

```

1 class HardDrive:
2     def __init__(self, hard_drive_id, capacity, computer_id):
3         self.hard_drive_id = hard_drive_id
4         self.capacity = capacity
5         self.computer_id = computer_id
6
7 class Computer:
8     def __init__(self, computer_id, model):
9         self.computer_id = computer_id
10        self.model = model
11
12 class ComputerHardDrives:
13     def __init__(self, hard_drive_id, computer_id):
14         self.hard_drive_id = hard_drive_id
15         self.computer_id = computer_id
16
17 def get_hard_drives_by_computer(computers, hard_drives):
18     result = {}
19     for computer in sorted(computers, key=lambda x: x.model):
20         drives_for_computer = [drive for drive in hard_drives if drive.computer_id == computer.computer_id]
21         result[computer.model] = drives_for_computer
22     return result
23
24 def get_total_capacity_by_computer(computers, hard_drives):
25     total_capacities = {}
26     for computer in computers:
27         total_capacity = sum([drive.capacity for drive in hard_drives if drive.computer_id == computer.computer_id])
28         total_capacities[computer.model] = total_capacity
29     return total_capacities
30
31 def get_computers_with_keyword(computers, hard_drives, keyword="computer"):
32     result = {}
33     for computer in computers:
34         if keyword.lower() in computer.model.lower():
35             drives_for_computer = [drive for drive in hard_drives if drive.computer_id == computer.computer_id]
36             result[computer.model] = drives_for_computer
37     return result
38

```

```
1 import unittest
2 from main import HardDrive, Computer, get_hard_drives_by_computer, get_total_capacity_by_computer, get_computers_with_keyword
3
4 class TestComputerHardDriveSystem(unittest.TestCase):
5     def setUp(self):
6         self.hard_drives = [
7             HardDrive(1, 500, 1),
8             HardDrive(2, 1000, 1),
9             HardDrive(3, 250, 2),
10            HardDrive(4, 750, 3),
11        ]
12        self.computers = [
13            Computer(1, "Lenovo Computer"),
14            Computer(2, "HP"),
15            Computer(3, "Dell Computer"),
16        ]
17
18    def test_get_hard_drives_by_computer(self):
19        result = get_hard_drives_by_computer(self.computers, self.hard_drives)
20        self.assertIn("Lenovo Computer", result)
21        self.assertEqual(len(result["Lenovo Computer"]), 2)
22
23    def test_get_total_capacity_by_computer(self):
24        result = get_total_capacity_by_computer(self.computers, self.hard_drives)
25        self.assertEqual(result["Lenovo Computer"], 1500)
26        self.assertEqual(result["HP"], 250)
27
28    def test_get_computers_with_keyword(self):
29        result = get_computers_with_keyword(self.computers, self.hard_drives, "computer")
30        self.assertIn("Lenovo Computer", result)
31        self.assertIn("Dell Computer", result)
32        self.assertNotIn("HP", result)
33
34 if __name__ == '__main__':
35     unittest.main()
```

```
PS E:\Courses\BAUMAN\sem3\ПикЯП\РК 2> & C:/Users/Zalman/AppData/Local/Programs/Python/Python312/python.exe "e:/Courses/BAUMAN/sem3/ПикЯП/РК 2/test.py"
```

```
...
```

```
-----
```

```
Ran 3 tests in 0.000s
```

```
OK
```

```
PS E:\Courses\BAUMAN\sem3\ПикЯП\РК 2>
```