```
#
# Aravind Kumar Kaspe
# Banner ID : 001291145
#
# Description : We have defined a class named Student with the details like
#               Instance Varaiables, Constructor, Public Instance Methods
#               and Private Instance Methods. The Student class encapsulates
#               student information and test scores. Each student object has
#               a unique identifier, consisting of 2 initials and 3 random
#               digits. The class tracks the student's first and last name,
#               test scores, and calculates the average score. When
#               initialized, the test scores list is empty, and the
#               average is set to 0.0. The class provides methods to
#               retrieve student ID, name, test scores, and average. It also
#               allows for adding new test scores and updating the average
#               score accurately.
#
#
# Student Class Definition
#
# 4 Instance Variables:
# _stuID str - 5 characters, 2 alpha (initials), 3 numeric (random)
# _name list - 2 str elements, first name, last name
# _testScores list - int values (could be empty, no tests taken)
# _avg float - the average of _testScores (0.0 if no tests taken)
#
# Constructor:
# __init__(self, stuID) Initialize _stuId to stuID, _name to a list with
# 2 empty str elements, _testScores to the empty list,
# _avg to 0.0
#
# 6 Public Instance Methods:
# getID(self) return _stuID
# getName(self) return _name
# getTestScores(self) return _testScores
# getAvg(self) return _avg
# setName(self, firstName, lastName) Change the 2 elements in _name to
# firstName and lastName.
# addTest(self, testScore) Append 1 testScore onto _testScores,
# then call _calcAvg() to set _avg to
# the updated value.
#
# 1 Private Instance Method:
# _calcAvg(self) called by addTest() to keep _avg accurate every time
# a new test score is added, returns a float value that
# is the average of the test scores, 0.0 if no test scores
#
class Student:
    def __init__(self, stuID):
        self.__stuID = stuID
```

```
    self.__name = ["",""]
    self.__testScores = []
    self.__avg = 0.0

def getID(self):
    return self.__stuID

def getName(self):
    return self.__name

def getTestScores(self):
    return self.__testScores

def getAvg(self):
    return self.__avg

def setName(self, firstName, lastName):
    self.__name[0] = firstName
    self.__name[1] = lastName

def addTest(self, testScore):
    self.__testScores.append(testScore)
    self.__avg = self.__calcAvg()

def __calcAvg(self):
    if len(self.__testScores):
        return sum(self.__testScores)/len(self.__testScores)
    return 0.0
```