machine_problems/mp5.py

```python
1   from student import Student
2
3
4   def getStudents():
5       #
6       # Opens the data file of student info... studentID, firstName, lastName, then
7       # testScores - the number of test scores is variable, from zero up... reads
8       # each line of data as a str, divides the line into the values... str, str, str,
9       # int, int, int... (a variable number of int values - could be none)...
10      # instantiates a new Student object, uses the Student object's instance methods
11      # to add those values to the object, adds the Student object to a dict of
    objects,
12      # and returns the dict of Student objects.
13      #
14      # There are no parameters.
15      #
16      # Returns a dict of objects from the Student class
17      #
18      students_dict = {}
19
20      with open("5010 - MP5 Data.txt") as source:
21          for line in source:
22              data = line.rstrip().split()
23
24              id = data[0]
25              student = Student(id)
26
27              firstName = data[1]
28              lastName = data[2]
29              student.setName(firstName, lastName)
30
31              testScores = data[3:]
32              for score in testScores:
33                  student.addTest(int(score))
34
35              students_dict[id] = student
36
37      return students_dict
38
39
40  def updateName(roster, stuID):
41      #
42      # Prompts the user to enter the first name and last name of a student. Then,
43      # uses the setName() instance method to change the name for the Student object
44      # in the dictionary roster with the key = stuID.
45      #
46      # roster A dictionary of objects from the Student class, using stuID as the
47      # key, and a Student object as the value.
48      # stuID A str that is the key for a Student object in the dictionary roster.
49      #
50      # There is no return value.
51      #
```

```python
52          student = roster[stuID]
53          student.setName(input("Enter first name: "), input("Enter last name: "))
54          print(f"\n{student}")
55
56
57  def updateTests(roster, stuID):
58          #
59          # Prompts the user to enter test scores for a student using addTest() to add
60          # each one to the Student object in the dictionary roster with the key = stuID.
61          #
62          # roster A dictionary of objects from the Student class, using stuID as the
63          # key, and a Student object as the value.
64          # stuID A str that is the key for a Student object in the dictionary roster.
65          #
66          # There is no return value.
67          #
68          user_prompt = "Enter a Test Score (<enter> to stop): "
69          testScore = input(user_prompt)
70
71          student = roster[stuID]
72          while testScore:
73              student.addTest(int(testScore))
74              testScore = input(user_prompt)
75
76
77  if __name__ == "__main__":
78      roster = getStudents()
79
80      for student in roster.values():
81          print(student)
82
83      print("You may update any student info, or add a student.\n")
84      idPrompt = "Enter Student ID (<enter to stop>): "
85      userInput = input(idPrompt)
86
87      while userInput:
88          if userInput not in roster:
89              print("This student does not exist. You may enter the info now.")
90
91              student = Student(userInput)
92              student.setName(input("First Name: "), input("Last Name: "))
93              roster[student.getID()] = student
94              updateTests(roster, student.getID())
95
96              print("\nNew Student Added:")
97              print(student)
98
99              userInput = input(f"{idPrompt}")
100         else:
101             student = roster[userInput]
102             print(student)
103
104             print("(1) Change the Name")
105             print("(2) Add a Test\n")
```

```python
106
107                userChoice = input("What would you like to do? (<enter> to stop): ")
108                if userChoice == "1":
109                    updateName(roster, student.getID())
110                elif userChoice == "2":
111                    updateTests(roster, student.getID())
112                    print(f"\n{student}")
113                else:
114                    print()
115
116            userInput = input(f"{idPrompt}")
117
```