

# Code Tracing: Assignment Statement

Given the following code

```
double x, y, z;
Human persona, role, part;

x = 3.1;
y = 4.2;
z = x;

persona = new Human("Leela");
role = new Human("Frye");
part = persona;
// LOCATION 1

x = 8.2;
y = x;
persona.setName("Hermes");
// LOCATION 2

persona = new Human("Amy");
// LOCATION 3

role = persona;
// LOCATION 4

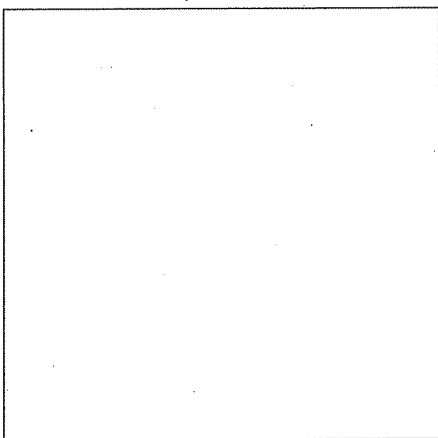
persona.haveBirthday();
// LOCATION 5

System.out.println(x);
System.out.println(y);
System.out.println(z);

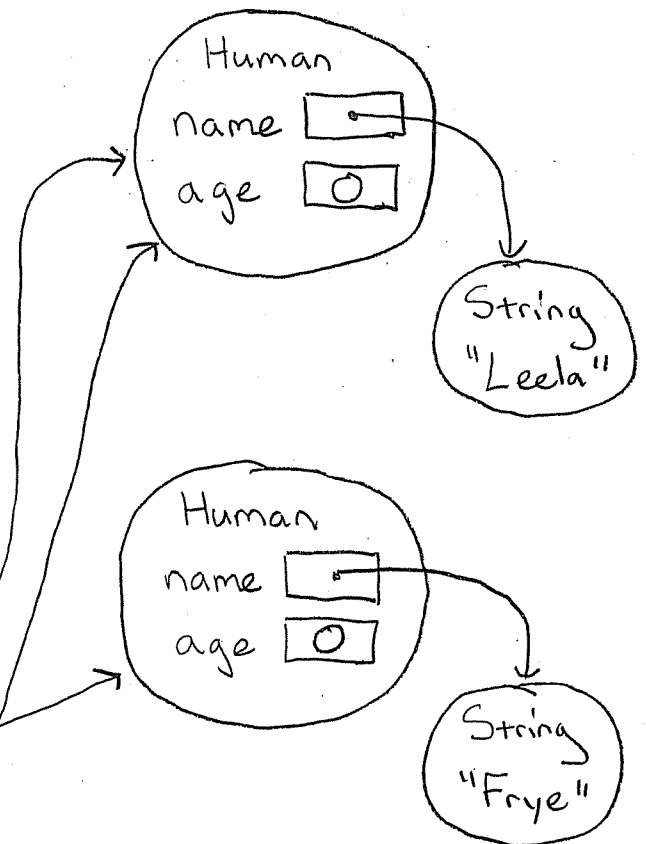
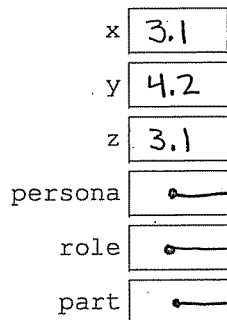
System.out.println(persona.getName());
System.out.println(role.getAge());
System.out.println(part.getName());
```

What is printed out?

Console window:



Memory diagram when execution reaches LOCATION 1



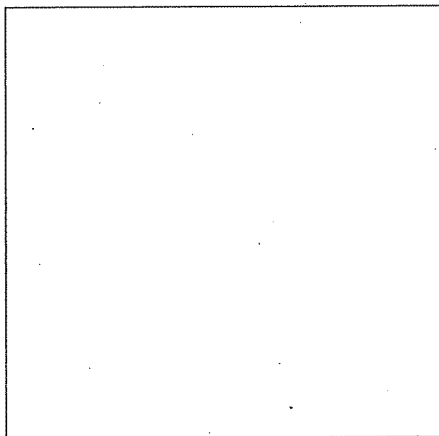
# Code Tracing: Assignment Statement

Given the following code

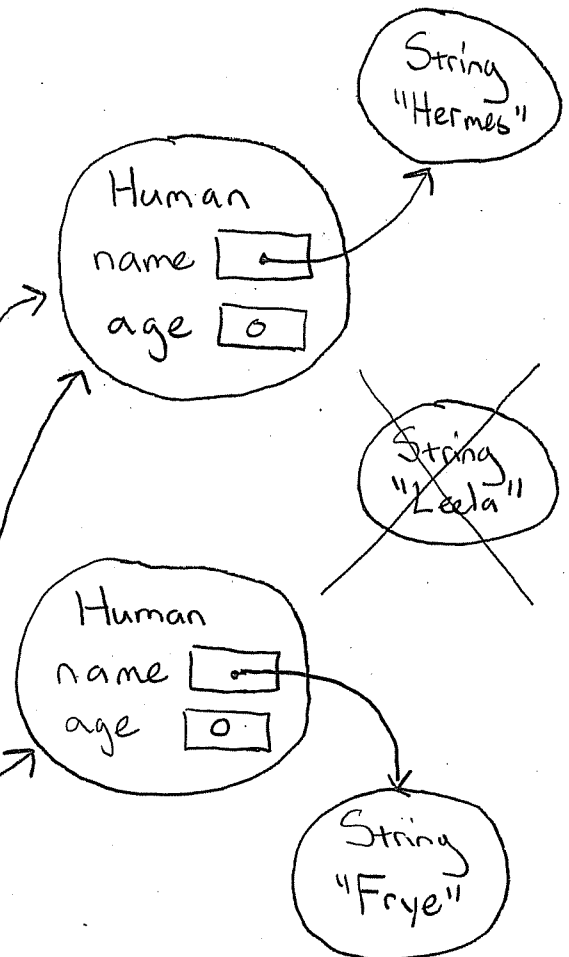
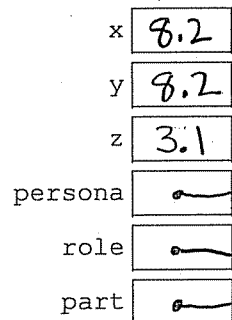
```
double x, y, z;  
Human persona, role, part;  
  
x = 3.1;  
y = 4.2;  
z = x;  
  
persona = new Human("Leela");  
role = new Human("Frye");  
part = persona;  
// LOCATION 1  
  
x = 8.2;  
y = x;  
persona.setName("Hermes");  
// LOCATION 2  
  
persona = new Human("Amy");  
// LOCATION 3  
  
role = persona;  
// LOCATION 4  
  
persona.haveBirthday();  
// LOCATION 5  
  
System.out.println(x);  
System.out.println(y);  
System.out.println(z);  
  
System.out.println(persona.getName());  
System.out.println(role.getAge());  
System.out.println(part.getName());
```

What is printed out?

Console window:



Memory diagram when execution reaches LOCATION 2



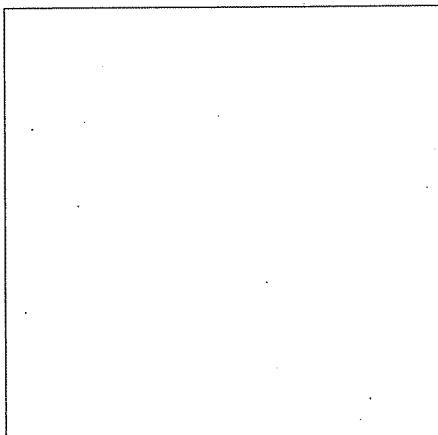
# Code Tracing: Assignment Statement

Given the following code

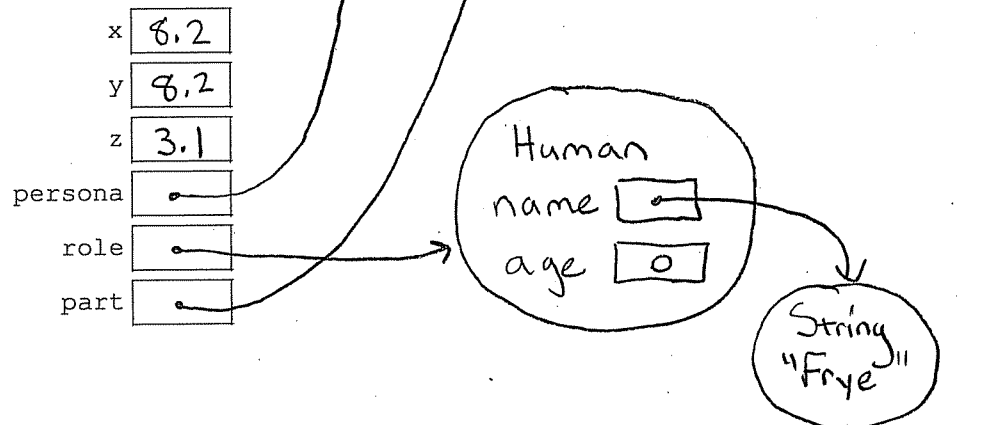
```
double x, y, z;  
Human persona, role, part;  
  
x = 3.1;  
y = 4.2;  
z = x;  
  
persona = new Human("Leela");  
role = new Human("Frye");  
part = persona;  
// LOCATION 1  
  
x = 8.2;  
y = x;  
persona.setName("Hermes");  
// LOCATION 2  
  
persona = new Human("Amy");  
// LOCATION 3  
  
role = persona;  
// LOCATION 4  
  
persona.haveBirthday();  
// LOCATION 5  
  
System.out.println(x);  
System.out.println(y);  
System.out.println(z);  
  
System.out.println(persona.getName());  
System.out.println(role.getAge());  
System.out.println(part.getName());
```

What is printed out?

Console window:



Memory diagram when execution reaches LOCATION 3



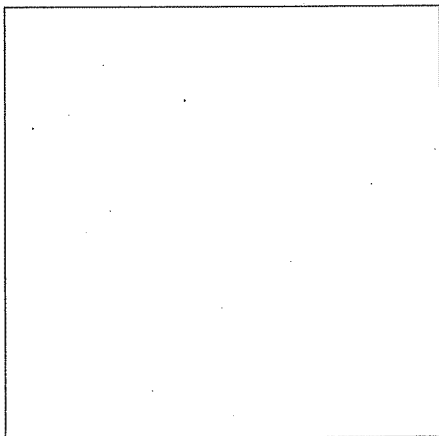
# Code Tracing: Assignment Statement

Given the following code

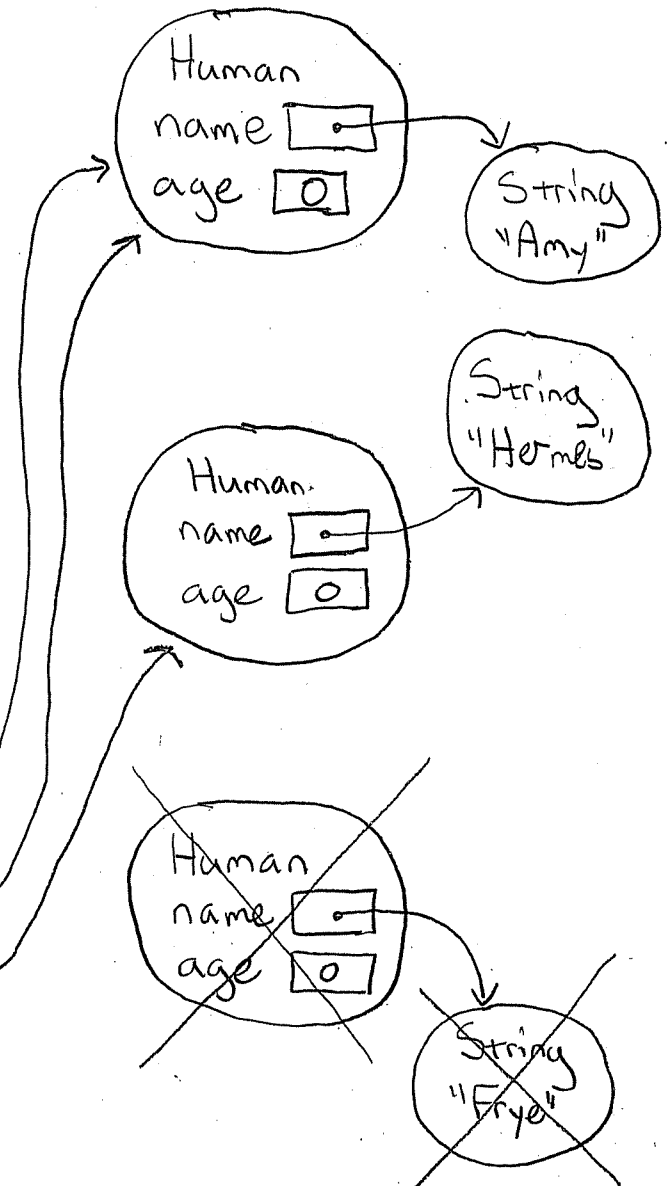
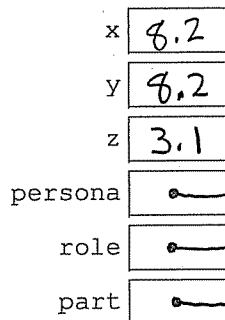
```
double x, y, z;  
Human persona, role, part;  
  
x = 3.1;  
y = 4.2;  
z = x;  
  
persona = new Human("Leela");  
role = new Human("Frye");  
part = persona;  
// LOCATION 1  
  
x = 8.2;  
y = x;  
persona.setName("Hermes");  
// LOCATION 2  
  
persona = new Human("Amy");  
// LOCATION 3  
  
role = persona;  
// LOCATION 4  
  
persona.haveBirthday();  
// LOCATION 5  
  
System.out.println(x);  
System.out.println(y);  
System.out.println(z);  
  
System.out.println(persona.getName());  
System.out.println(role.getAge());  
System.out.println(part.getName());
```

What is printed out?

Console window:



Memory diagram when execution reaches LOCATION 4



# Code Tracing: Assignment Statement

Given the following code

```
double x, y, z;  
Human persona, role, part;  
  
x = 3.1;  
y = 4.2;  
z = x;  
  
persona = new Human("Leela");  
role = new Human("Frye");  
part = persona;  
// LOCATION 1  
  
x = 8.2;  
y = x;  
persona.setName("Hermes");  
// LOCATION 2  
  
persona = new Human("Amy");  
// LOCATION 3  
  
role = persona;  
// LOCATION 4  
  
persona.haveBirthday();  
// LOCATION 5  
  
System.out.println(x);  
System.out.println(y);  
System.out.println(z);  
  
System.out.println(persona.getName());  
System.out.println(role.getAge());  
System.out.println(part.getName());
```

**What is printed out?**  
(after code executes)

Console window:

```
8.2  
8.2  
3.1  
Amy  
1  
Hermes
```

Memory diagram when execution reaches LOCATION 5

