

CSCI 132:

Basic Data Structures and Algorithms

Lecture 7: References, Program 1

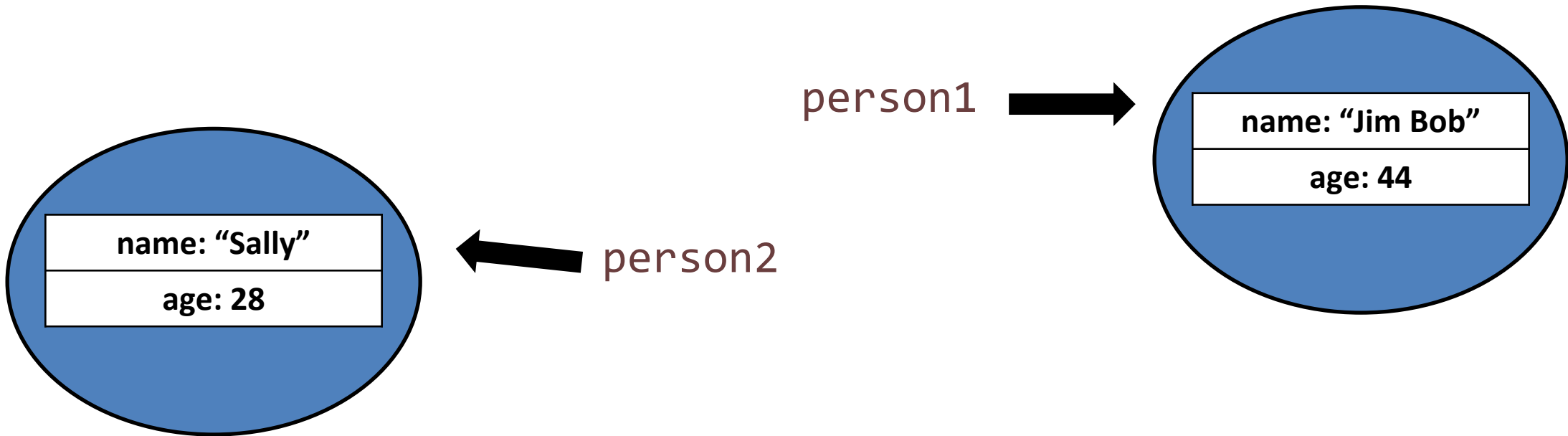
Reese Pearsall
Fall 2023

Announcements

- Program 1 posted, due 9/18 @ 11:59 PM

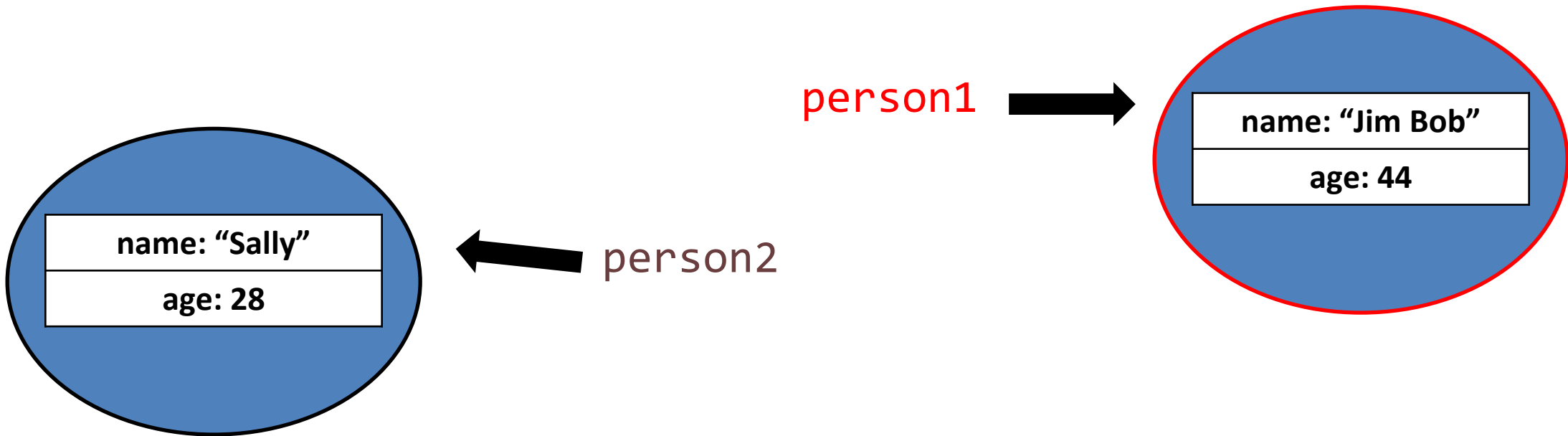
```
public class ReferencesDemo {  
    public static void main(String[] args) {  
  
        Person person1 = new Person("Jim Bob", 44);  
        Person person2 = new Person("Sally", 28);  
  
    }  
}
```

person1 and person2 are references to a Person object



```
public class ReferencesDemo {  
    public static void main(String[] args) {  
  
        Person person1 = new Person("Jim Bob", 44);  
        Person person2 = new Person("Sally", 28);  
        person1.changeName("Jack");  
  
    }  
}
```

person1 and person2 are references to a Person object



```

public class ReferencesDemo {
    public static void main(String[] args) {

        Person person1 = new Person("Jim Bob", 44);
        Person person2 = new Person("Sally", 28);
        person1.changeName("Jack");

    }
}

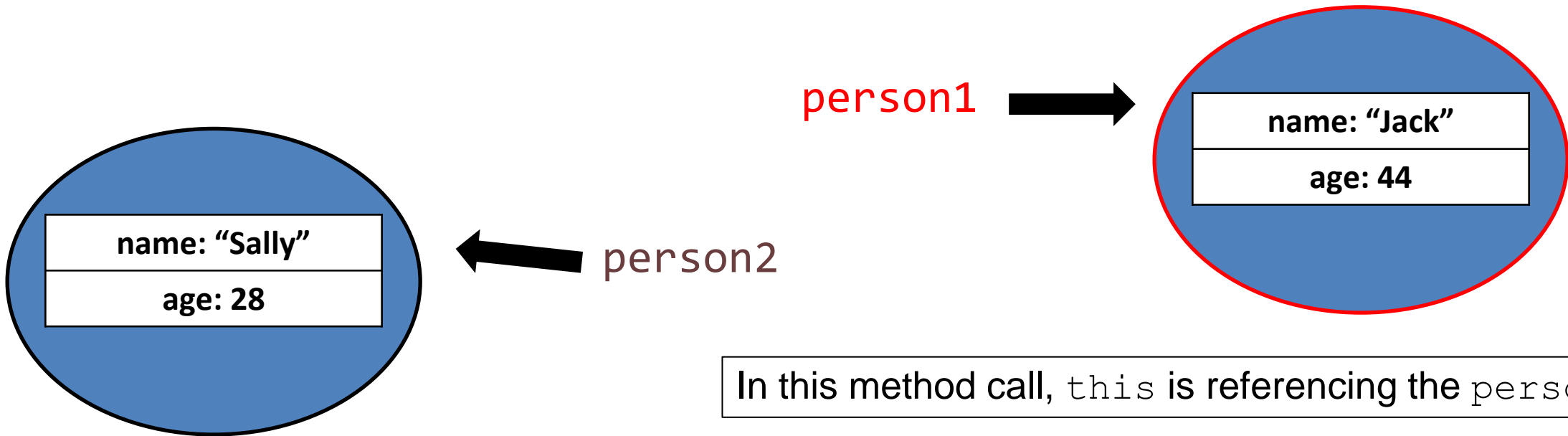
```

```

public void changeName(String newName) {
    this.name = newName;
}

```

person1 and person2 are references to a Person object



In this method call, `this` is referencing the `person1` object

```

public class ReferencesDemo {
    public static void main(String[] args) {

        Person person1 = new Person("Jim Bob", 44);
        Person person2 = new Person("Sally", 28);

        Person person3 = person1;

    }
}

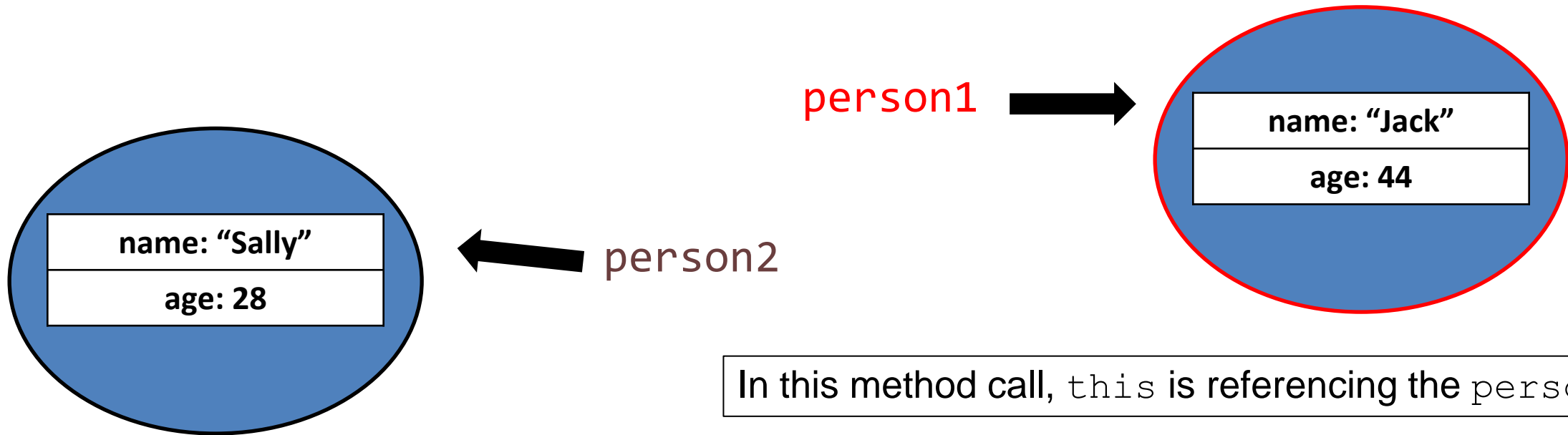
```

Suppose we create a new reference variable and link it to an existing object

```

public void changeName(String newName) {
    this.name = newName;
}

```



In this method call, `this` is referencing the `person1` object

```

public class ReferencesDemo {
    public static void main(String[] args) {

        Person person1 = new Person("Jim Bob", 44);
        Person person2 = new Person("Sally", 28);

        Person person3 = person1;

    }
}

```

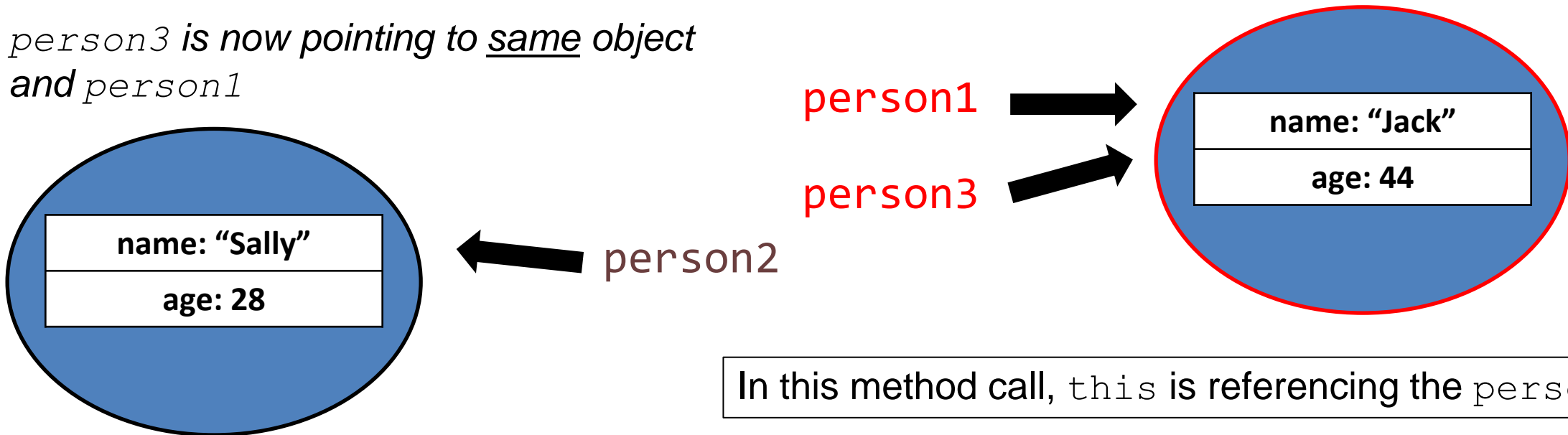
```

public void changeName(String newName) {
    this.name = newName;
}

```

Suppose we create a new reference variable and link it to an existing object

person3 is now pointing to same object and person1



In this method call, `this` is referencing the `person1` object

```

public class ReferencesDemo {
    public static void main(String[] args) {

        Person person1 = new Person("Jim Bob", 44);
        Person person2 = new Person("Sally", 28);

        Person person3 = person1;
        person1.changeName("test");

    }
}

```

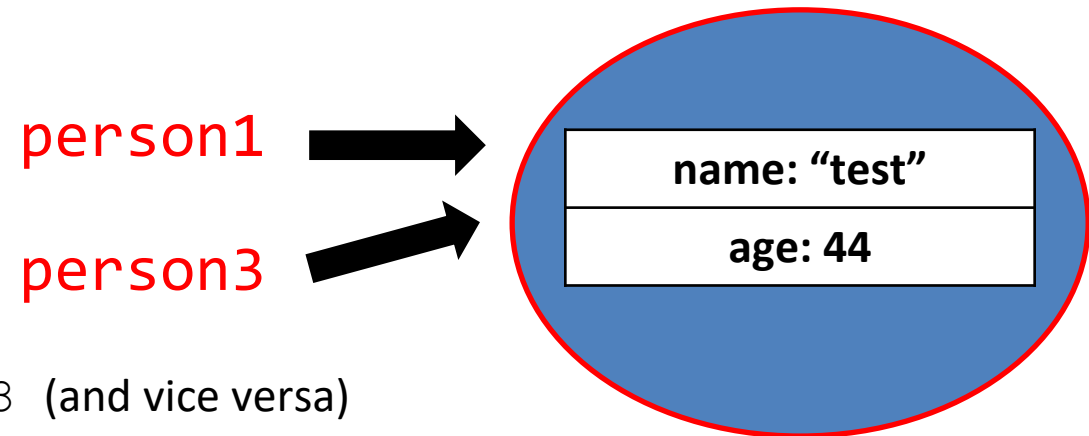
```

public void changeName(String newName) {
    this.name = newName;
}

```

Suppose we create a new reference variable and link it to an existing object

person3 is now pointing to same object and person1



Any changes to person1 will also update person3 (and vice versa)

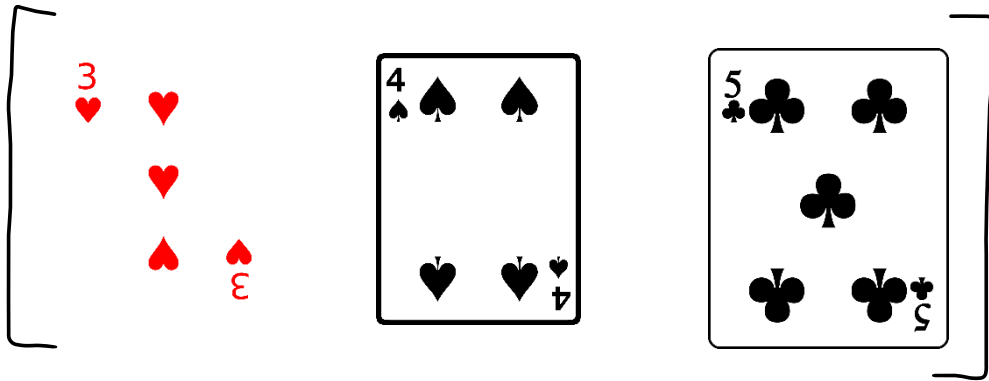
System.out.println(person1.getName()) → "test"

System.out.println(person3.getName()) → "test"

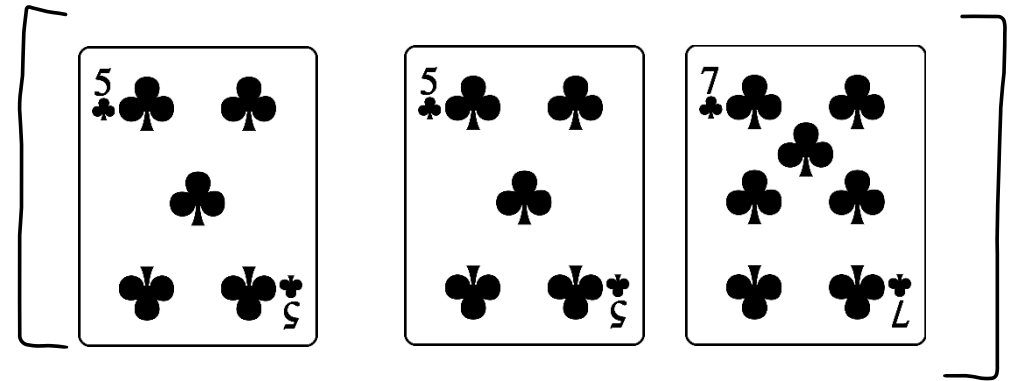
Program 1

Write a program that will evaluate a three card poker hand. Your program should be able to identify

1. Three of a kind
2. Flush
3. Sequence
4. Two of a kind



Sequence



Flush
Two of a kind