Introducing Objects and Object Oriented Programming

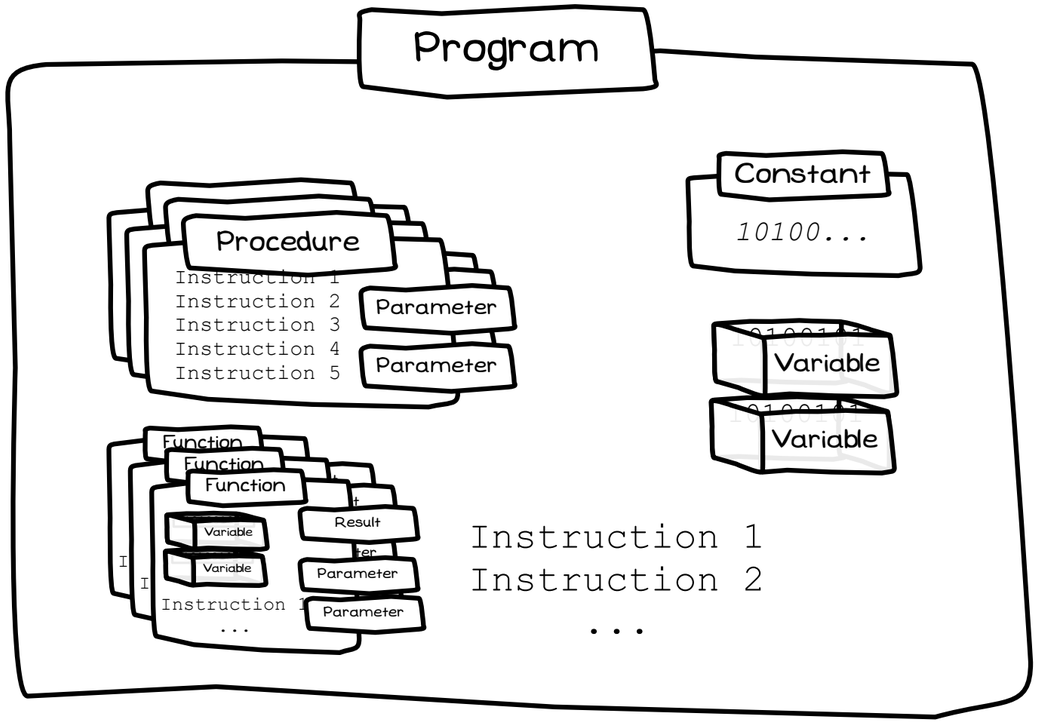
Dr Chris McCarthy

SWINBURNE UNIVERSITY OF TECHNOLOGY

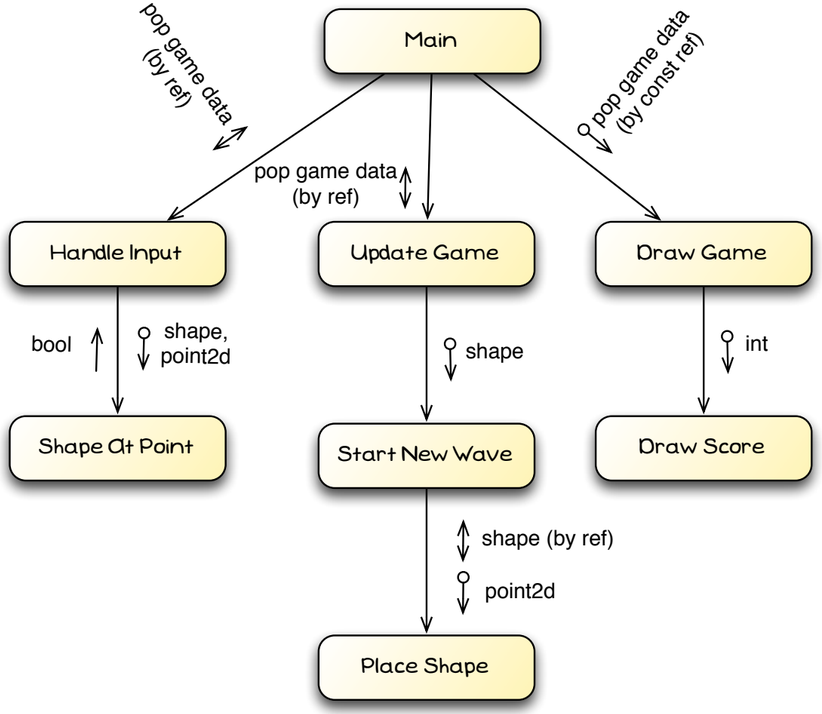
Software Development is about defining instructions for computers

Code

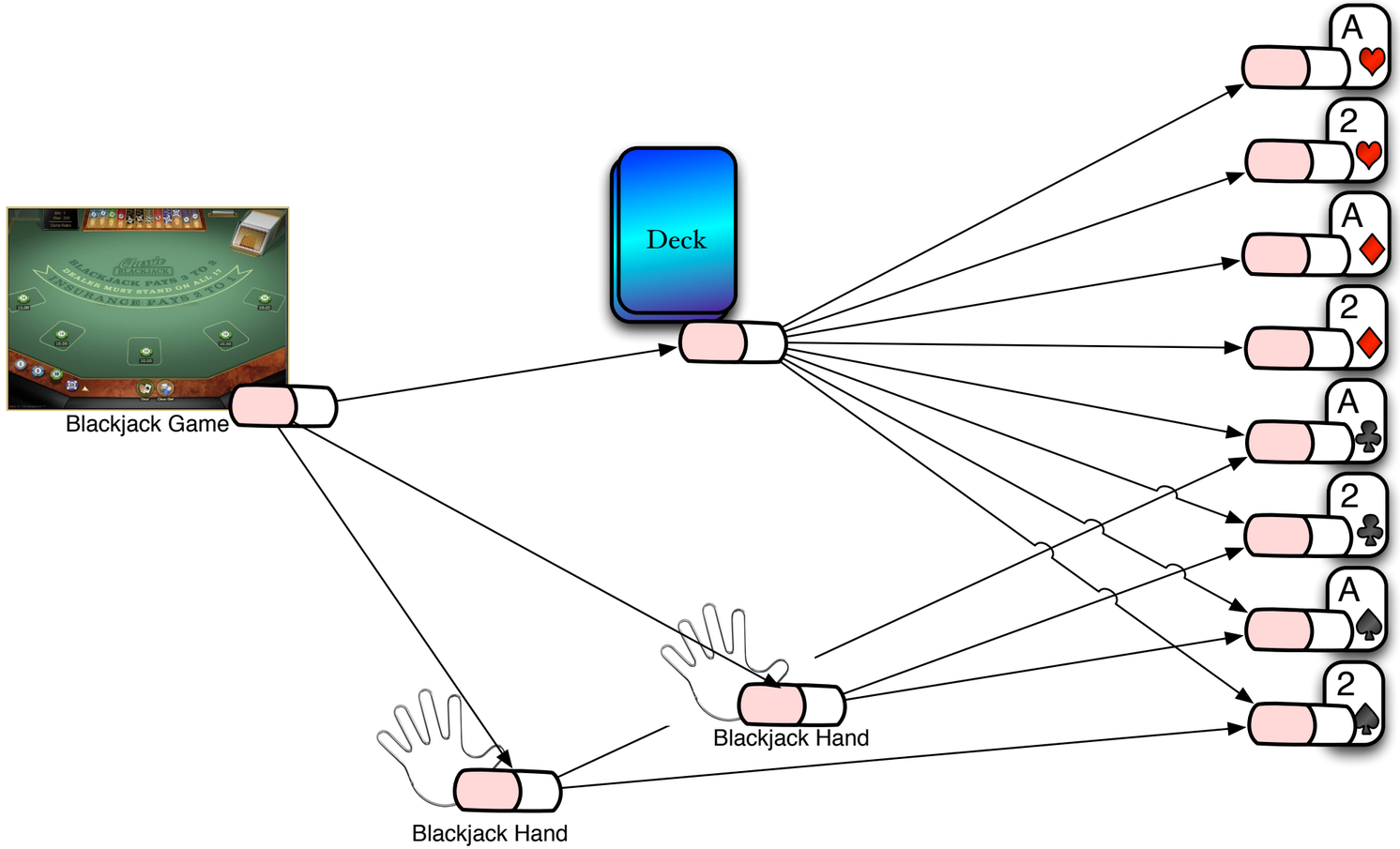
Developers create programs using a range of artefacts to manage complexity



Procedural programming uses functional decomposition, but has limits as size grows



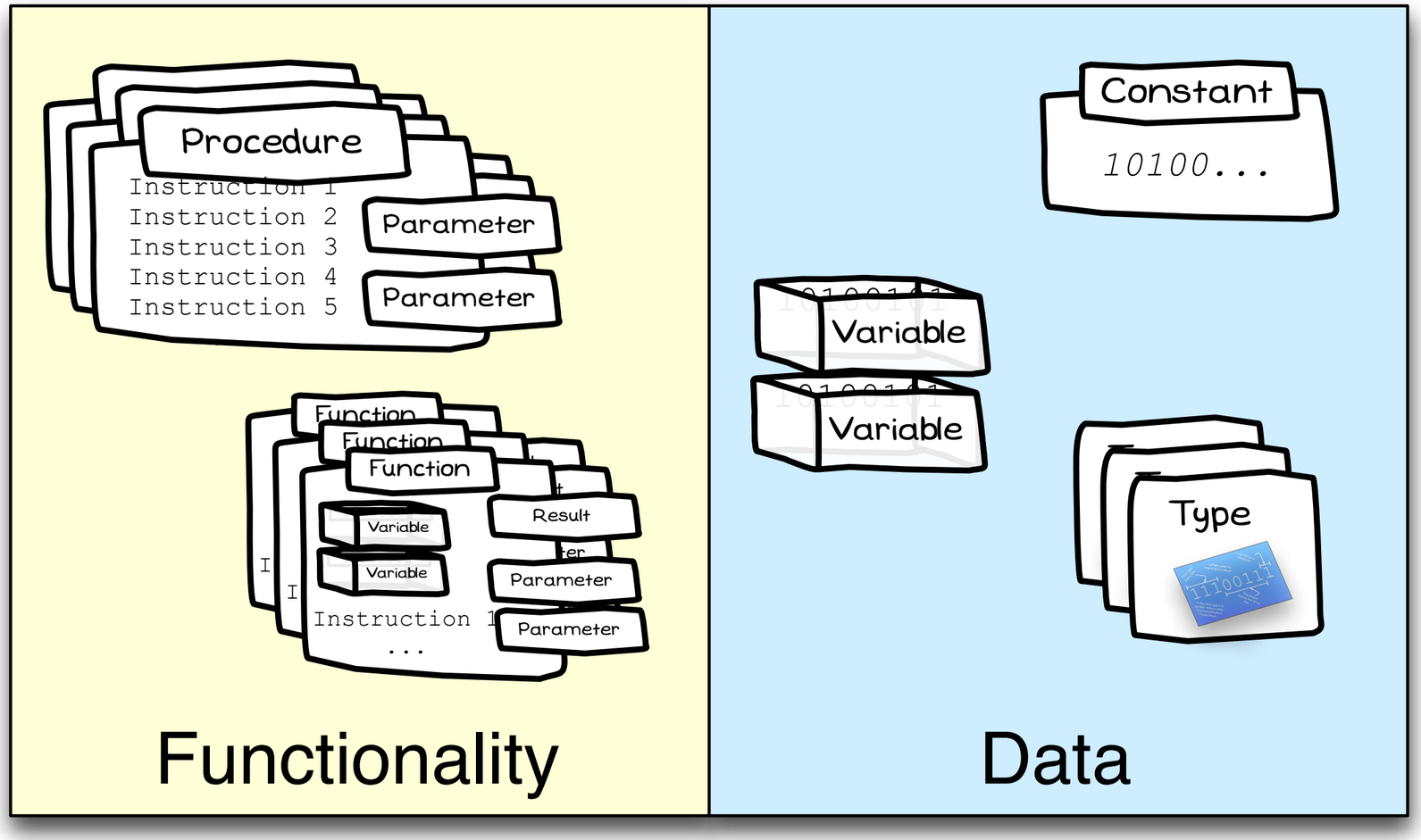
Object oriented programming offers means of managing complexity for larger software



Change your approach to software design to master OO programming

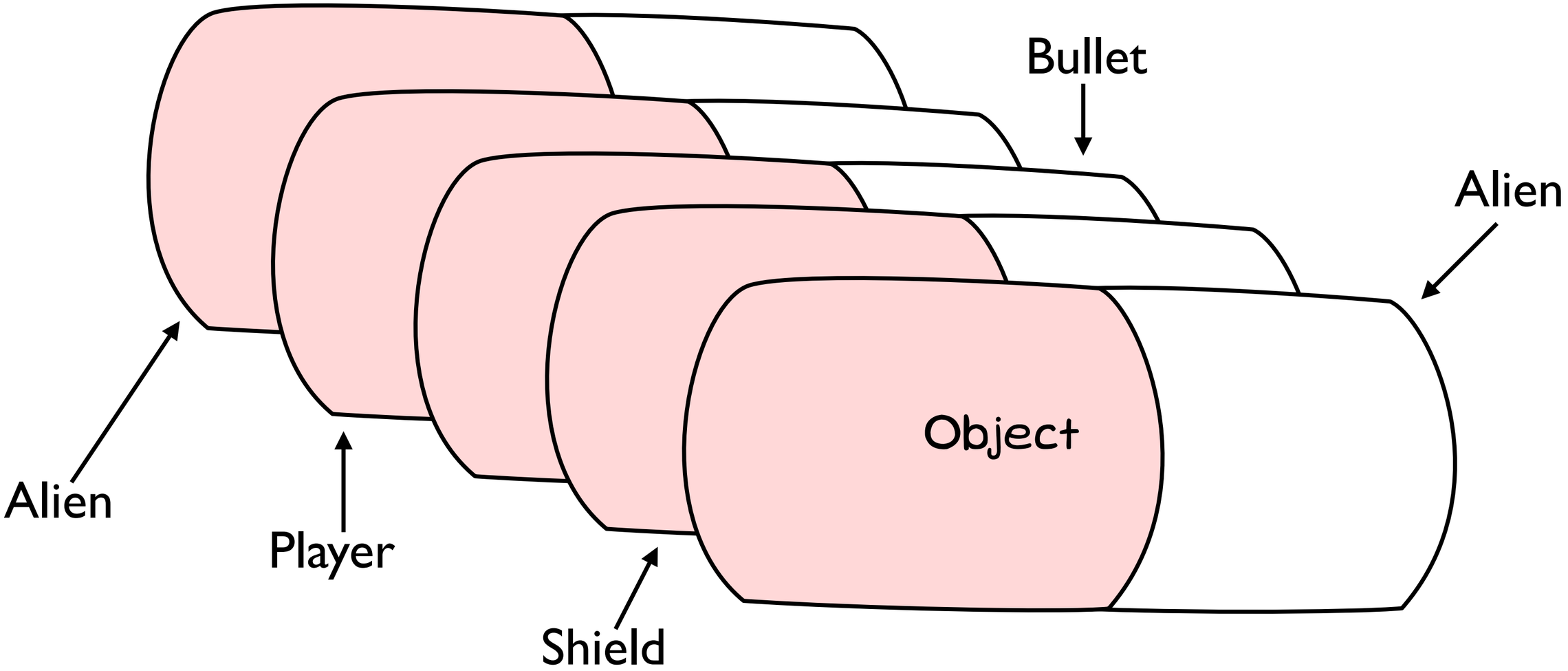
See software as involving collaborating entities (objects) that know and do things

Program procedurally by organising code into separate artefacts for data and functionality



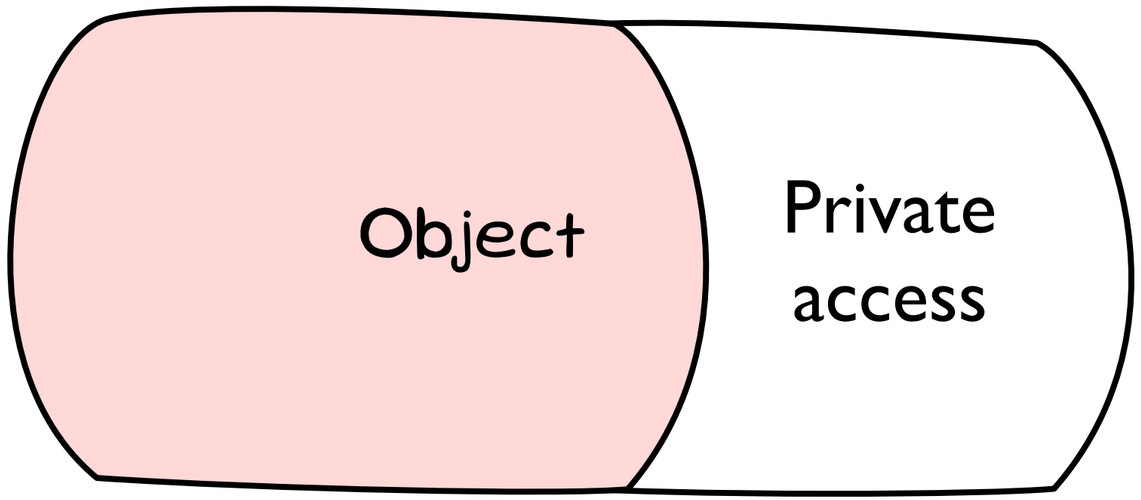
With Objects, you create entities that encapsulate **both** functionality and data — they know and can do things

Build programs from many interacting objects, each playing a role in the overall solution



Picture each object as a capsule with an "inside" and "outside" — not everything is accessible

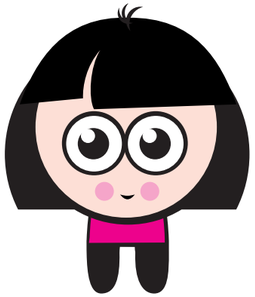
Public access



Things the object knows and can do can be hidden within the object.

Design programs by breaking problems down into objects

Use abstraction to classify the different kinds of roles objects will play in your software



classify

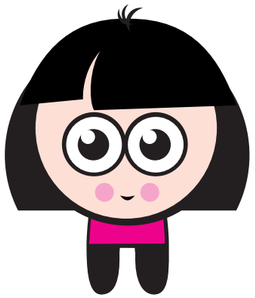
Player

classify

Alien

Use Abstraction (Classification) to define object classes

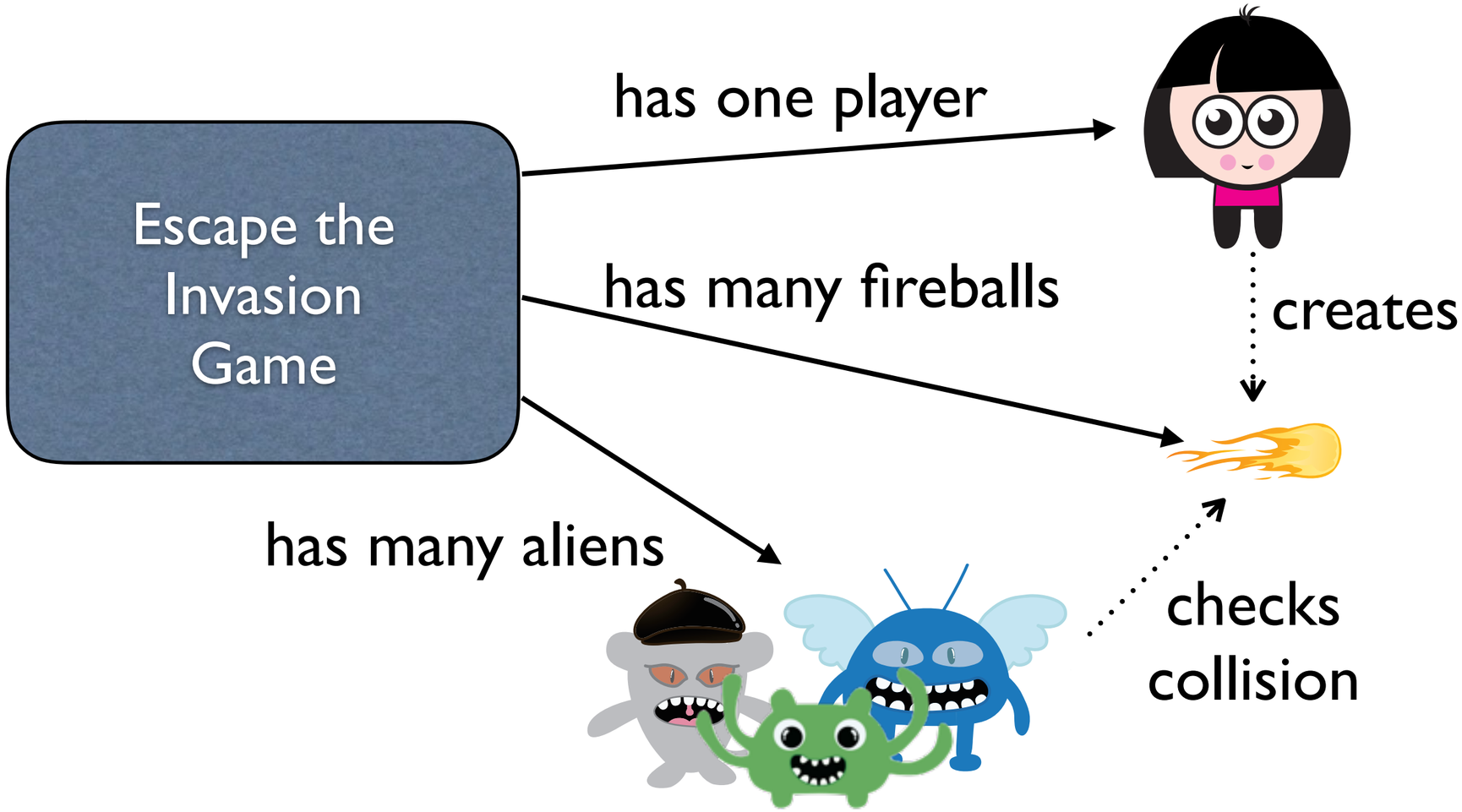
Record what each object for a role will know and be able to do



classify

Player
Knows its location Knows its health Knows it heading
Can move Can be hit by Aliens Can fire bullets …

Indicate other roles that the objects will need to collaborate with to achieve its goals



Activity - Design solutions to scenario 1

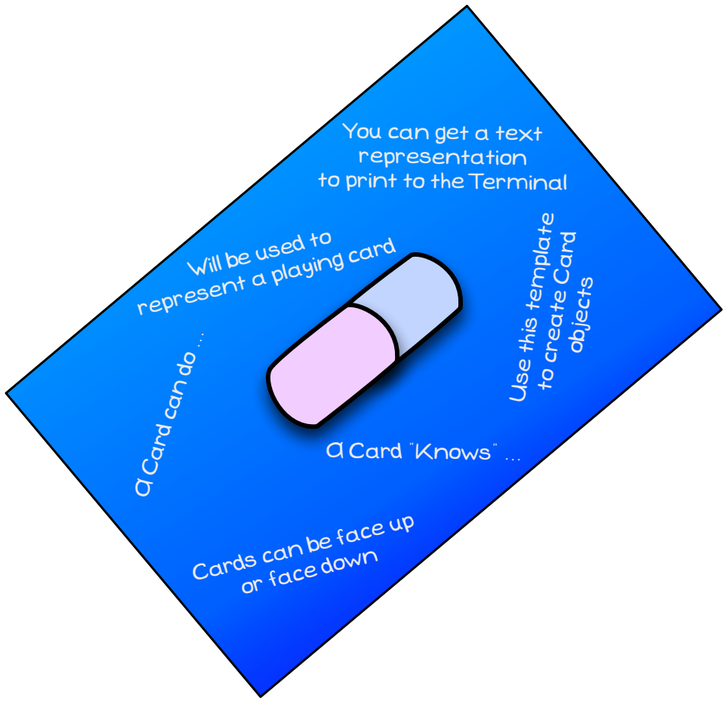
Roles, Responsibilities (know and do), Collaborations

Activity - Design solutions to other scenarios

Roles, Responsibilities (know and do), Collaborations

Implement your designs using an object oriented programming language

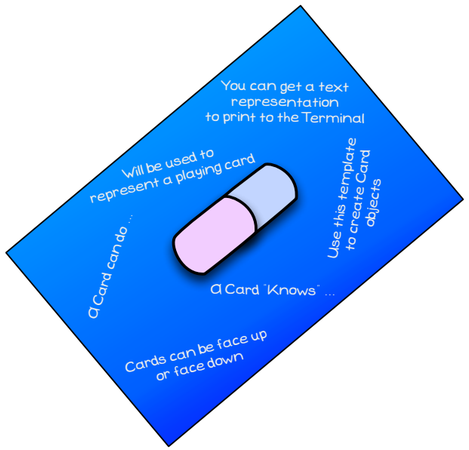
Define classes to capture object specifications

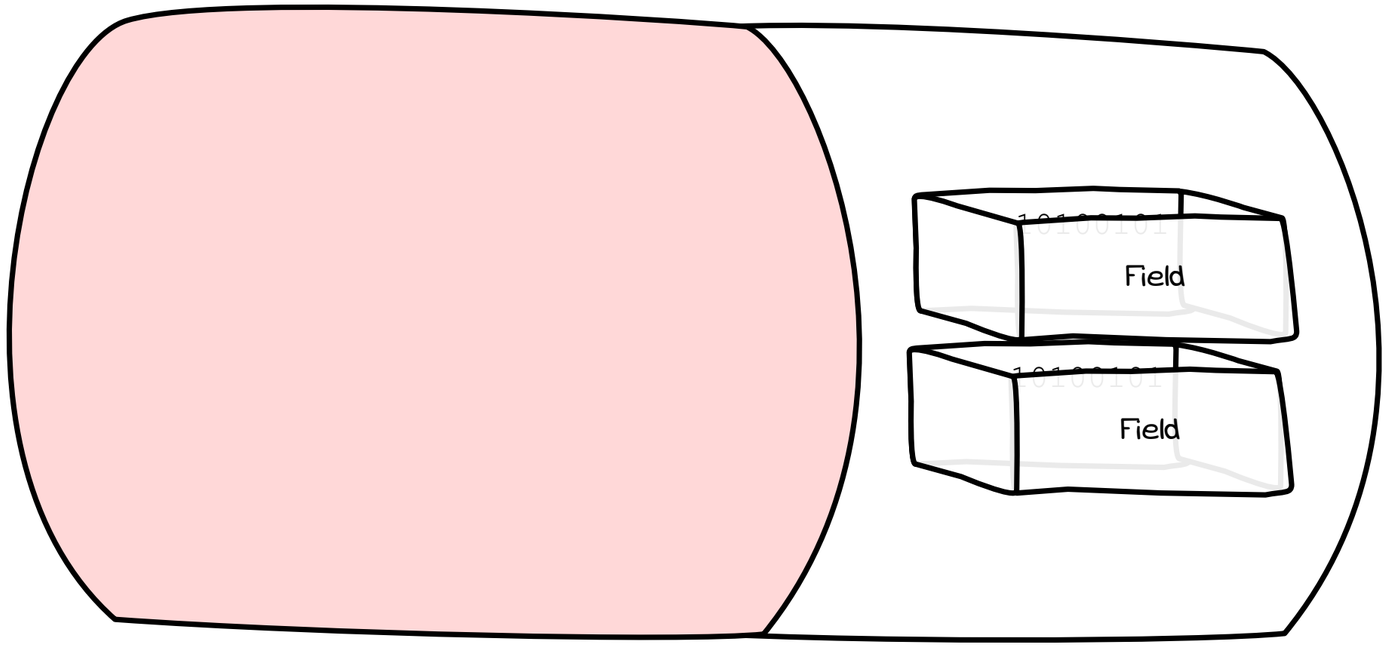


captured in

class

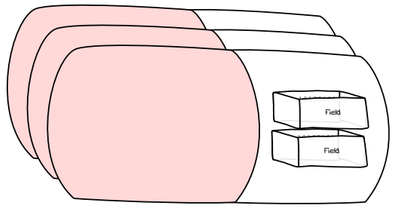
Add private fields to the class to store the things the object knows



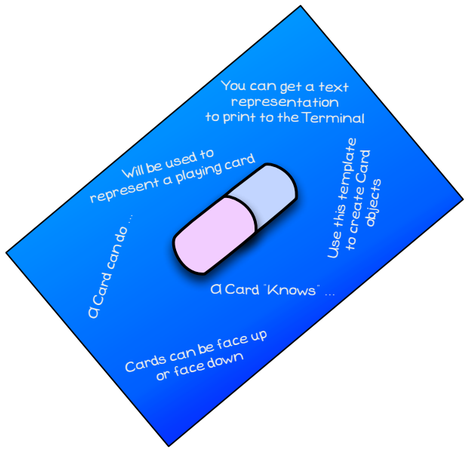


Field are declared within the class

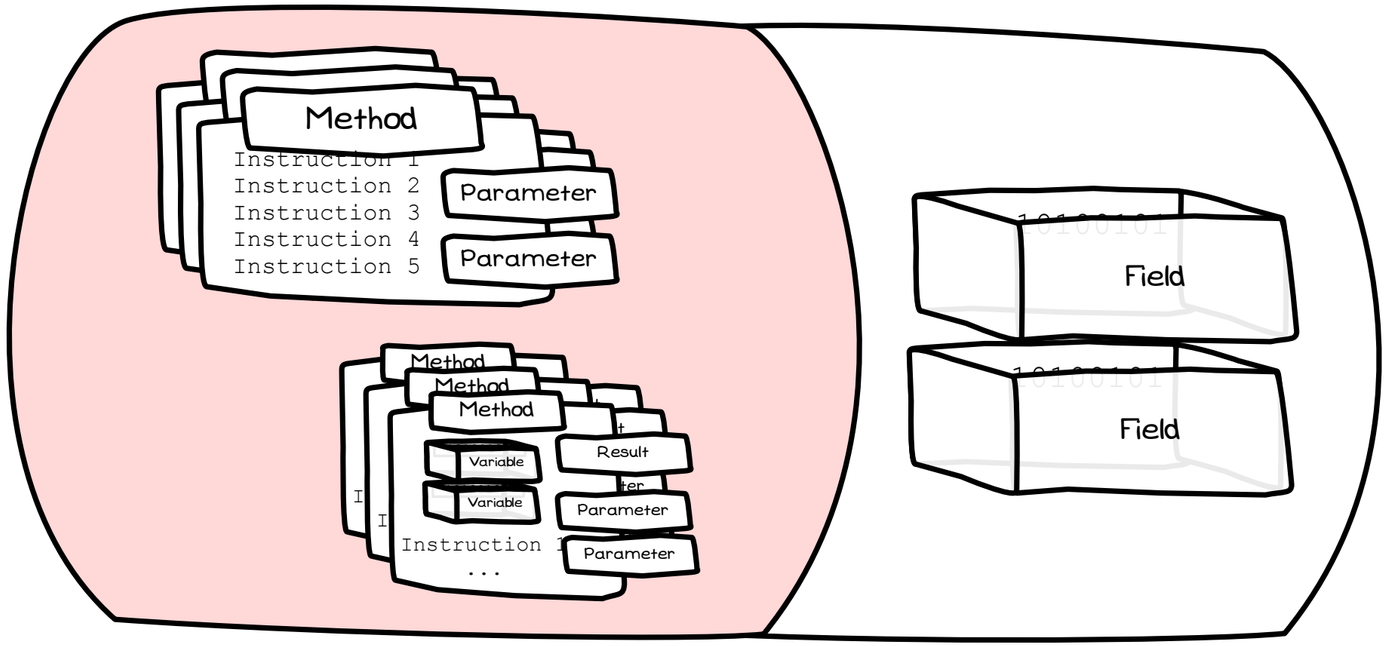
The field exists within each object



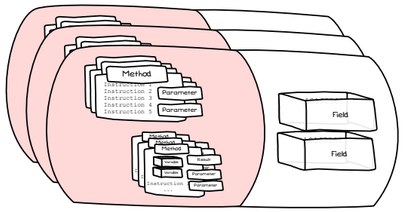
Add methods to the class to code the things the object can do



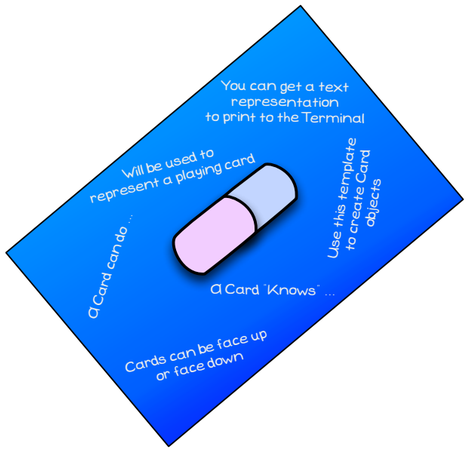
Methods are declared within the class



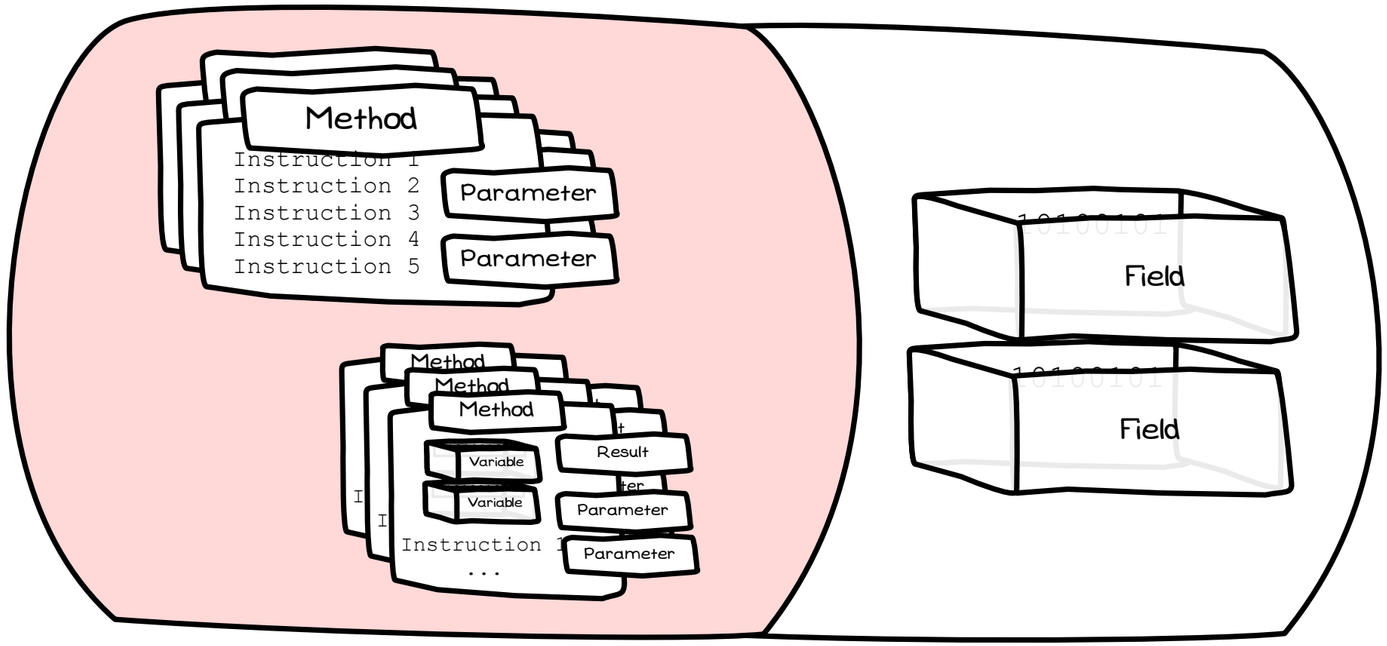
The methods exist for each object



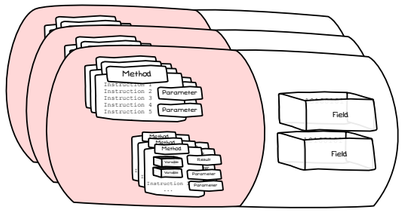
Add properties to the class to code give access to hidden data



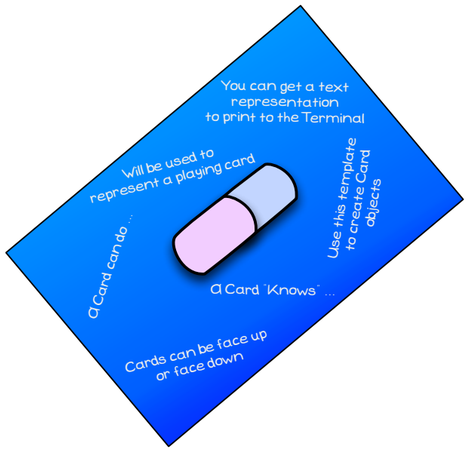
Properties are get and set methods declared within the class



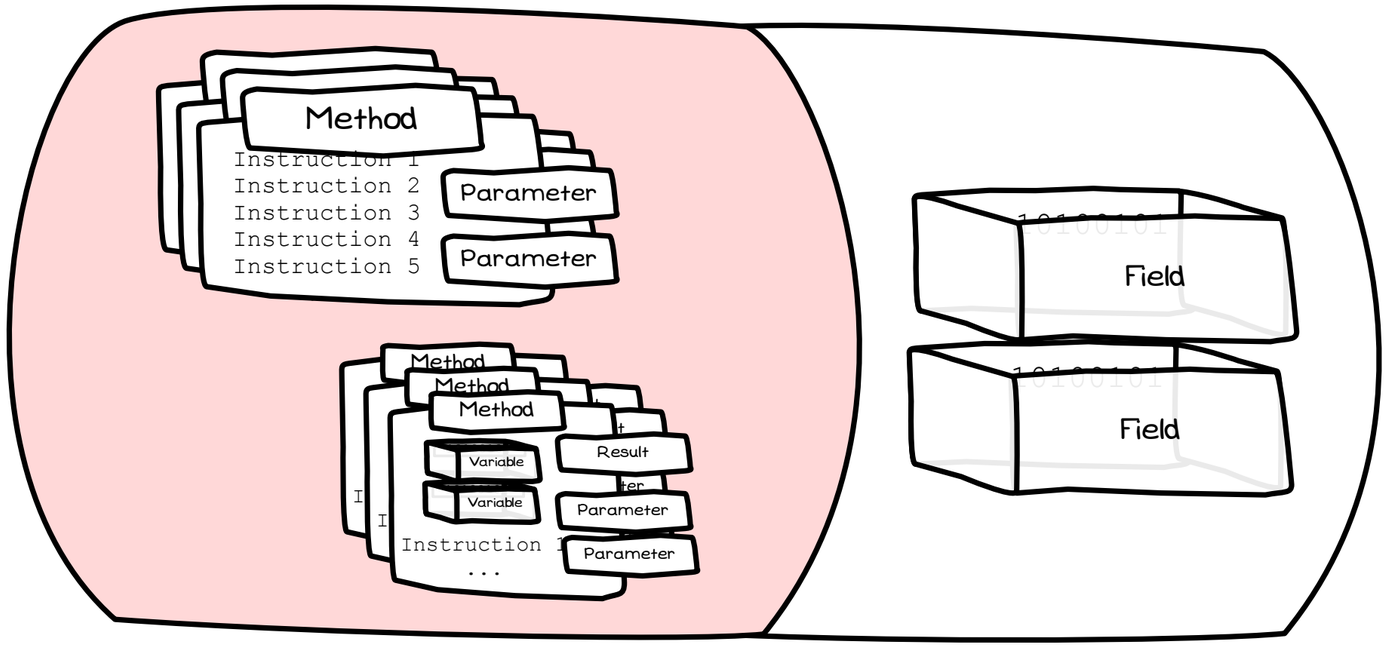
The properties exist for each object



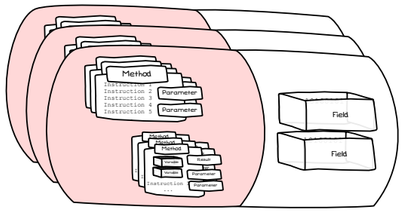
Add special methods called constructors to initialise your objects when created



Constructors are declared within the class

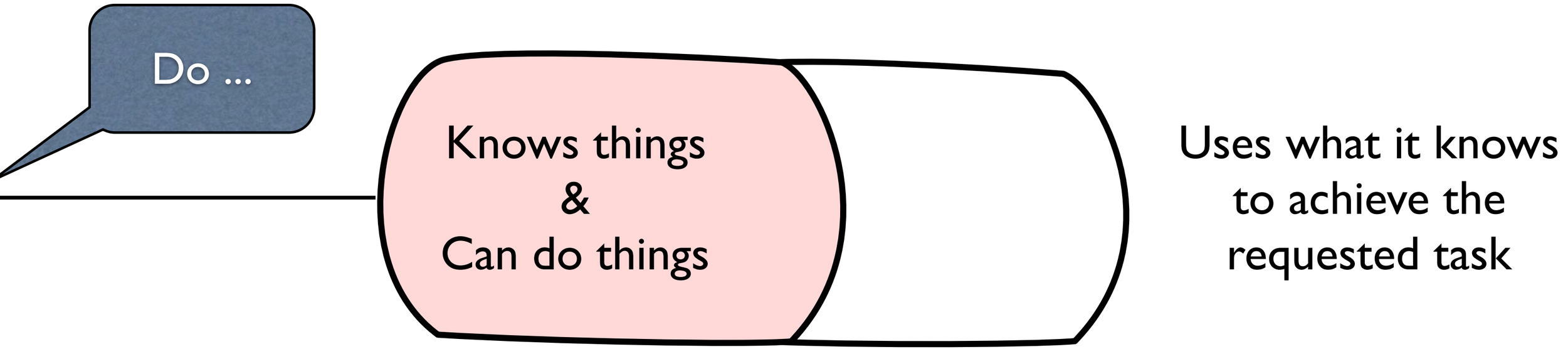


These define how to create/initialise the objects.

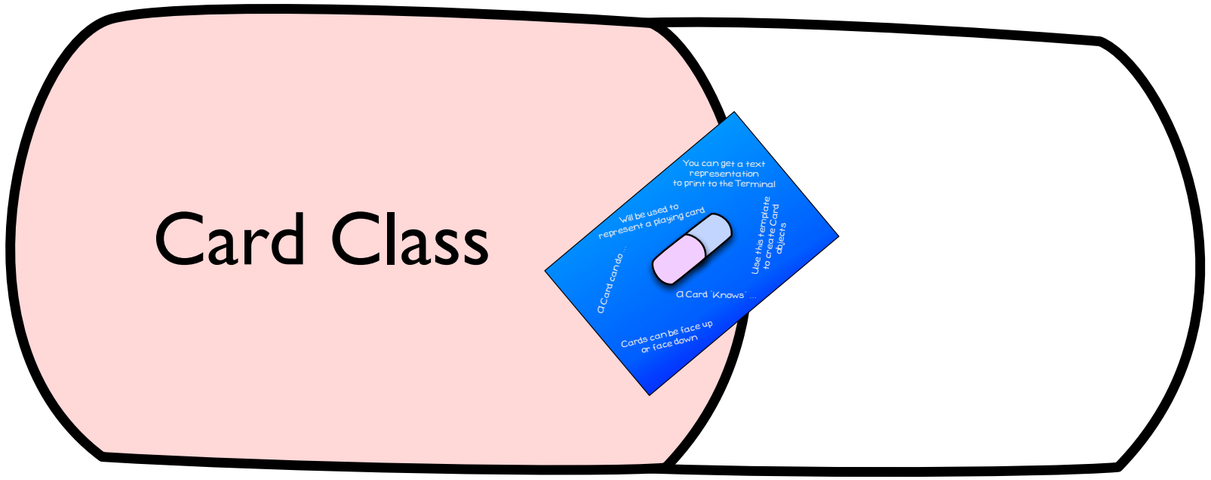


Demo - Card class

In your program create objects, and get them to do things...

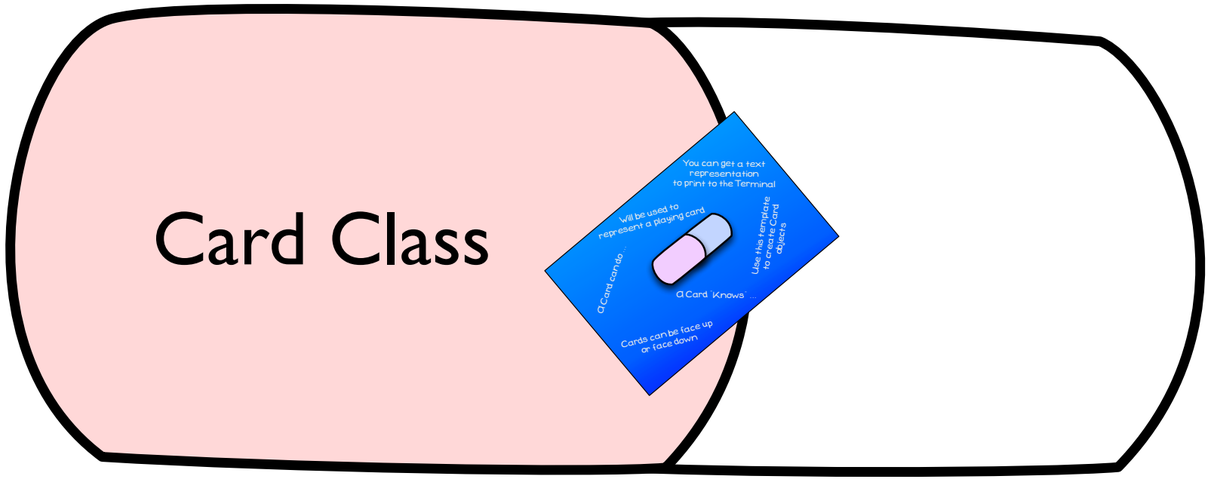


At runtime classes also become objects that provide services



Can create objects...

You can add features to class objects in your code



Can create objects...
You can add more...

Activity - Main

Will you be able to create object oriented programs?

Object orientation offers new means of managing complexity

Change your approach to software design to master OO programming

Objects combine data and functionality, creating larger more meaningful abstractions

Get started creating object oriented programs