

iFitLab: Low entrance-barrier fitting tool

- Web-based
 - written in js with d3.js
 - django/Python backend
- Syntax-free, generic UI system
- iFit backend:
 - excellent data loading
 - math and algorithms

Simple variables and values

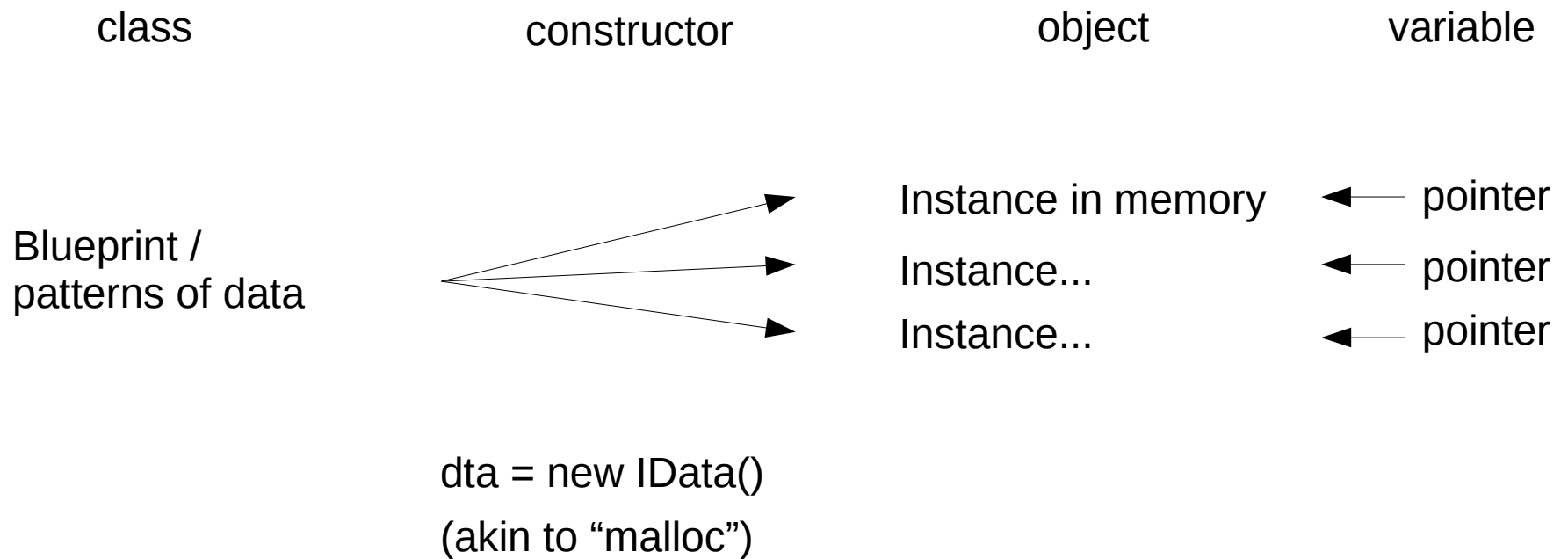
```
String s = "green"
```

```
int cnt = 7
```

```
bool isTrue = true
```

A typed pointer to a value on the “stack”
(immediately accessible, but limited)

Classes, constructors and objects



These live on the “heap” - the program’s non-stack memory
(potentially huge)

Class methods, and functions

`<Class>.<method>:`

A function attached to an object (class instance), which may change the data/state of that object.

e.g. `ldata.rmint(int, int) → void` (no return value)

`<function>:`

An operator that does not change the state of input objects, but returns a new one.

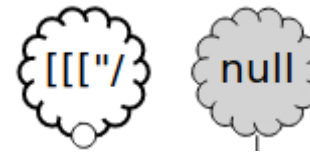
e.g. `fits(IData, IFunc) → IFunc`

Symbols

Void pointer:



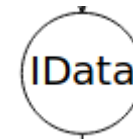
Simple variable:



Typed pointer/object reference:



Object constructor:



Class method:



Function:



