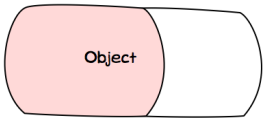
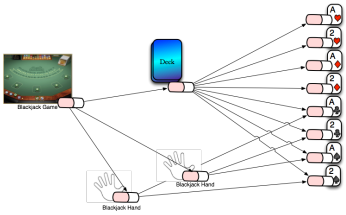
Object Collaborations

by Andrew Cain and Willem van Straten

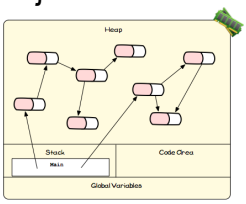
Object oriented programs are designed around the idea of objects



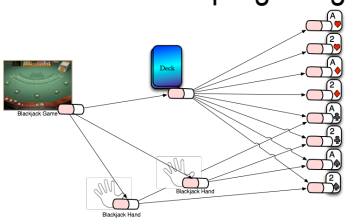
Developers use objects to create components for use in their software



Object oriented programs usually contain many objects of different kinds



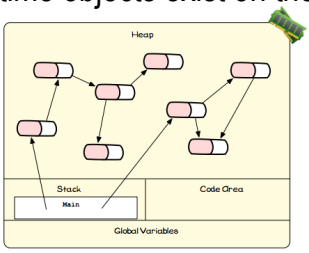
To work effectively objects will need to interact to achieve program goals



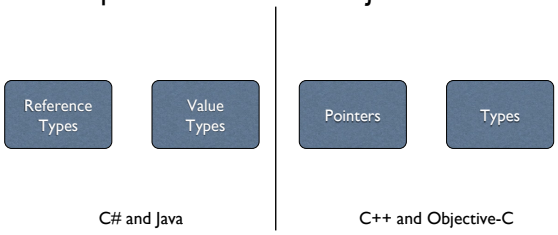
Use object relationships to enhance the power of your object oriented programs

See how references enable the creation of flexible networks of objects

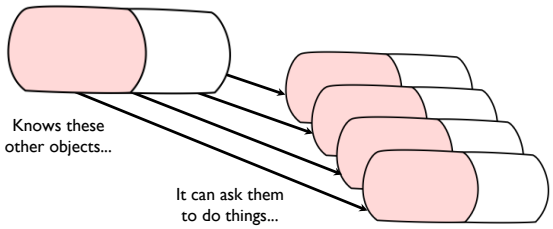
At runtime objects exist on the heap



Languages use some form of pointer to refer to objects...



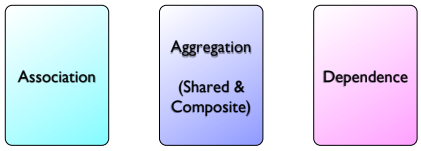
Pointers allow flexible relationships between objects



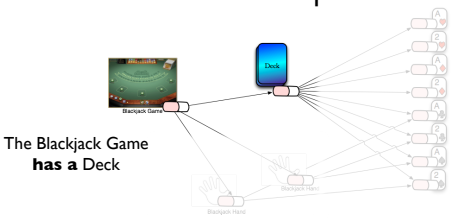
Activity - Design a Deck class

Design interactions between the objects in your solution

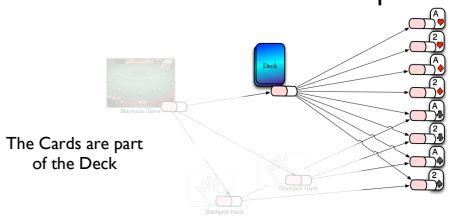
Remember there are three main kinds of relationships



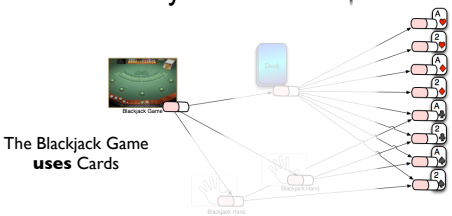
Use association for "has-a" relationships



Use aggregation for whole-part or container relationships



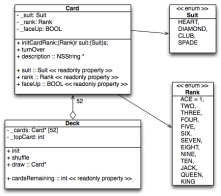
Use dependency for temporary "uses" style relationships



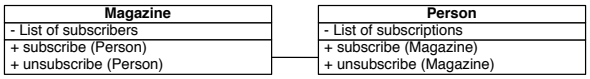
Activity - Design Battleships

Communicate your ideas using the Unified Modelling Language

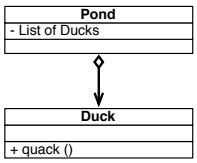
Communicate the static structure of your program using a Class diagram



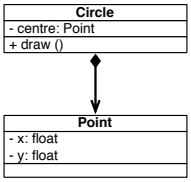
Association



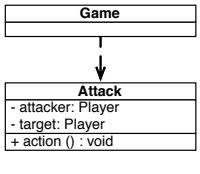
Shared Aggregation



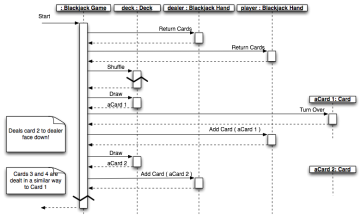
Composite Aggregation



Dependence



Communicate interactions using a Sequence Diagram



Activity: How can you communicate...

Will you be able to setup and use object interactions in your programs?

To achieve great results object oriented programs need many objects

Use object relationships to enhance the power of your object oriented programs

Achieve your program goals through object communication

Start your objects communicating!

This Week's Tasks

Pass Task 9 - Shape Drawer
Pass Task 10 - The Spell Book