Encapsulation

- The jEdit **Buffer class** and how scope can help understand code.

- The **Day class** and how encapsulation hides implementation details likely to change.

- The **GameRecord class** and how a class does not necessarily encapsulate all that much.

Take home message: Small leaks + time = Serious unpleasantness

Global

class Box { public static int x = 0; }

Object
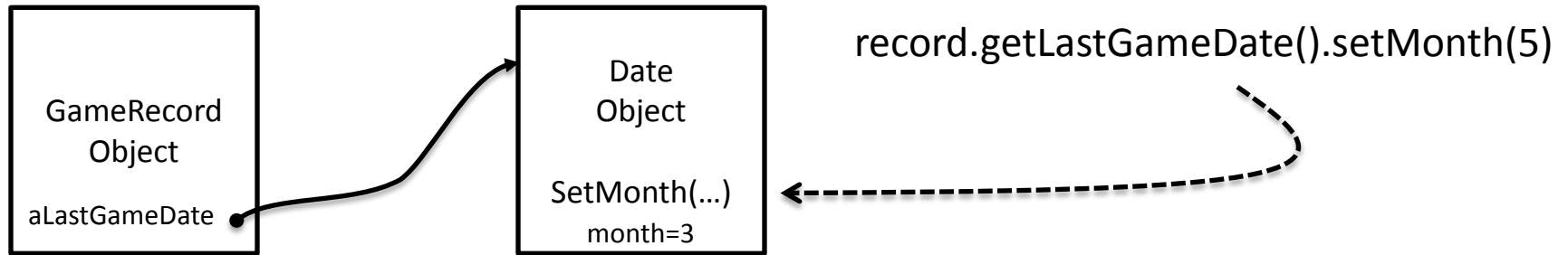
class Box { private int x = 0; }

Local

class Box {
    void doit() { int x = 0; }

None

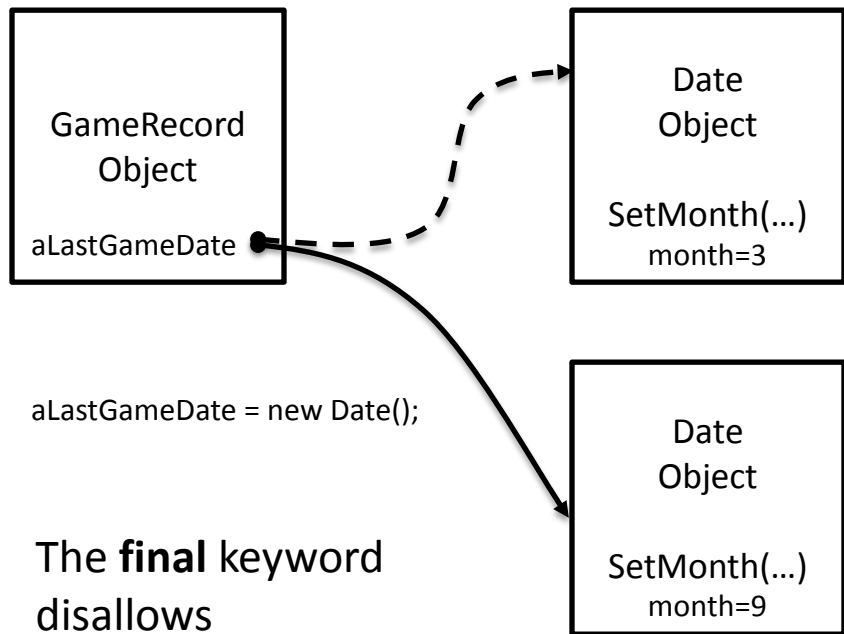doit(computeit())

record.getLastGameDate().setMonth(5)

GameRecord Object

aLastGameDate

Date Object

SetMonth(…)

month=3

But for mutable objects stored in class variables you can change the state of the object despite the final field.

Reassignment

GameRecord Object

aLastGameDate

Date Object

SetMonth(…)

month=3

aLastGameDate = new Date();

Date Object

SetMonth(…)

month=9

The **final** keyword disallows reassignments

Object.clone() (de)mystified