**Part 4: Draw the memory map when the program runs [1 point]**

Explain step by step what happened when the program runs and answer some questions.

- What is stored in the static heap, stack, dynamic heap?

**Answer:**

|  |  |  |
| --- | --- | --- |
| Stack | Heap | |
| Dynamic Heap | Static Heap |
| * Primitive variable used in method * Reference value | * Static variables, global variables * Static constants * Static object | * Object * Linked list, dynamic arrays, trees, strings,… * Large data/resources |

- What are objects in the program?

**Answer:**

***Objects*** *are instances of* ***classes*** *that represent data and behaviour.*

*Objects have characteristic of encapsulation, state, behaviour and identity*

- What is the state of obj1, obj2?

**Answer:**

Obj1: is created by default constructor so it only has default value

serialNumber: null

price: 0

builder: null

model: null

topWood: null

backWood: null

Obj2: is created by parameterized constructor so its attributes are initialized

serialNumber: "G123"

price: 2000

builder: "Sony"

model: "Model123"

topWood: "hardWood"

backWood: "softWood"

- Do you access all fields of obj1 in the class Tester.java? Why?

**Answer:**

You can only access indirectly to the fields of obj1 through getter/setter methods. The reason for this is encapsulation characteristic.

Encapsulation for object is need so it could ensure that the object data can only be modified as your desire and you can validate the data before allowing access/change

- What is the current object when the program runs to the line “obj2.createSound();”?

**Answer:**

At the line “obj2.createSound();” in the class Tester, the method createSound() defiened in the Guitar class is call in the obj2 and it used the data in obj2 and print out to the screen.

State of obj2:

serialNumber:G123

price:2000

builder:Sony

model:Model123

backWood:hardWood

topWood:softWood

- In the method main, can you use the keyword “this” to access all fields of obj2? Why?

**Answer:**

In the main method, you cannot use this to access fields of obj2 because main is static and does not have an associated instance.

If you want to access the fields of obj2 in main, you would need to refer to them directly through the object