

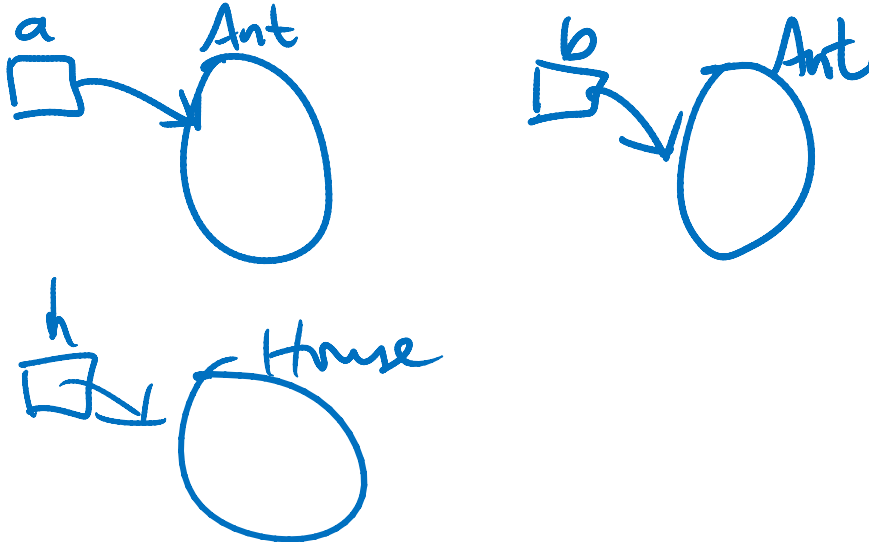
Suppose the following lines of Java code:

```
Ant a = new Ant();
```

```
Ant b = new Ant();
```

```
House h = new House();
```

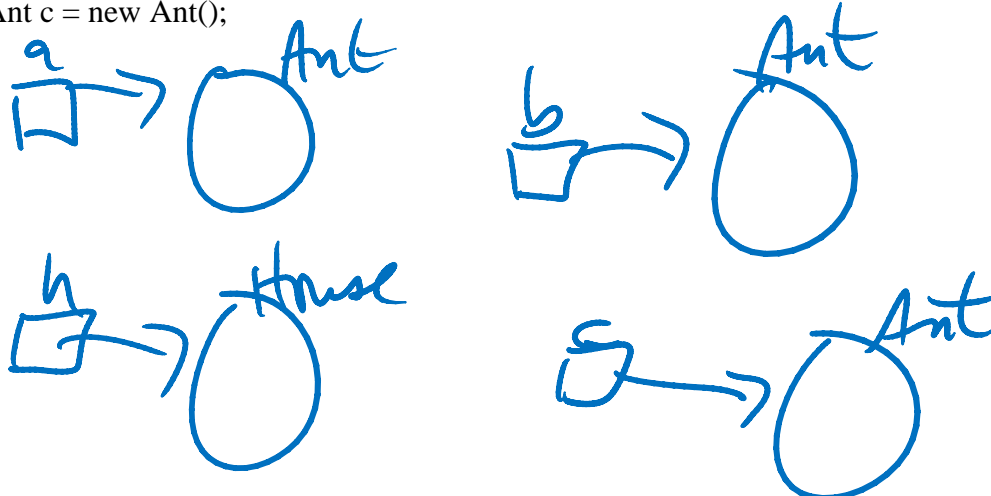
Draw the memory diagram showing the references and objects for these lines of code.



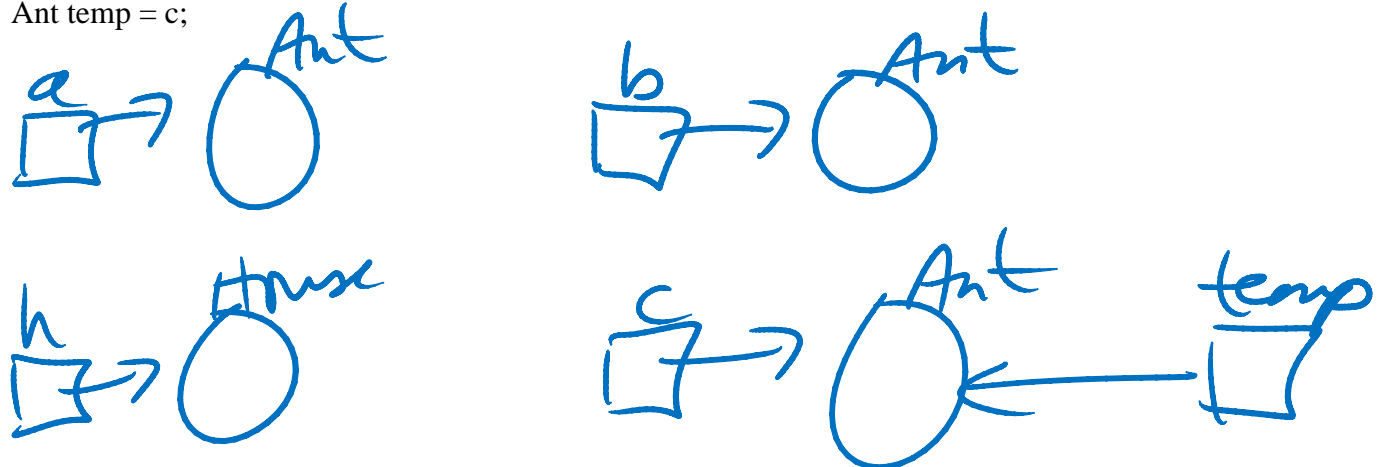
Now suppose we add the following lines:

Show the appropriate changes in the memory diagram after each line is executed.

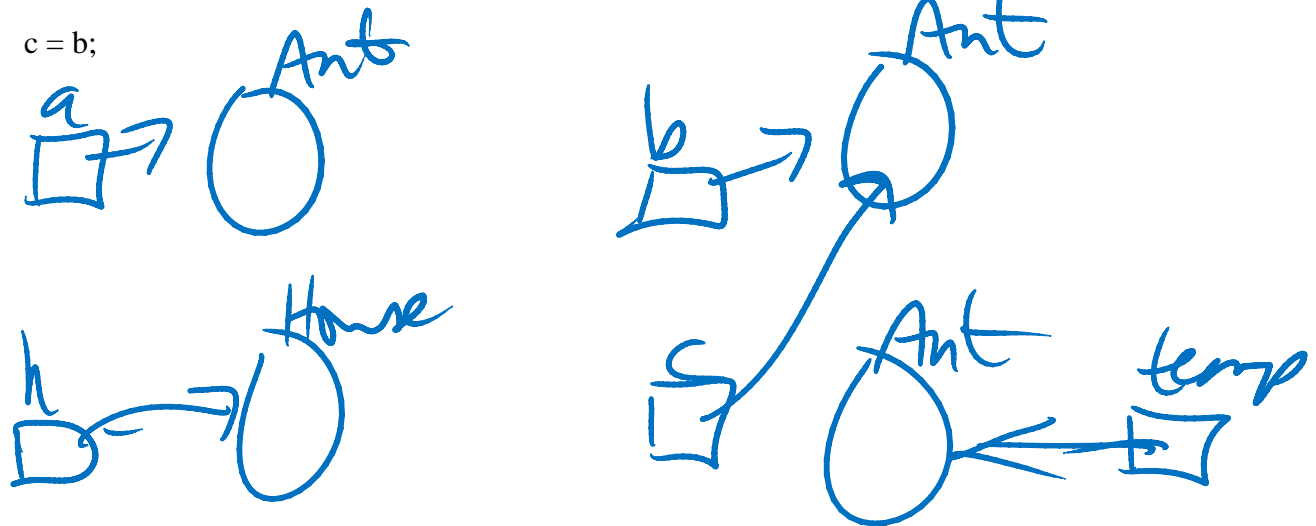
```
Ant c = new Ant();
```



```
Ant temp = c;
```



c = b;



b = temp;



a = temp;



Write the Java source code for a class named Question1 that lives in a package named exam.

```
package exam;
```

```
public class Question1 {
```

```
    public Question1() {
```

```

    }
}

```

Given a class definition (Java Source Code), circle and identify one example of the following:

- a) Class definition
- b) Class header
- c) Class body
- d) Class name
- e) Package declaration
- f) Package name
- g) Keyword
- h) Constructor definition
- i) Constructor header
- j) Constructor body
- k) Parameter list
- l) Argument list
- m) Method call
- n) Creation of an object
- o) Local variable name
- p) Assignment statement

```

package example1;

/**
 * A collection of bugs that live in a terrarium.
 *
 * @author
 * Created on: Feb 8, 2010
 */
public class MyBugCollection {

    /**
     * Creates a new instance of MyBugCollection
     */
    public MyBugCollection() {
        example1.Terrarium terrarium =
            new example1.Terrarium();

        example1.Ant ant1;
        ant1 = new example1.Ant();
        ant1.start();
        terrarium.add(ant1);
        terrarium.add(new example1.Ant());

        example1.Caterpillar caterpillar =
            new example1.Caterpillar();
        caterpillar.start();
        terrarium.add(caterpillar);

        new Caterpillar();
        new Caterpillar();
    }
}

```

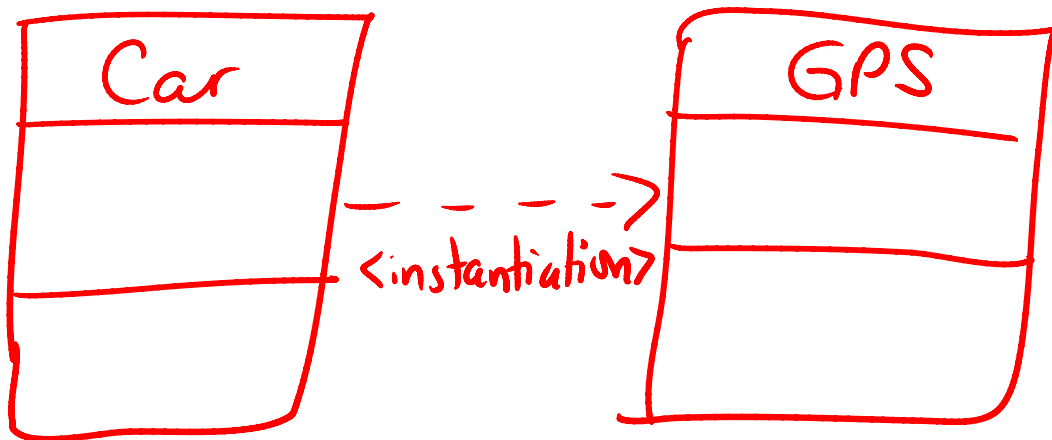
What is the scope of a local variable?

Scope is where the variable is accessible. The scope of a local variable is from the point of declaration to the end of the method body.

What is the difference between local variable dependency and instantiation dependency?

In both dependency relationships, the Source creates an instance of the Target, but in local variable dependency, there is a local variable created and in instantiation dependency, there is not.

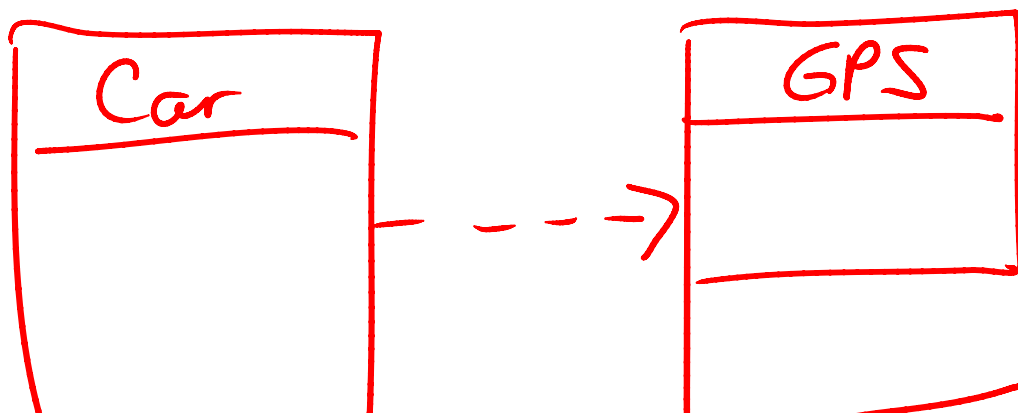
Draw the UML diagram for the relationship: Car uses a GPS. (instantiation)



Write the source code for the above relationship. (using instantiation)

```
public class Car {  
    public Car() {  
        new GPS();  
    }  
}
```

Draw the UML diagram for the relationship: Car uses a GPS. (local variable)



_____ {local variable}

Write the source code for the above relationship. (using local variable)

```
public class Car {  
    public Car() {  
        GPS gps = new GPS();  
    }  
}
```