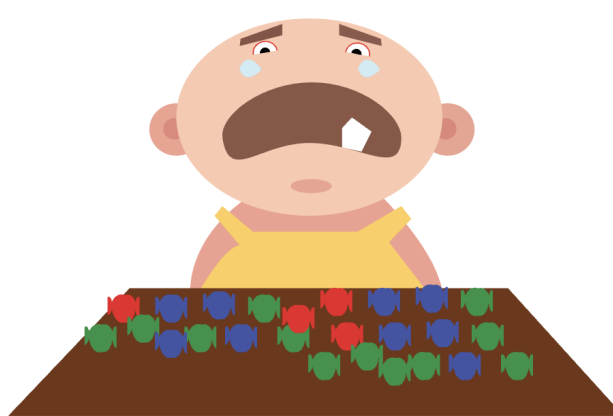


Containment



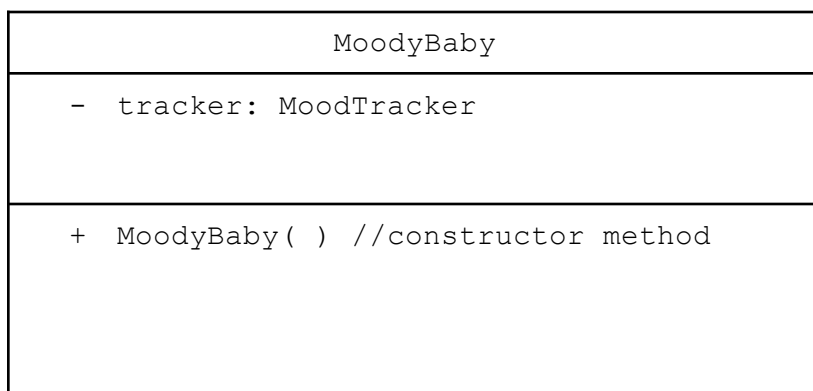
A **MoodyBaby** has an internal **MoodTracker**.

As the baby eats **Candy**, its mood level changes.

After eating, the MoodyBaby laughs or cries, depending on its mood.

Task 1

Open the `MoodyBaby` class and complete the UML diagram:



Task 2

Create a new project in your IDE. Using the code provided, create the `MoodyBaby` class and the `Candy` class.

Task 3

Use the following UML diagram to design the `MoodTracker` class:

MoodTracker
- moodLevel : int
+ MoodTracker() //initialise moodLevel to 0 + getMoodLevel() : int + increaseMood() : void //increase moodLevel by 1 + decreaseMood() : void //decrease moodLevel by 1

Task 4

Now we have designed all of the classes required for the project. Let's test our design with the following code:

```
public class Main {  
    public static void main(String[] args) {  
  
        //instantiate a MoodyBaby  
        MoodyBaby baby = new MoodyBaby();  
  
        //feed the baby 10 candies  
        for(int i = 0; i<10; i++){  
            baby.eatCandy(new Candy());  
        }  
  
        //check the baby's mood!  
        baby.speak();  
    }  
}
```

Task 5

This computer program takes advantage of Object Oriented Programming to manage data during the program's execution.

Answer the following questions. You may have to do some research.

- a. The relationship between the `MoodyBaby` class and the `MoodTracker` class is either:

- *Containment*
- *Aggregation*

Which one is it? Explain your answer.

- b. Can a `MoodTracker` object exist without a `MoodyBaby` object, in the context of this program?

YES	NO
-----	----

- c. The `MoodyBaby` **eats** `Candy` objects.

Does the baby store these objects anywhere eg in an `ArrayList` property or a static array?

Which of the following sentences is true about the relationship between the `MoodyBaby` class and the `Candy` class?

- A `MoodyBaby` **USES** `Candy`
- A `MoodyBaby` **HAS** `Candy`

d. In the main program, after the baby speaks let's do this:

```
baby = null;
```

What will be the consequence on the internal MoodTracker object, tracker, once this code is executed?

e. Reconfirm your chosen relationship between the MoodyBaby class and the MoodTracker class in the context of this program:

- *Containment*
- *Aggregation*

Make any further notes or questions that you have on the themes covered in this unit: