

CS161: Introduction to Computer Science

Lab Assignment 6

This week you will practice writing your own Java classes. You will write 2 Java classes that represent a primitive stomach and brain. For your homework, you will use these classes to implement a simple organism.

Recall that we are writing classes that *do not have a `main()` method*. The classes we are writing are not meant to be “executed” like the classes we have written in the past. Instead, these classes encapsulate some particular functionality and, like `String`, `Scanner`, and `Random`, are meant to be used in other classes.

The Stomach Class



Let's begin by writing a Java class to simulate a primitive stomach. At a minimum, a primitive stomach should be able to ingest (i.e. take in) some amount of food and digest some amount of food.

Create a new Java class named `Stomach`. Add instance variables to represent the current amount of food in the stomach and the total amount of food that has been digested (over the entire “lifespan” of the stomach).

Next add a constructor to initialize your instance variables. Finally, add the following methods to your class:

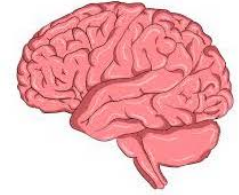
- `public void ingest(int amount)` – This method causes the stomach to ingest the specified amount of food
- `public void digest()` – This method causes the stomach to digest a random amount of the food currently in the stomach. Specifically, this should decrease the amount of food by some random amount and increase the amount digested by the same amount.
- `public int getAmountFood()` – This method simply returns the current amount of food in the stomach
- `public int getAmountDigested()` – This method simply returns the total amount of food that has been digested
- `public String toString()` – This method returns a `String` that summarizes the *state* of the stomach. A `toString()` method *does not print anything!*

Be sure to write Javadoc comments for each of these methods and to include the appropriate `@param` and `@return` tags where necessary.

Once you are finished, open the `Controller` class and add code to the `testStomach()` method to create a new `Stomach` object and to call various methods on it. You can call this method from `main()` in order to test if your stomach is working correctly.

The Brain Class

Now let's think about a primitive brain. At a minimum, a primitive brain should be able to hold a single thought, hold a single memory, and keep track of whether the organism (that the brain is a part of) is asleep or awake.



Create a new Java class named `Brain`. Add instance variables to represent a single thought, a single memory, and a variable (of what type?) to keep track of whether the brain is awake or asleep.

Next add a constructor to initialize your instance variables. Finally, add the following methods to your class:

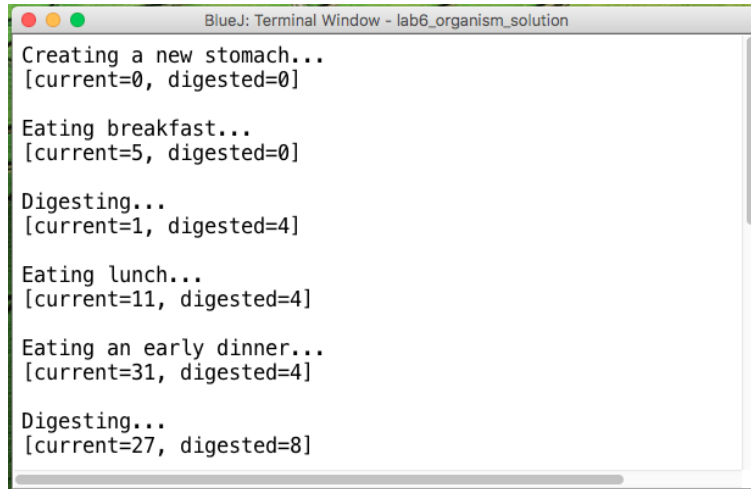
- `public String getThought()` – This method returns the current thought
- `public void setThought(String newThought)` – This method updates the current thought
- `public void remember()` – This method copies the current thought into the memory (thereby “remembering” it)
- `public void recall()` – This method copies the memory into the current thought (thereby “recalling” it)
- `public void wakeUp()` – This method wakes the brain up
- `public void goToSleep()` – This method puts the brain to sleep
- `public boolean isAwake()` – This method returns true if the brain is awake and false otherwise
- `public String toString()` – This method returns a `String` that summarizes the *state* of the brain. Again, a `toString()` method does not print anything.

Again, be sure to write Javadoc comments for each of these methods and to include the appropriate `@param` and `@return` tags where necessary.

Once you are finished, open the `Controller` class and add code to the `testBrain()` method to create a new `Brain` object and to call various methods on it. You can call this method from `main()` in order to test if your brain is working correctly.

An Example of My Stomach and Brain

Here is an example of the code I wrote in my `Controller` class to test my stomach and brain classes. I am making heavy use of my `toString()` methods to inspect the state of my stomach and brain after I call various methods on them. Your `toString()` methods can be formatted however you want. I prefer a comma-delimited list but you do not have to follow this convention.



```
BlueJ: Terminal Window - lab6_organism_solution
Creating a new stomach...
[current=0, digested=0]

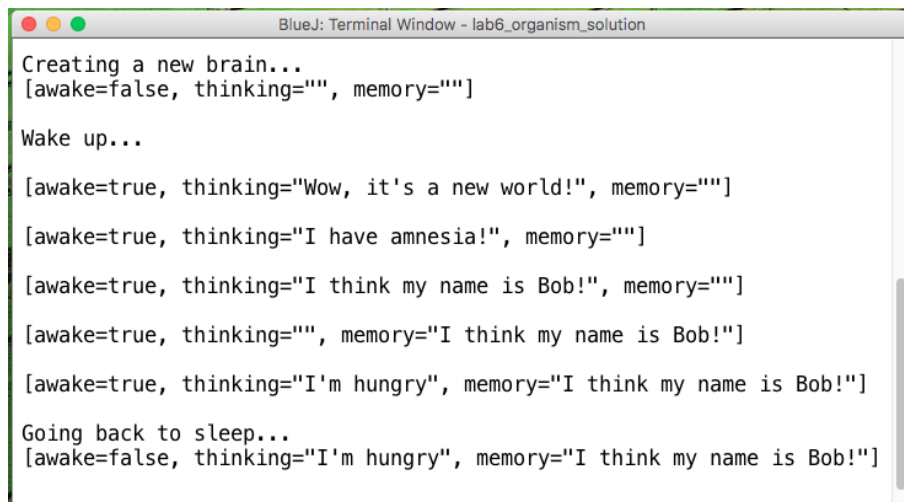
Eating breakfast...
[current=5, digested=0]

Digesting...
[current=1, digested=4]

Eating lunch...
[current=11, digested=4]

Eating an early dinner...
[current=31, digested=4]

Digesting...
[current=27, digested=8]
```



```
BlueJ: Terminal Window - lab6_organism_solution
Creating a new brain...
[awake=false, thinking="", memory=""]

Wake up...

[awake=true, thinking="Wow, it's a new world!", memory=""]

[awake=true, thinking="I have amnesia!", memory=""]

[awake=true, thinking="I think my name is Bob!", memory=""]

[awake=true, thinking="", memory="I think my name is Bob!"]

[awake=true, thinking="I'm hungry", memory="I think my name is Bob!"]

Going back to sleep...
[awake=false, thinking="I'm hungry", memory="I think my name is Bob!"]
```

Submitting your lab assignment

Submit whatever you finished in lab via Moodle. Note that if you did not finish lab you will need to do so since your `Brain` and `Stomach` classes are necessary for this week's homework assignment.