vendor constructor:

This used to create the vendors object.

registerVendors():

this is used to add a new object within the defined vendor object

removeVendors():

this is used to delete an object within the defined vendor object memory

visitVendors():

this will let the user increment the interviewnum if it’s less than 15 for the vendor object or report an error if the vendor is unavailable.

displayVendors():

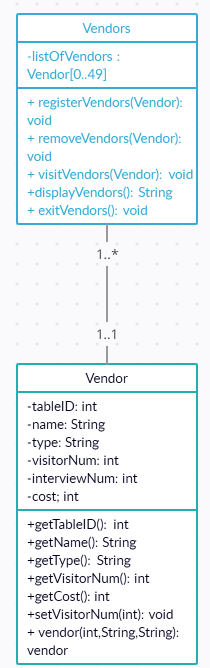
this is just like printing an entire array, it’s just outputs all the vendors stored within the object.

Vendor getters & setters:

They get and set values for the vendor object

Vendor constructor:

Streamlined object creation



For the vendors object, they only have the listOfVendors variable, which is just an array to hold all the vendors. As for the commands, the registerVendors takes a vendor and adds it to the listOfVendors while returning nothing. removeVendors does the same thing, but instead of adding it removes. visitVendor checks if the vendor object exists and if it’s visitorNum is less than 15 so it can increment it. The functions are all internal so it returns nothing. displayVendors() just returns a massive string of all the vendors within the listOfVendors. exitVendors() prints a large report and closes the program.

For the vendor object they have relatively self-explanatory variables. tableID is the tableID, name is the name, type is the type, visitorNum is the total number of visitors, interviewNum is the amount of in-person interviews, and lastly cost is the cost of the table. There are getters for all the variables, which by name, gets the values of the variables. There is one setter for the visitor since that’s the only variable that’s changing. Lastly the constructor is just there for simplicity as those are the necessary inputs for a vendor object.

Method: registerVendors

Purpose: adds a vendor to the listOfVendors

Input: Vendor

Return: none

Method: removeVendors

Purpose: removes a vendor from the listOfVendors

Input: Vendor

Return:none

Method: visitVendors

Purpose: checks if the vendor object exists and if it’s visitorNum is less than 15 so it can increment it

Input: Vendor

Return: none

Method: displayVendors

Purpose: displays all objects within listOfVendors

Input: none

Return: String

Method: exitVendors

Purpose: Prints out a final report and exits the program

Input: none

Return: none

Method: getTableID

Purpose: gets the tableID

Input: none

Return: tableID : int

Method: getName

Purpose: gets the name

Input: none

Return:name : String

Method: getType

Purpose: returns the vendor type (also I’m not importing anything other than scanner so this name should be fine)

Input: none

Return: type: String

Method: getVisitorNum

Purpose: returns the number of visitors

Input: none

Return: visitorNum:int

Method: getCost

Purpose: returns the cost

Input: none

Return: cost:int

Method: setVisitorNum

Purpose: sets the number of visitors

Input: int

Return: none

Method: vendor

Purpose: vendor constructor

Input: int, String, String

Return: Vendor