1 Predict the final value

1. One thread executes

   ```
   v = 1;
   v = v + 1;
   ```

   and another thread executes

   ```
   v = 0;
   ```

   What is the final value of v?

2. One thread executes

   ```
   v = v + 1;
   ```

   and another thread executes

   ```
   v = v + 1;
   ```

   If the initial value of v is 0, then what is the final value of v?

3. One thread executes

   ```
   v = 0;
   ```

   and another thread executes

   ```
   v = Long.MAX_VALUE;
   ```

   How many different final values can v have?

2 Printer (Thread)

- **`public Thread(String name)`**
  Initializes a new **`Thread`** object with the specified name as its name.
- **`public void start()`**
  Causes this thread to begin execution; the Java virtual machine calls the run method of this thread.
- **`public void run()`**
  This method does nothing and returns.

Develop a Java class called **`Printer`** that is a **`Thread`** and prints its name 1000 times.

```
public class




}
```

3 Two printers

Develop an app that creates two **`Printer`**s with names 1 and 2 and run them concurrently.

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









  }
}
```

4 Printer (Runnable)

Develop a Java class called **`Printer`** that implements **`Runnable`** and prints the thread's name 1000 times.

```
public class Printer implements Runnable {









}
```

5 Two printers

Develop an app that creates two **`Printer`**s with names 1 and 2 and run them concurrently.

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









  }
}
```

6 Incrementers

Develop a Java class called **`Incrementer`** that is a **`Thread`** and increments a shared static attribute named **`value`**, which is initialized to 0.

```
public class Incrementer extends Thread {









}
```

Develop an app that creates two **`Incrementer`**s and run them concurrently. Assert that the final value of **`value`** is two.

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









  }
}
```