Model 1 Objects

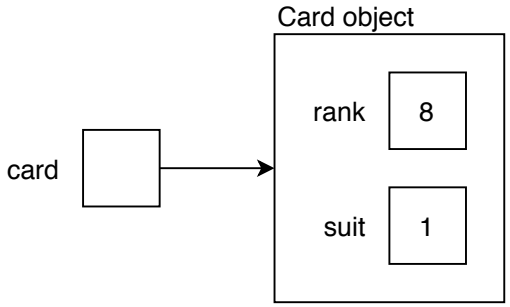
Consider the definition for a playing card:

```
public class Card {
    private int rank;  // 1=Ace, ..., 11=Jack, 12=Queen, 13=King
    private int suit;  // 0=Clubs, 1=Diamonds, 2=Hearts, 3=Spades

    public Card(int rank, int suit) {
        this.rank = rank;
        this.suit = suit;
    }
}
```

Here is a memory diagram of a `Card` object:

```
Card card = new Card(8, 1);
```



Questions (15 min) Start time:

1. Which card (i.e., "the _____of _____") is represented in the diagram?

The 8 of Diamonds.

2. In one line of code, show how you would construct the "4 of Clubs".

```
Card card = new Card(4, 0);
```

3. What is the difference between lowercase `card` and uppercase `Card`? Explain in a few sentences how these concepts are illustrated in the diagram.

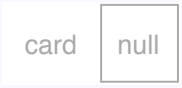
Lowercase `card` is a variable that contains a reference. Uppercase `Card` is a class, from which we create objects. Variables are small boxes; objects are large boxes that contain variables.

4. How are arrays and objects similar? How are arrays and objects different? Explain your answer in terms of how they are drawn in memory diagrams.

Both are reference types; their variables have an arrow pointing somewhere else. Arrays are drawn as contiguous boxes, representing multiple values of the same type. Objects are drawn as larger boxes, representing multiple variables by name.

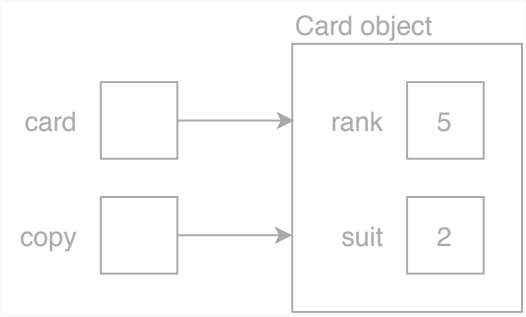
5. Draw (or describe) a diagram of the following source code:

```
Card card = null;
```



6. Draw (or describe) a diagram of the following source code:

```
Card card = new Card(5, 2);
Card copy = card;
```



7. (Optional) Paste the contents of *Card.java* into Java Visualizer. What differences do you notice between the diagram in Java Visualizer and those in Model 1?

Answers might include:

- JV says "Card instance" instead of "Card object".
- The object looks like a table, not separate boxes.