Programming I (Python)
Assignment 6

Instruction

Please compress the `a6` directory (and not the `code` directory). First, come to the parent directory of `a6`. Then, use the following command:

```
tar cvzf a6.tar.gz a6
```

Please note that you can use the above command verbatim (using copy-paste).

Theory

1. Discuss the following: Classes and objects Methods and attributes

   1. Constructor
   2. self keyword
   3. Static attributes. Need. Advantages.
   4. Compare static attributes with static variables in C

   (**File:** `theory.doc`)

2. Define a class `MyMath` with the functions `add`, `subtract`, `multiply` and `divide` with their usual meanings.

   For example:

   ```python
   mymath = MyMath()
   print mymath.add(1, 2)
   print mymath.subtract(1, 2)
   print mymath.multiply(1, 2)
   print mymath.divide(4, 2)
   ```

   will give us:

   ```
   3
   -1
   2
   2
   ```

(**File:** `mymath.py`)

3. Write a class `Student` with the following features:

   (a) Attributes: `name` and `rollNumber`

   (b) A constructor to initialise a `Student` object.

   (**File:** `student1.py`)

4. Write a class `Institute` with the following features:

   (a) An attribute `students` that is the list of `Student` objects (Q.3).

   (b) A constructor that initialises the newly constructed `Institute` object with appropriate initial value of `students` attribute. If nothing is passed, `students` is initialised to an empty list.

   (c) A method `isStudentOf` that, given a `Student` object `s`, returns `True` if `s` is a student of the given institute, else returns `False`.

   (d) A method `isAddable` that, given a `Student` object `s`, returns `True` if `s` can be added as student of the given institute. A student can be added to an institute if he/she has not already been added, and has a roll number distinct from all existing roll numbers of the students of that institute.

   (e) A method `addStudent` that, given a `Student` object `s`, adds `s` and returns `True` if `s` can be added as student of the given institute. Otherwise, it returns `False`.

   (**File:** `institute.py`)

5. Write two classes `Circle` and `Rectangle` with the following:

   1. `radius` is the attribute of `Circle`.

   2. `length` and `breadth` are attributes of `Rectangle`.

   3. their own `area()` and `circumference()` methods.

   4. `Circle` should use a static attribute `PI` for computing its area and circumference. `PI = 3.1415`.

   (**File:** `shapes.py`)

6. Write a `Student` class with the following:

   1. An attribute `name` of type `String`

   2. An attribute `rollNumber` of type `String`, of the format 'imt2020xxx'.

   3. A constructor which takes only one parameter of `String` type used to initialise the `name` attribute. The `rollNumber` attribute is given a value which is distinct from the `rollNumber` of all other `Student`s created so far. *(Hint: static attribute)*

   (**File:** `student2.py`)

   For example:

```
s1 = Student("Amar")
s2 = Student("Akbar")
s3 = Student("Anthony")
```

```
print s1.rollNumber
print s2.rollNumber
print s3.rollNumber
```

will give us:

```
1
2
3
```