Part 2 .

- Classes:

1. Guitar:

- Fields: serialNumber, price, builder, model,

backWood, topWood

- Methods: createSound

2. Inventory:

- Fields: list of guitars

- Methods: addGuitar, searchBySerialNumber

- UML:

|  |
| --- |
| Guitar |
| - serialNumber:String  - price : Double  - builder : String  - model : String  - blackWood : String  - topWood :srting |
| + createSound() : void |

|  |
| --- |
| Inventory |
| - guitar : List<guitar> |
| +addGuitar(guitar:Guitar):void  +searchBySerialNumber(serialNumber: String):Guitar |

Part 4 .

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

- Static heap, the class objects and static variables are

stored.

- Stack, the method calls, local variables, and object

references are stored.

- Dynamic heap, the objects are stored.

**- What are objects in the program?**

- The objects in the program are *obj1* and *obj2*.

**- What is the state of obj1, obj2?**

- The state of obj1 is empty values for all fields.

- The state of obj2 is the values assigned during its

constructor.

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

**Why?**

- No.

- Because *obj1*’s fields are private and we can access

all fields of *obj1* in its class.**- What is the current object when the program runs to the**

**line “obj2.createSound();”?**

- It’s *obj2*.

**- In the method main, can you use the keyword “this” to**

**access all fields of obj2? Why?**

- No.

- Reason 1: main is a static method.

- Reason 2: main is in Tester class, not in Guitar class.

Memory map :

|  |
| --- |
| String Pool |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Heap Guitar obj1   |  | | --- | | String serialNumber | | int price | | String builder | | String model | | String backWood | | String topWood | | | Guitar obj2   |  | | --- | | String serialNumber | | int price | | String builder | | String model | | String backWood |   String topWood | |
| Stack  main()   |  | | --- | | Guitar obj1 | | Guitar obj2 | |