgabe01.entry import Entry
8
9
10 class MapImpl:
11
12     def __init__(self):
13         self._list = []
14
15     def size(self):
16         # TODO: Implement here...
17         return None
18
19     def is_empty(self):
20         # TODO: Implement here...
21         return None
22
23     def put(self, key, value):
24         # TODO: Implement here...
25         return None
26
27     def get(self, key):
28         # TODO: Implement here...
29         return None
30
31     def remove(self, key):
32         # TODO: Implement here...
33         return None
34
35     def values(self):
36         # TODO: Implement here...
37         return None
38
39     def key_set(self):
40         # TODO: Implement here...
41         return None
42
43     def entry_set(self):
44         # TODO: Implement here...
45         return None
46
47     def printMap(self, prefix = ""):
48         print(prefix + "Printing map (" + str(self.size()) + " Entries): ")
49         for e in self._list:
50             print(f"    {e.get_key():3d}: {e.get_value()}")
51

```

17.3.2025 10:49:24

entry.py

Page 1/1

```

1
2 # HSLU / ICS/AIML : Modul ADS : Algorithmen & Datenstrukturen
3 # Path : uebung05/al/aufgabe01
4 # Version: Mon Mar 17 10:49:24 CET 2025
5
6
7 class Entry:
8
9     def __init__(self, key, value):
10         self._key = key
11         self._value = value
12
13     def get_key(self):
14         return self._key
15
16     def get_value(self):
17         return self._value
18
19     def set_value(self, value):
20         old_value = self._value
21         self._value = value
22         return old_value
23
24     def __iter__(self):
25         yield self._key
26         yield self._value
27
28     def __eq__(self, other):
29         return (isinstance(other, type(self)) and tuple(self) == tuple(other))
30
31     def __hash__(self):
32         return hash(tuple(self))
33
34     def __str__(self):
35         return "(" + str(self._key) + "," + str(self._value) + ")"

```