Model 1   Common Methods

Classes are often used to represent abstract data types, such as `Color` or `Point`:

Color
-red: int -green: int -blue: int
+Color() +Color(red:int,green:int,blue:int) +add(other:Color): Color +darken(): Color +equals(obj:Object): boolean +lighten(): Color +subtract(other:Color): Color +toString(): String

Point
-x: int -y: int
+Point() +Point(x:int,y:int) +Point(other:Point) +equals(obj:Object): boolean +getX(): int +getY(): int +setX(x:int) +setY(y:int) +toString(): String

As shown in the UML diagrams, classes generally include the following kinds of methods (in addition to others):

- **constructor** methods that initialize new objects
- **accessor** methods (getters) that return attributes
- **mutator** methods (setters) that modify attributes

Questions (15 min)   Start time:

1. Identify the constructors for the `Color` class. What is the difference between them?

2. What kind of constructor does the `Point` class have that the `Color` class does not?

3. Identify an accessor method in the `Point` class.

   a) What is the name of the method?

   b) Which instance variable does it get?

   c) What arguments does the method take?

   d) What does the method return?

4. Identify a mutator method in the `Point` class.

   a) What is the name of the method?

   b) Which instance variable does it set?

   c) What arguments does the method take?

   d) What does the method return?

5. How would you define accessor methods for each attribute of the `Color` class? Write your answer using UML syntax.

6. How would you define mutator methods for each attribute of the `Color` class? Write your answer using UML syntax.

7. The `Color` class does not provide any accessors or mutators. Instead, it provides methods that return new `Color` objects. Why do you think the class was designed this way?