

National University of Singapore
School of Computing
CS1101S: Programming Methodology (JavaScript)
Semester I, 2012/2013

Recitation 8
Object-Oriented Programming

Problems:

1. Write a `Food` class

- Input state is the name, nutrition value, and `good_until` time.
- Additional state is the age of the food, initially 0.
- Methods are:
 - `getName` - returns the name of the food
 - `getAge` - returns the age of the food
 - `sitThere` - takes an amount of time, and increases the age of the food by the amount.
 - `eat` - return the nutrition if the food is still good; 0 otherwise.

2. Write an `AgedFood` class

- Input state is the same as the `Food` class, with an additional parameter, which is the `good_after` time.
- Should inherit from the `Food` class.
- Methods are:
 - `sniff` - returns `true` if it has aged enough to be good.
 - `eat` - returns 0 if the food is not good yet; otherwise behaves like normal food.

3. Write a `VendingMachine` class

- Input state is the same as the `Food` class.
- Additional state is age of the `VendingMachine`, initially 0.
- Methods are:
 - `sit_there` - takes an amount of time, and increases the age of the `VendingMachine` by *half* that amount (it's refridgerated!).
 - `sell_food` - returns a new `Food` instance with the appropriate name, nutrition, `good_until` and age.

4. **Homework:** How would you implement the `VendingMachine` so that it can sell both `Food` and `AgedFood` (and possibly other things too?).