CT3: CSE 203 (Sec B) - Object-Oriented Programming I: Java

Time: 25 min **Mark: 20**

Name: **ID:**

Question 1: **[12]**

Given the interface below, create a class named **House** that should implement the interface. Write minimum code in **House** class so that compiler doesn't give any error. Do not keep any method empty, at least write one meaningful line. Additionally, add the following to this **House** class.

a) Attributes: ***width*** *and* ***height***

b) Add a parameterized **constructor** and initialize both attributes.

```
package ct3.secB;

public interface Door {
	String material = "Wood";

	void openDoor();
	void closeDoor();
}
```

Question 2: **[8]**

Carefully observe the code below. Will the code execute successfully or give error? If it executes successfully, then write the output of the code. If it gives error, then explain what is the error and how to fix it.

```
public static void main(String[] args) {
	House myHouse = new House();
	myHouse.openDoor();
	myHouse.material = "RFL Plastic";
	myHouse.closeDoor();
}
```

CT3: CSE 203 (Sec B) - Object-Oriented Programming I: Java

Time: 25 min **Mark: 20**

Name: **ID:**

Question 1: **[12]**

Given the abstract class **Musician** below, create a subclass named "**Singer**". Write minimum code in **Singer** class so that compiler doesn't give any error. Additionally, add the following to this **Singer** class.

a) Attributes: ***noOfRecords***
b) Add a parameterized **constructor** which will take parameters for all 3 instance variables (2 from parent and one from this class). Implement the constructor in proper way.

```
package ct3.secB;

public abstract class Musician {
	private String name, field;

	public Musician(String name, String field) {
		this.name = name;
		this.field = field;
	}

	public abstract void play(String item);
	public abstract int noOfRecords();

	@Override
	public String toString() {
	    return String.format("name=%s, field=%s", name, field);
	}
}
```

Question 2: **[8]**

Carefully observe the class headers and the method calls below. The ***tryMe*** method is inside the class named **ClassTest3**. The method takes only **one parameter** and return an **int**. **Declare** the ***tryMe*** method such that all 3 calls below execute without error. Note: You just need to write the header of the ***tryMe*** method.

Class headers	Method calls

`class X extends A implements C,D`
`class Y extends B implements C,E`
`class Z extends B implements C`
	`// The tryMe method is inside ClassTest3 class.`
	`int a = tryMe(new X());`
	`int b = tryMe(new Y());`
	`int c = tryMe(new Z());`