

Project 1

OOP Project 1 – Create a virtual gumball machine

by

William Wright

CS-255\_L01\_2020

CS-225: Intermediate Java Programming

Prepared for

Professor Daniel Packer

**Introduction**

Create a virtual gum ball machine. You will create a **GumBallMachine** class, and use the following main function to interact with your class. You should get the expected output.

public static void main(String[] args) {  
  
 GumBallMachine machine = new GumBallMachine(2);  
 machine.inventory(); // prints "2"  
 machine.insertCoin(); // prints a random flavor  
 machine.insertCoin(); // prints a random flavor  
 machine.insertCoin(); // prints "SOLD OUT"  
 machine.inventory(); // prints "0"  
}

our gum ball machine will have at least 3 different flavors (e.g. cherry, lemon, etc). You will dispense a random flavor each time the user calls **insertCoin()**. If the machine is out of gum balls, you will print SOLD OUT. Each time the user calls ***inventory()***you will print the number of gum balls left in the machine. Use an array to hold the flavors. NOTE: you do not need to figure out what flavor every gum ball will be from the beginning, you can give a random flavor each time.

Things you will need to review:

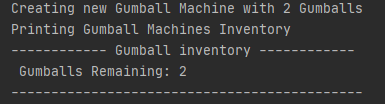
* classes & objects
* field and method modifiers (encapsulate! make everything private!)
* random integers
* arrays

**Extra Credit:**Copy your class to another class called **GumBallMachineEC.**Add an extra array to your gum ball machine, and use it to pre-calculate the flavor of every gum ball ahead of time. Then add a method called **shakeMachine()** which randomly shuffles the positions of the gum balls. When the user calls **insertCoin()**then the first available gum ball will be dispensed. Do not re-use dispensed gum balls! Modify **inventory()**so that it prints the contents of the gum ball machine. Shake it and print the before and after results. Sometimes they will be in the same order, and sometimes in a new order.

**Screen Shots of the expected output**

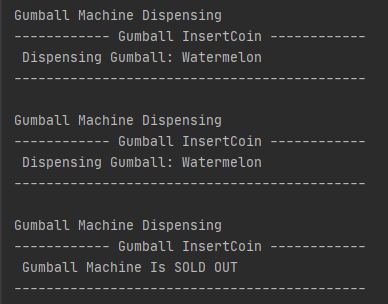
**Calling inventory() After Creating a new gumball machine**

//Create new Gumball Machine with 2 Gumballs  
GumBallMachine machine = new GumBallMachine(2);  
//print Gumball Machines Inventory  
machine.inventory();



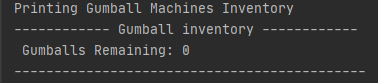
**Calling insertCoin() 3 Times**

//Call Gumball Machine insertCoin()  
machine.insertCoin();  
//Call Gumball Machine insertCoin()  
machine.insertCoin();  
//Call Gumball Machine insertCoin()  
machine.insertCoin();



**Calling inventory()**

//print Gumball Machines Inventory  
machine.inventory();



**Screenshots of Extra Credit:**

//Start extra credit  
GumBallMachineEC machine2 = new GumBallMachineEC(10);  
//print Gumball Machines Inventory  
machine2.inventory();  
//Call Gumball Machine insertCoin()  
machine2.insertCoin();  
//Call Shake Machine  
machine2.shakeMachine();  
  
////Call Gumball Machine insertCoin() 11 Times  
for(int i=0;i<11;i++){  
 machine2.insertCoin();  
}  
//print Gumball Machines Inventory  
machine2.inventory();

