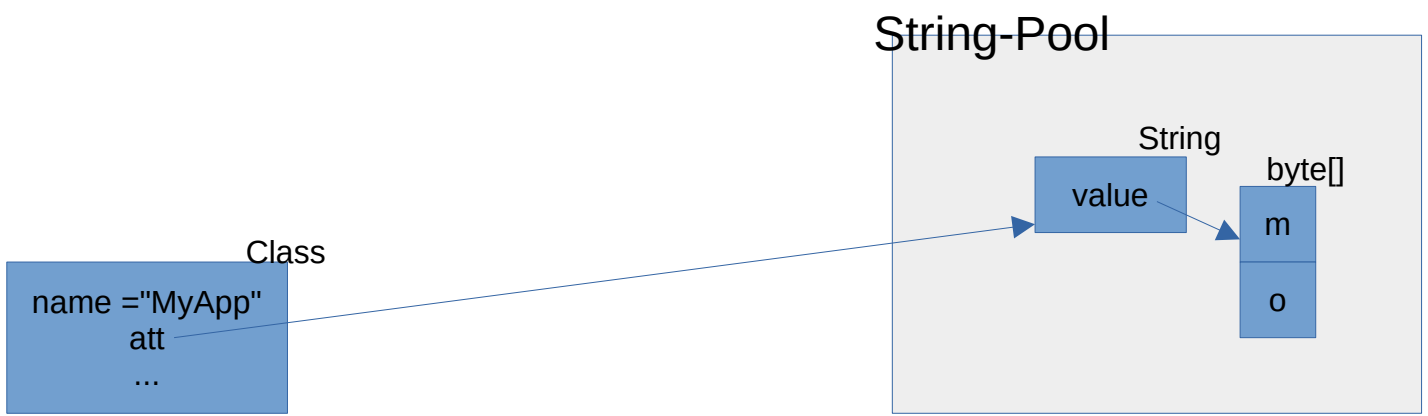
Bsp. 1

Anwendung wird so gestartet:

java MyApp

Die JVM lädt die Klasse MyApp.
Die JVM erweitert dabei den String-Pool mit den Strings, die aus Literalen der Klasse entstehen.
Das Class-Objekt der Klasse MyApp wird auch erzeugt.

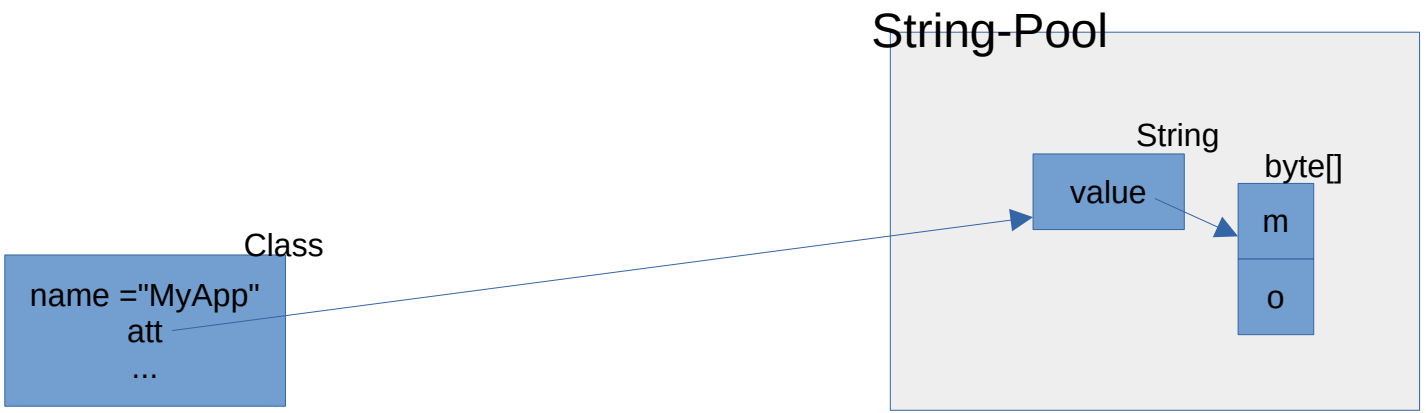
```
class MyApp {
   …main… {

      m();
   }

   static String m() {
      String s = "mo";
      String s2 = s;
      String s3 = "mo";
      String s4 = att;
      return s.toString();
   }

   static String att = "mo";
}
```



Jetzt kann die JVM die main-Methode starten

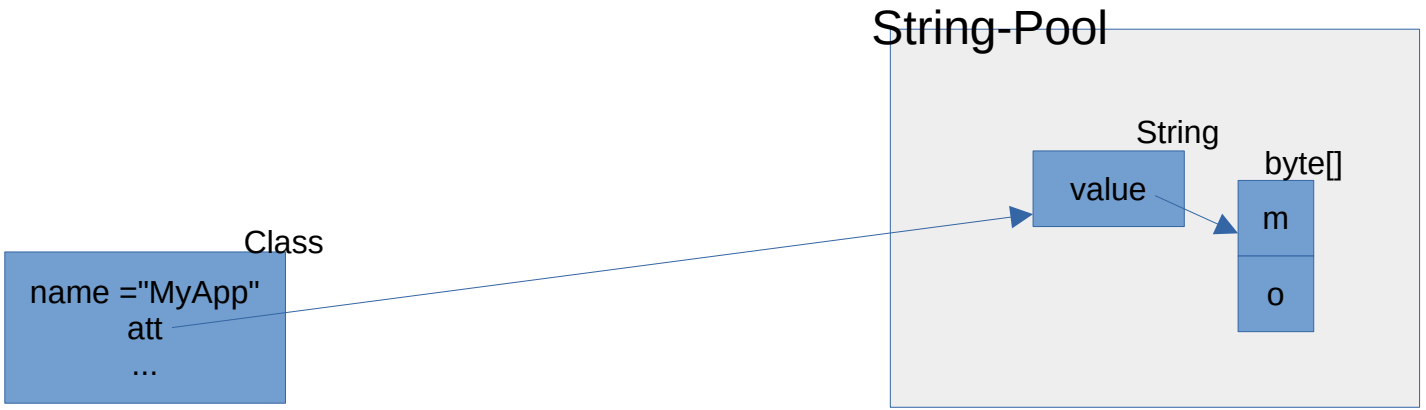
STACK

```
class MyApp {
   …main… {

      m();
   }

   static String m() {
      String s = "mo";
      String s2 = s;
      String s3 = "mo";
      String s4 = att;
      return s.toString();
   }

   static String att = "mo";
}
```

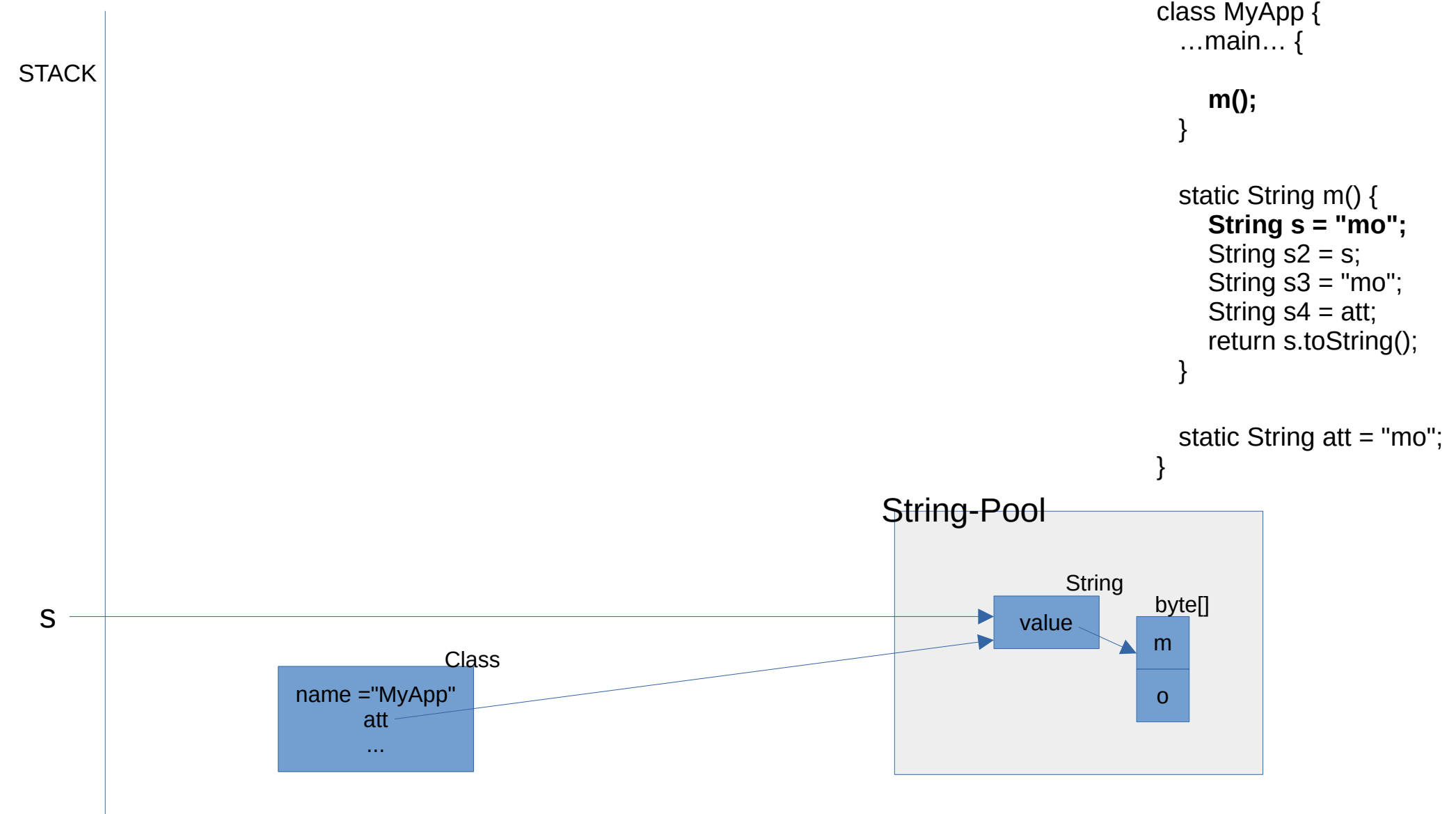


STACK

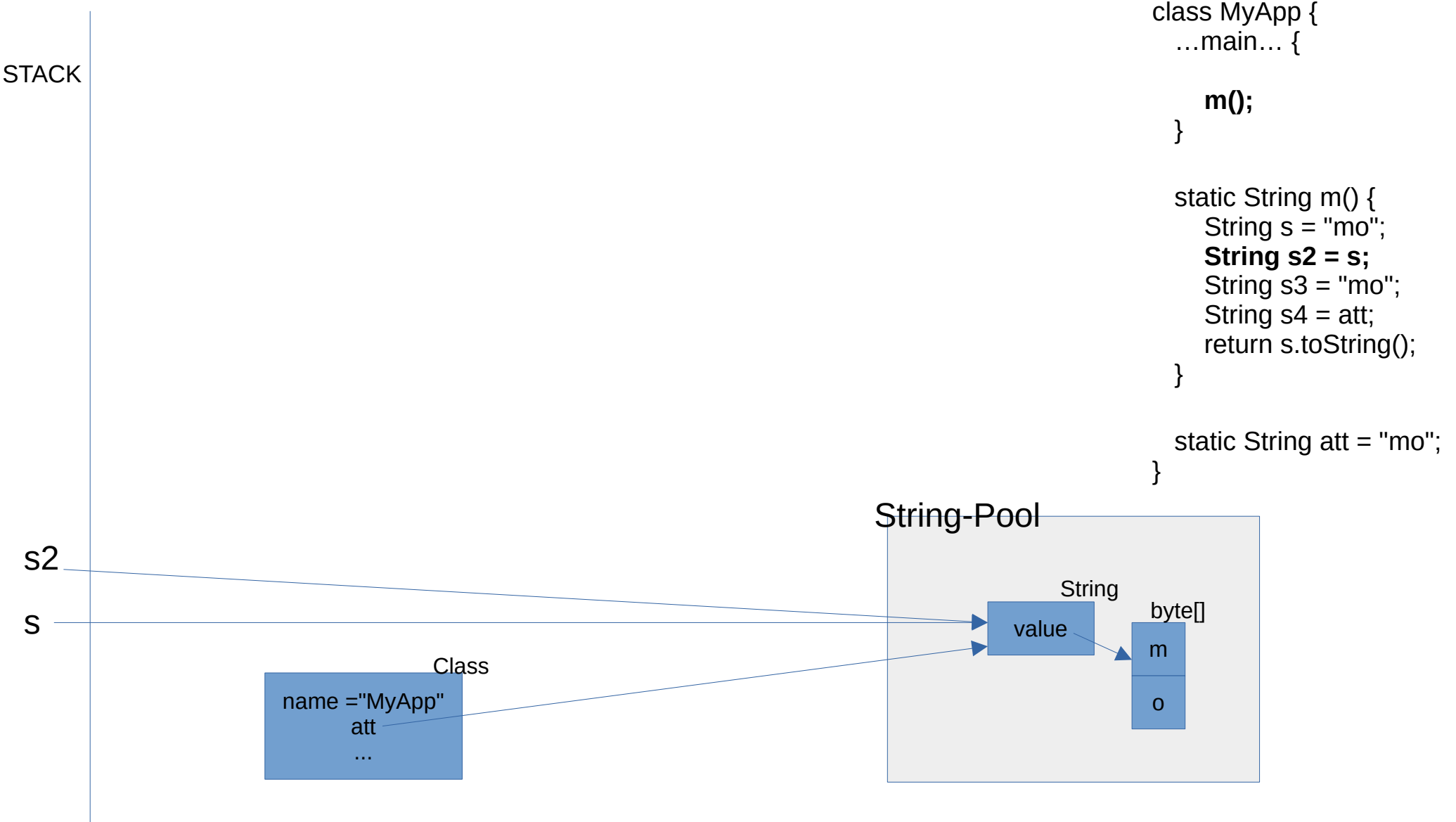
```
class MyApp {
   …main… {

      m();
   }

   static String m() {
      String s = "mo";
      String s2 = s;
      String s3 = "mo";
      String s4 = att;
      return s.toString();
   }

   static String att = "mo";
}
```



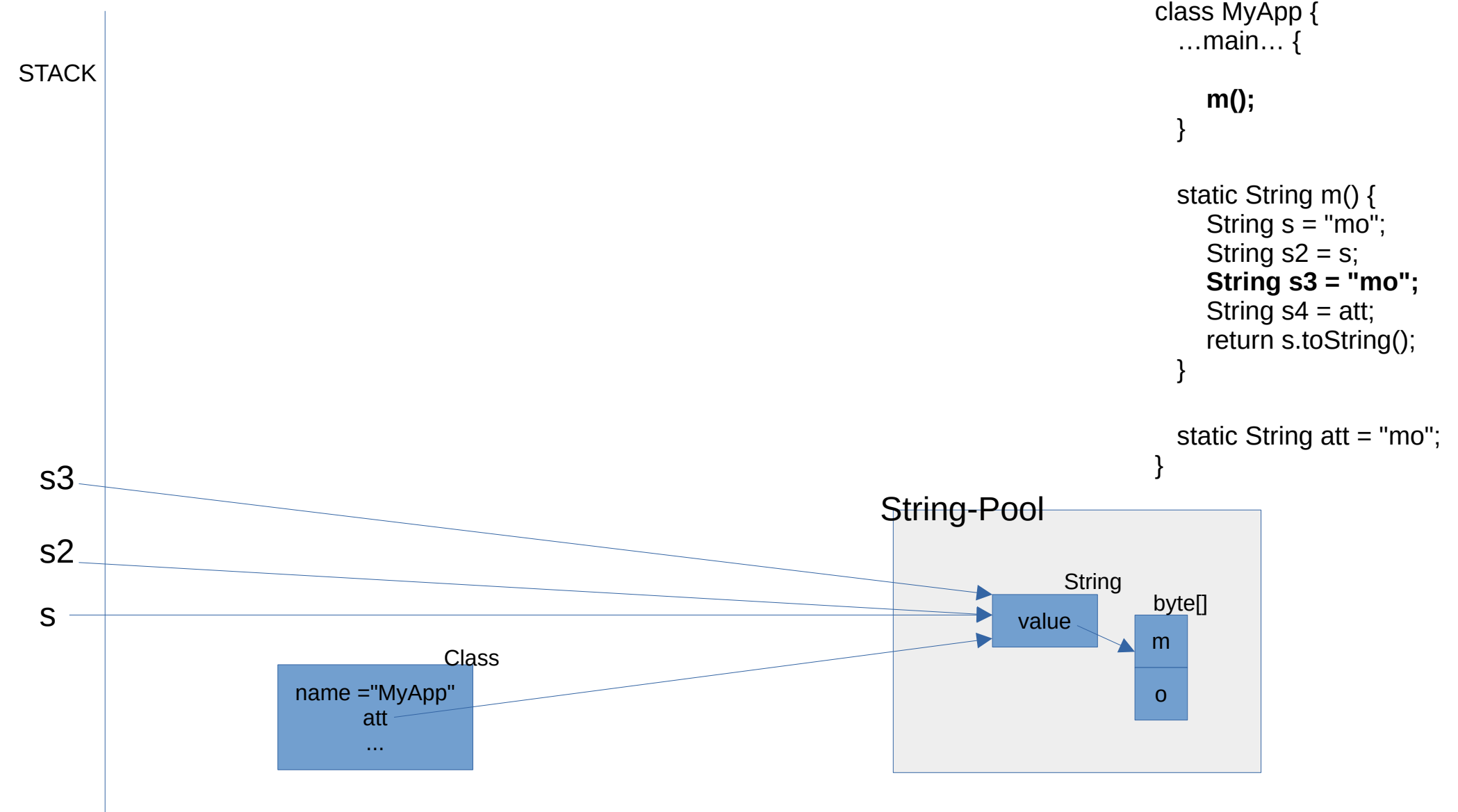
```
class MyApp {
   …main… {

      m();
   }

   static String m() {
      String s = "mo";
      String s2 = s;
      String s3 = "mo";
      String s4 = att;
      return s.toString();
   }

   static String att = "mo";
}
```



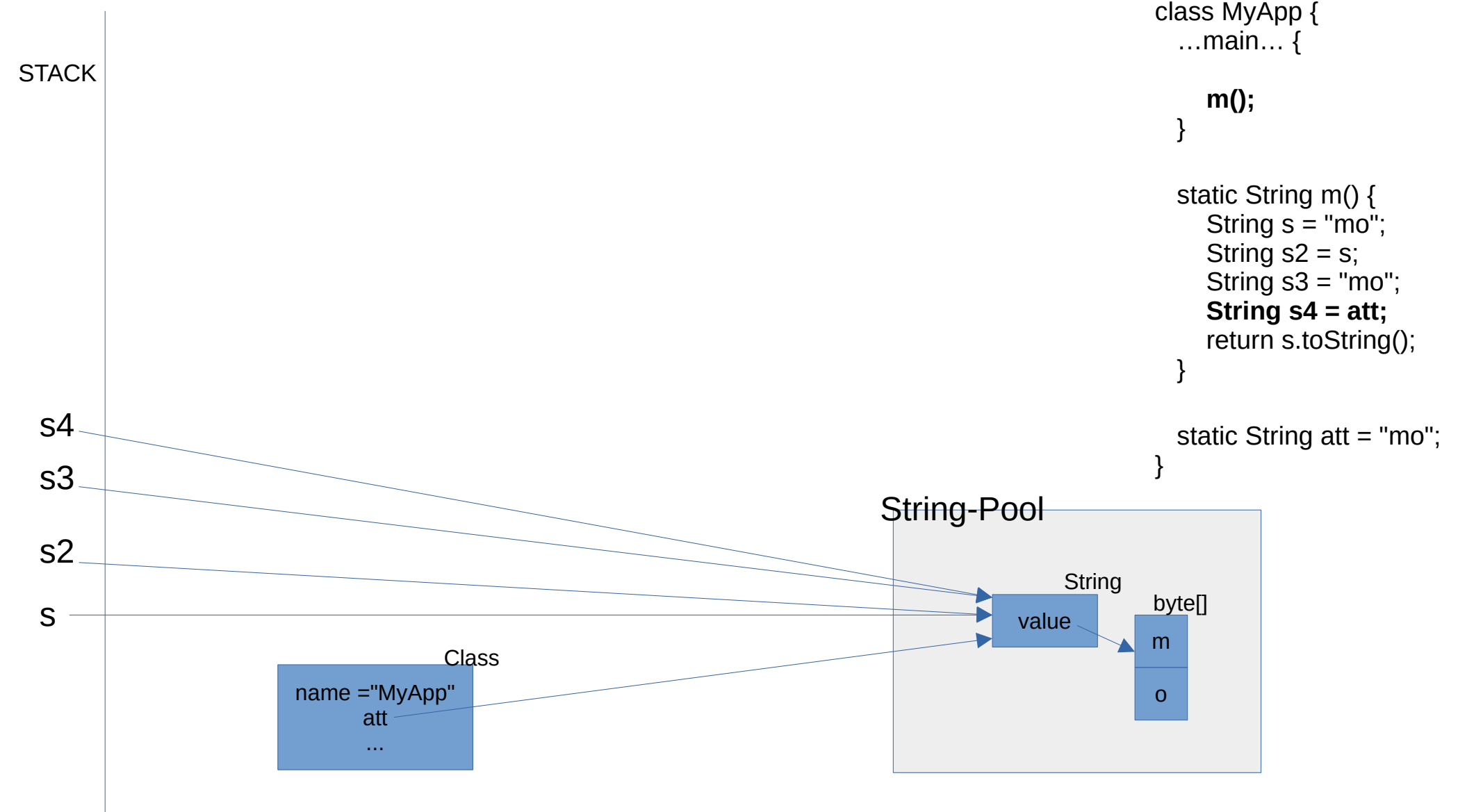
```
class MyApp {
   …main… {

      m();
   }

   static String m() {
      String s = "mo";
      String s2 = s;
      String s3 = "mo";
      String s4 = att;
      return s.toString();
   }

   static String att = "mo";
}
```



```
class MyApp {
   …main… {

      m();
   }

   static String m() {
      String s = "mo";
      String s2 = s;
      String s3 = "mo";
      String s4 = att;
      return s.toString();
   }

   static String att = "mo";
}
```



```
class MyApp {
   …main… {

      m();
   }

   static String m() {
      String s = "mo";
      String s2 = s;
      String s3 = "mo";
      String s4 = att;
      return s.toString();
   }

   static String att = "mo";
}
```

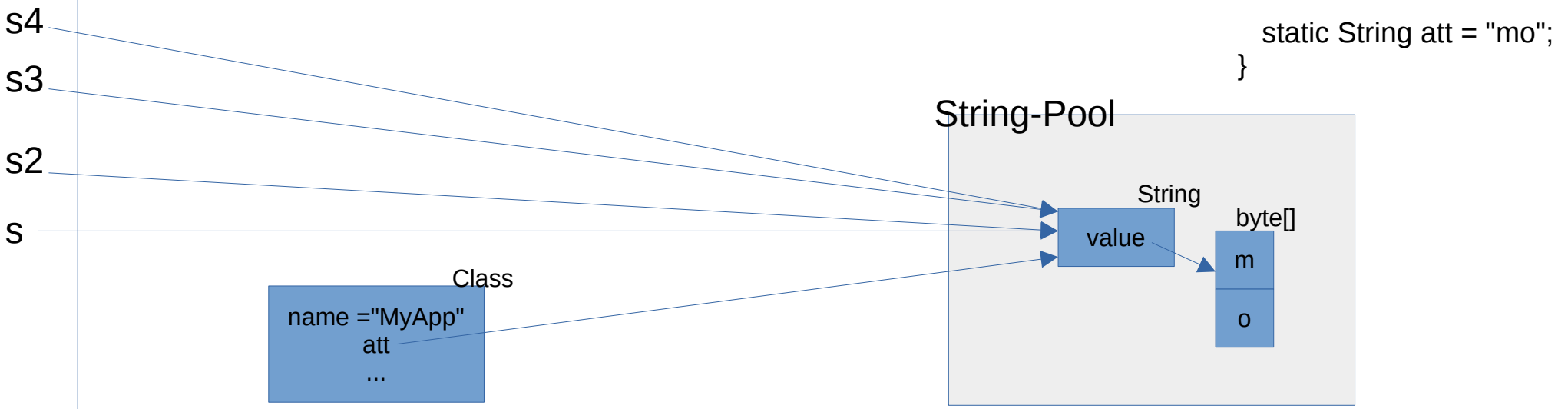


STACK

toString in der Klasse String ist so realisiert:

```
public String toString() {
    return this;
}
```

```
class MyApp {
   …main… {

      m();
   }

   static String m() {
      String s = "mo";
      String s2 = s;
      String s3 = "mo";
      String s4 = att;
      return s.toString();
   }

   static String att = "mo";
}
```

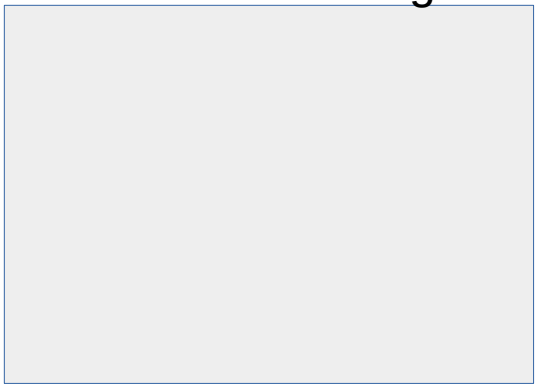


Bsp. 2

STACK HEAP

```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```

String-Pool

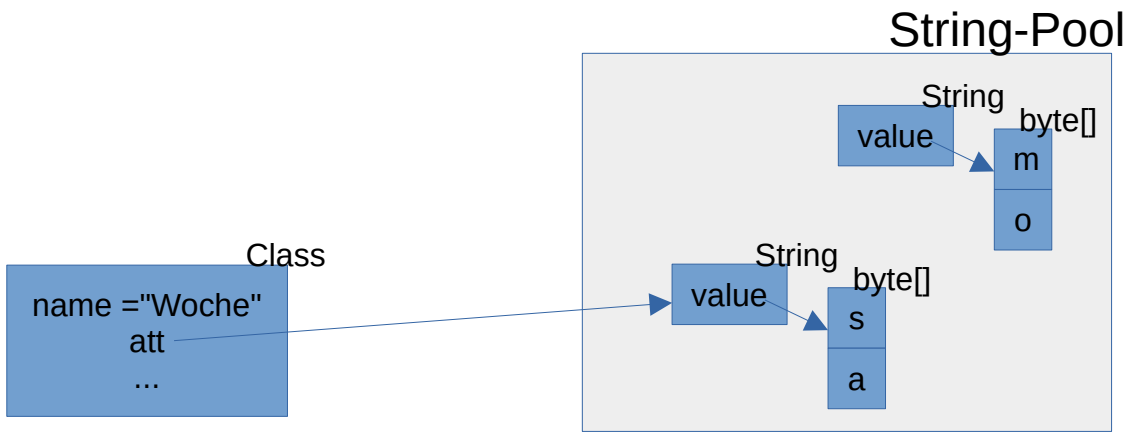


STACK HEAP

Klasse Woche wird geladen.
String-Pool wird mit den Strings aus
den Literalen dieser Klasse vervollständigt.

Nachdem das Class-Objekt für die Klasse Woche
erzeugt wurde (Klasse Woche wurde geladen),
startet die Methode m()

```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```

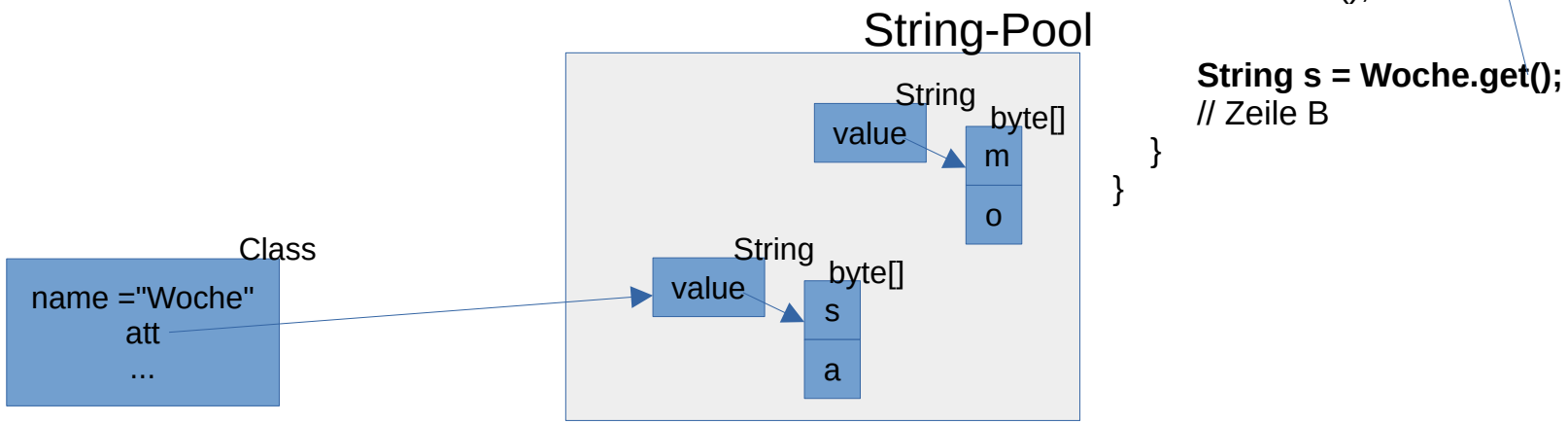


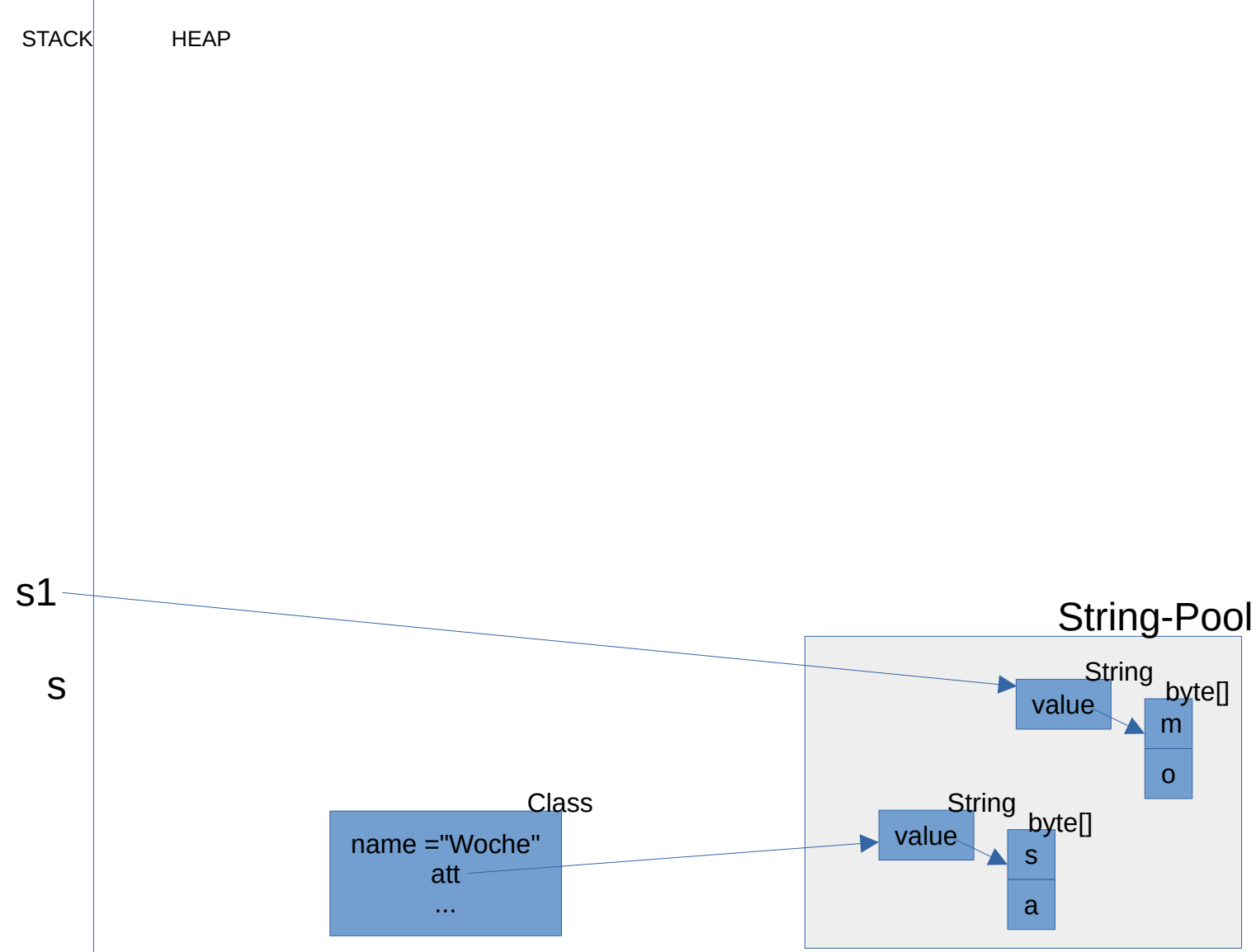
STACK

HEAP

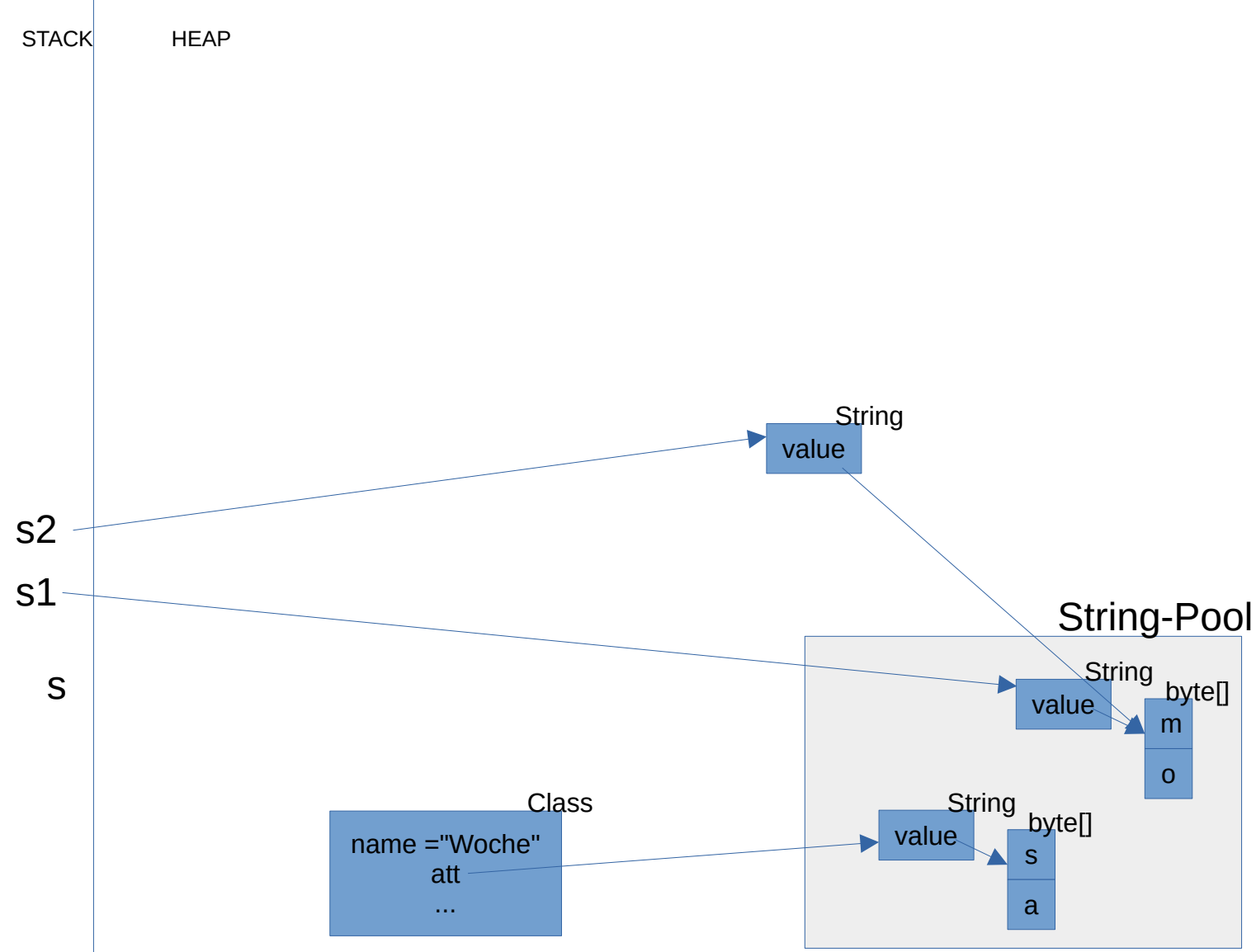
S

```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```

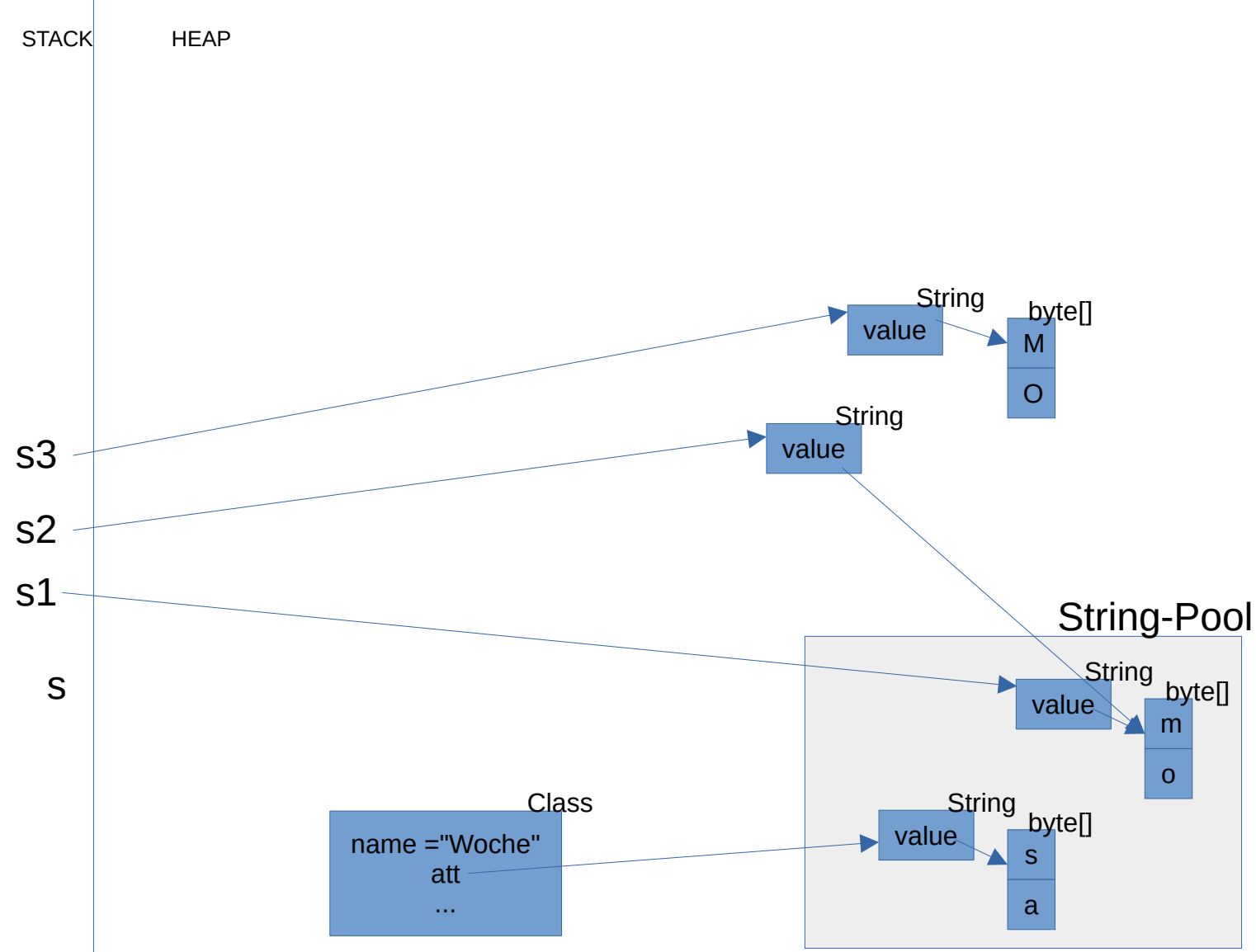




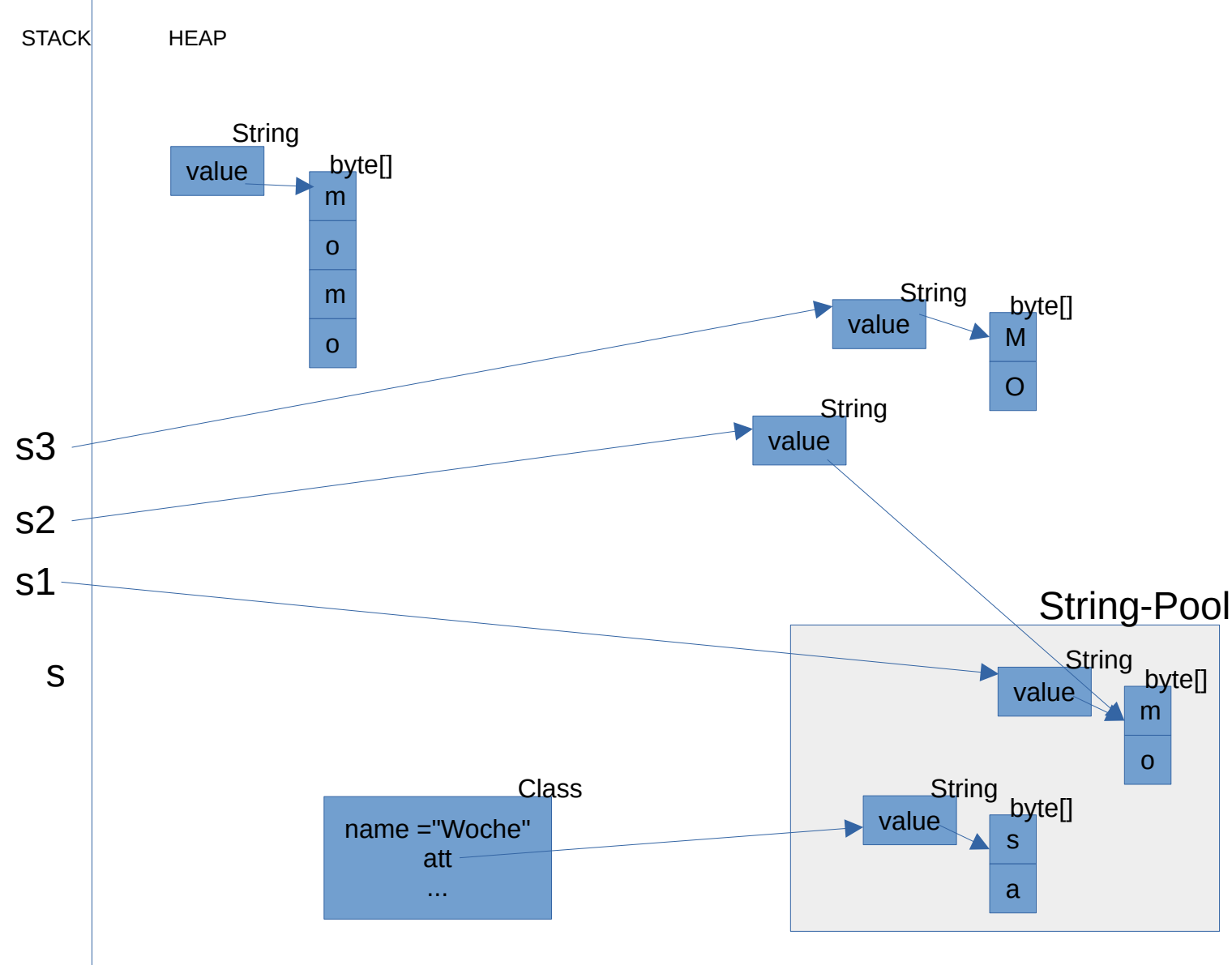

```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```



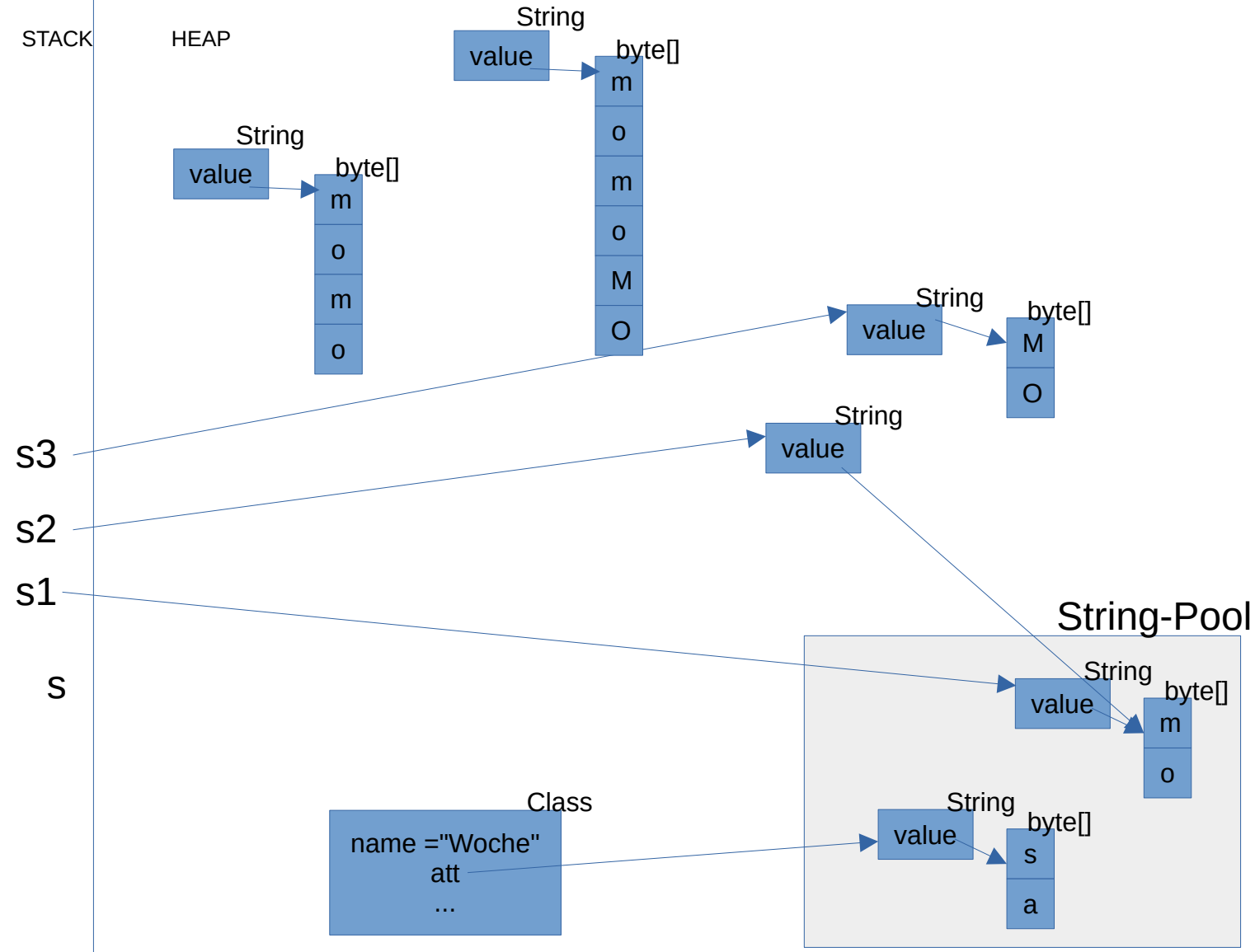

```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```



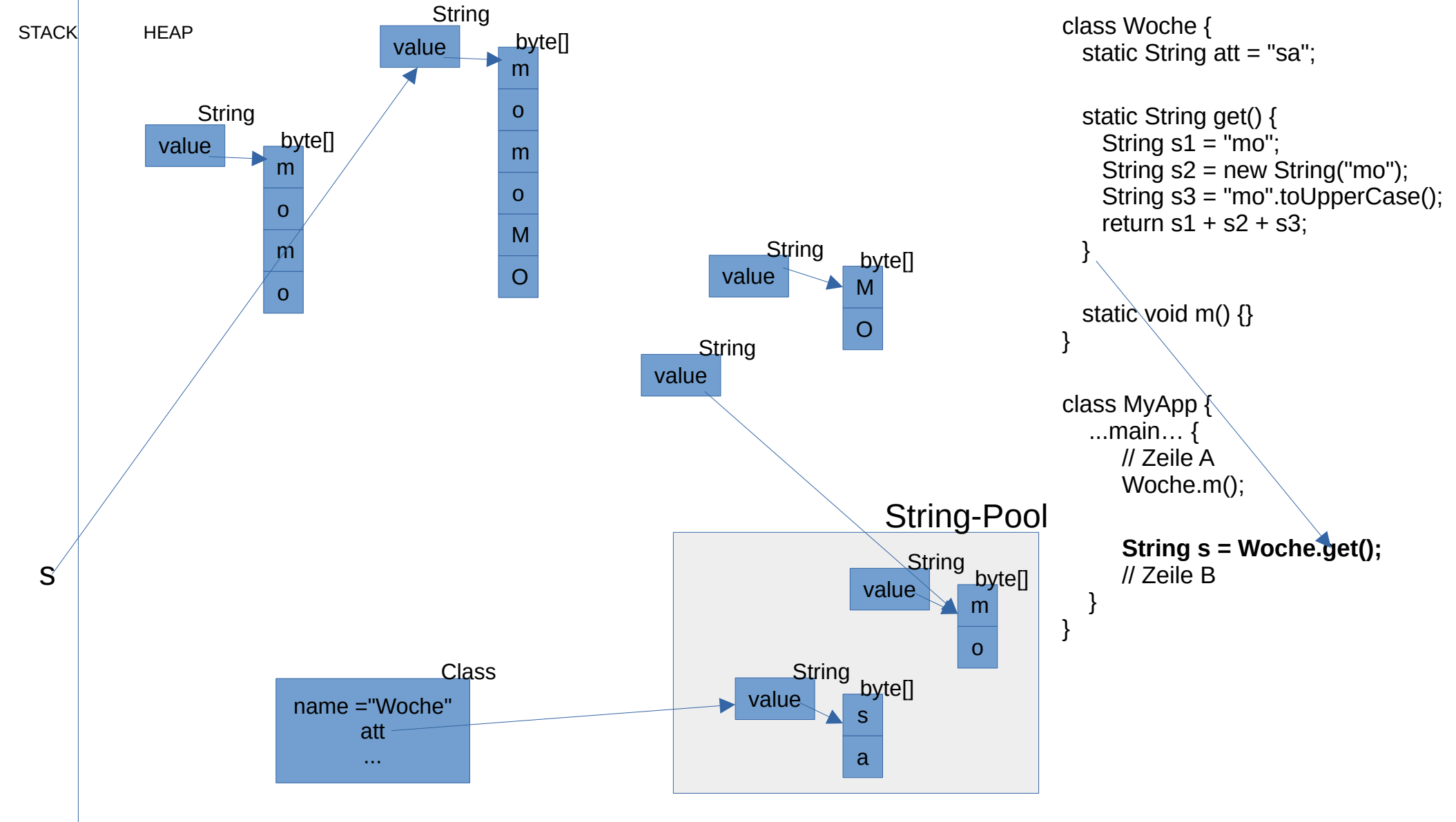

```
class Woche {
   static String att = "sa";


   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase()
      return s1 + s2 + s3;
   }


   static void m() {}
}


class MyApp {
    ...main… {
        // Zeile A
        Woche.m();


        String s = Woche.get();
        // Zeile B
    }
}
```



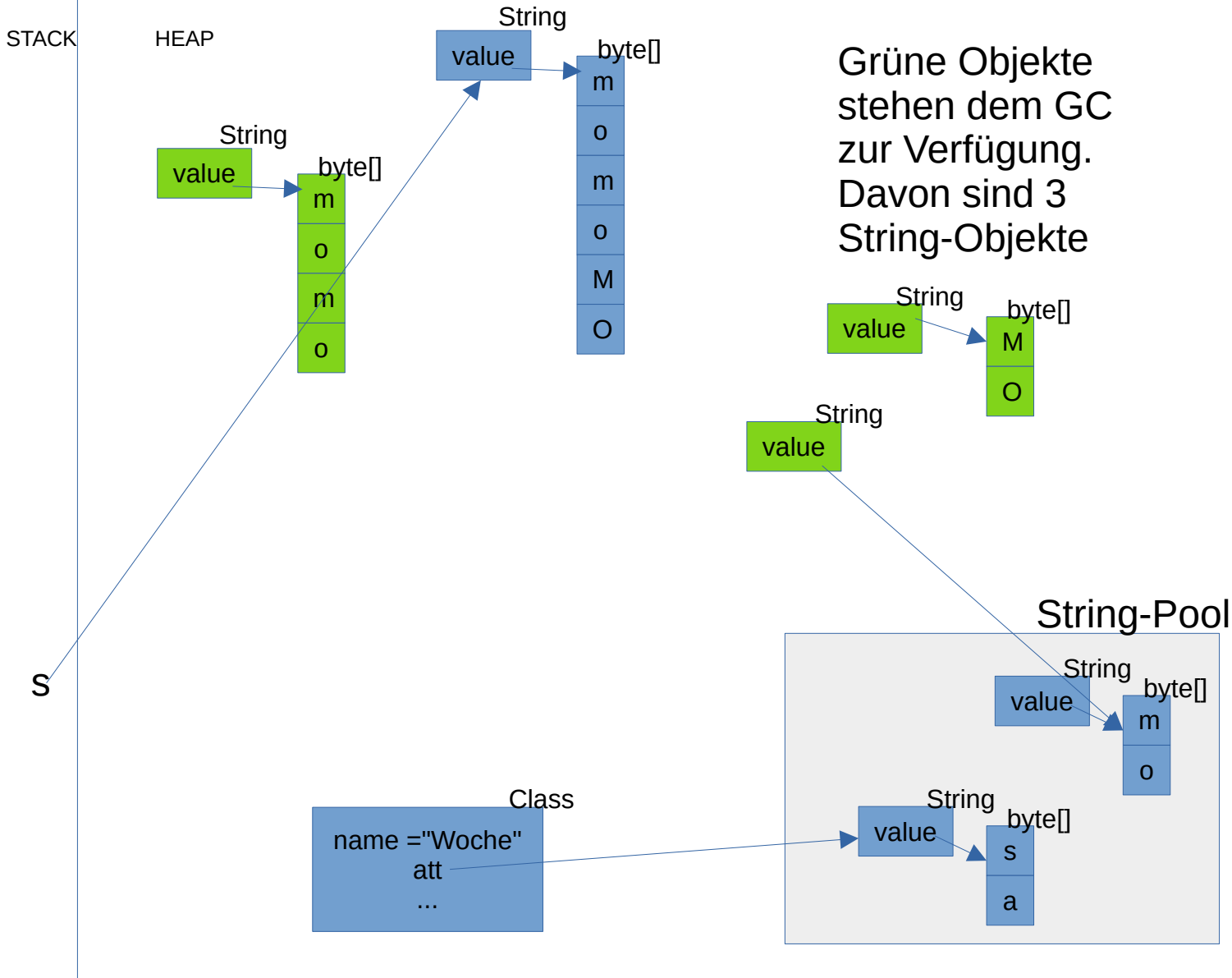

```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```




```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```

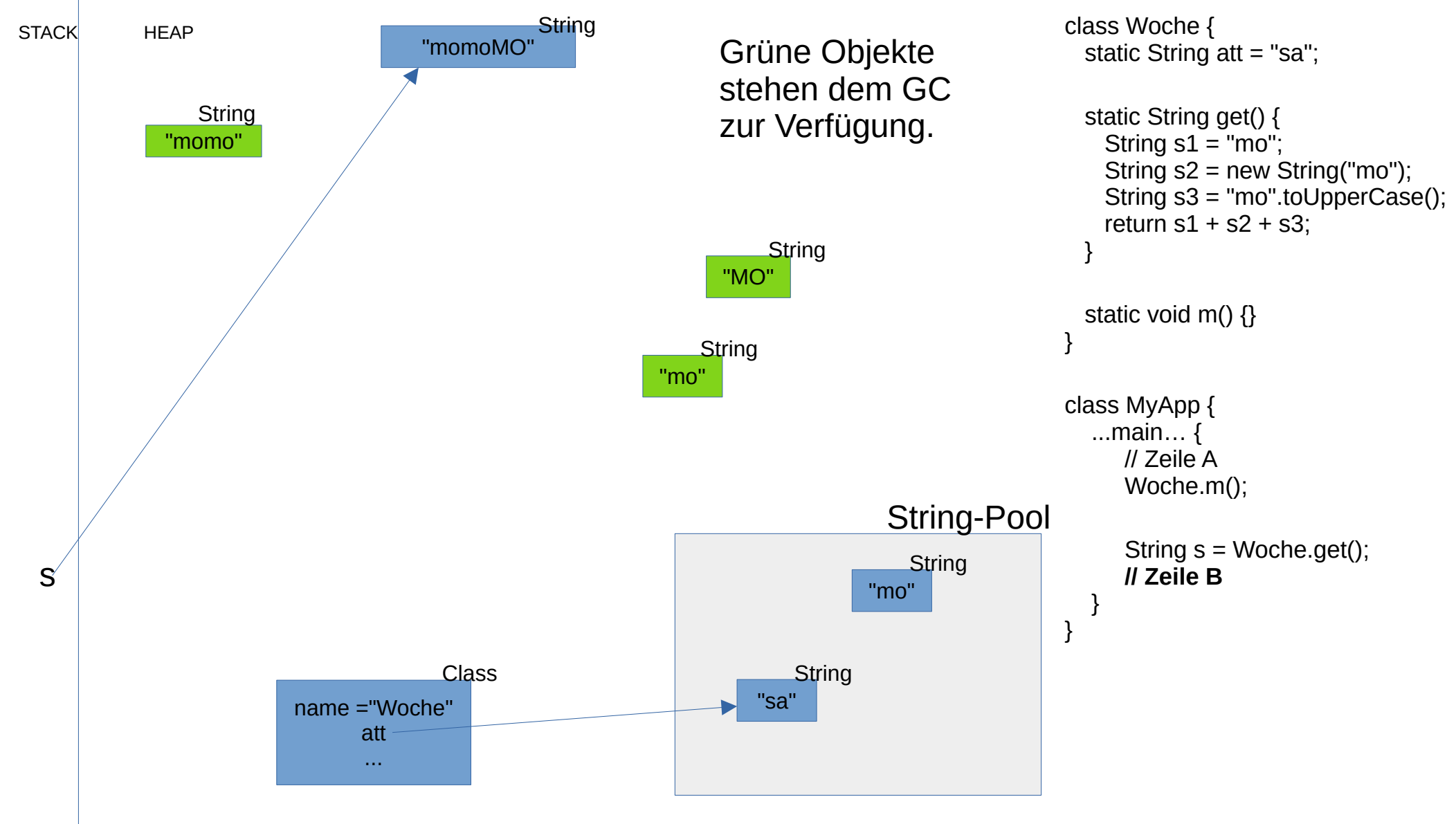



```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```




```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```

Da in der Prüfung die internen byte[] aus Strings nicht gefragt werden, kann man vereinfacht die String-Objekte darstellen:




```
class Woche {
   static String att = "sa";

   static String get() {
      String s1 = "mo";
      String s2 = new String("mo");
      String s3 = "mo".toUpperCase();
      return s1 + s2 + s3;
   }

   static void m() {}
}

class MyApp {
    ...main… {
        // Zeile A
        Woche.m();

        String s = Woche.get();
        // Zeile B
    }
}
```