Exercise 5.1
Optimizing code

SKILLS:
After this exercise you get the ability to find bottlenecks in code and to remove them.

GOALS AND OBJECTIVES:
The goal is to find a better solution to the given ones.

WORK STEPS:
Improve:

```java
// class initialization
public class cls_init1 {
static class Data {
private int month;
private String name;
Data(int i, String s) {
month = i;
name = s;
}
}
Data months[] = {
new Data(1, "January"),
new Data(2, "February"),
new Data(3, "March"),
new Data(4, "April"),
new Data(5, "May"),
new Data(6, "June")
};
public static void main(String args[]) {
final int N = 250000;
cls_init1 x;
Timer t = new Timer();
for (int i = 1; i <= N; i++)
x = new cls_init1();
t.print("I am inefficient");
}
}
```

Tell the difference between using **==** and **String.equals()** to compare strings. Make use of interning strings.

Exercise 5.2
Avoiding memory leaks

SKILLS:
After this exercise you will know how memory leaks occur and to avoid them.

GOALS AND OBJECTIVES:
The goal is to create memory leaks in Java and to find them using jVisualVM.

WORK STEPS:
- Define a pool of objects and try to create a memory leak inside of it.
- Show at least two different ways for achieving the previous goal.
- Find the leaks using jVisualVM and store the memory dump as documentation for this exercise.