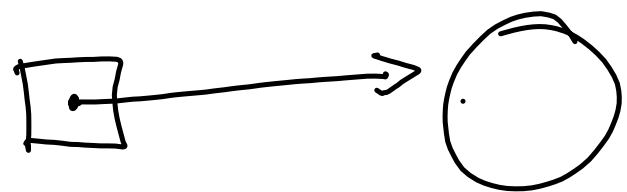
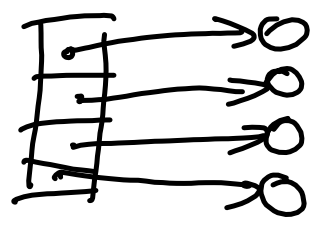


Object references P



Many references



TODAY: Higher level Modelling  
and description

Object Oriented Design

From a problem description

FIND:

- classes
- methods } Interface
- variables } Object state

## Problem Example Bank Account

A bank in Oslø has a number of customers that hold one or more Bank Accounts. Customer can deposit and withdraw money to/from their accounts & check the balance on accounts

### Class - Find NOUN

one card per class

### Responsibilities - Find VERBS

- knowing something
- Doing something

Bank ✓  
 Customer ✓  
 money? ✓  
 currency ✓  
 Oslo ✓  
 number ✓  
 Account ✓  
 balance ✓

Responsibility Driven Design

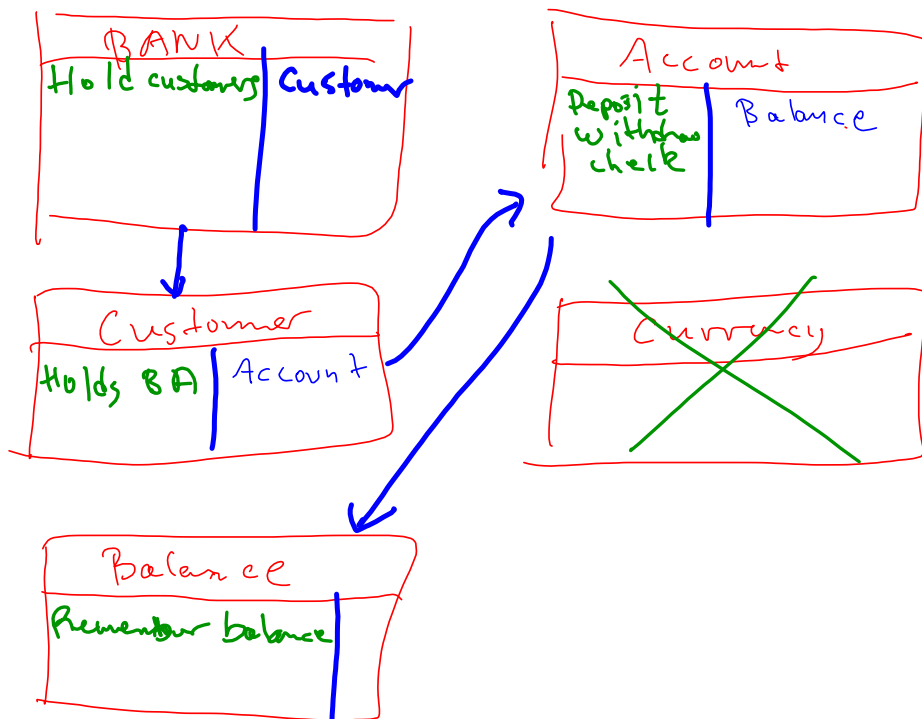
Separating Responsibility

CR C Cards

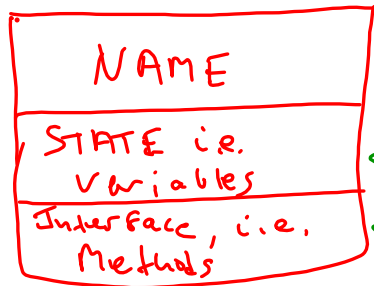
Class ✓

Responsibility ✓

Collaborators



# UML - Unified Modelling Language CLASS DIAGRAM



CRC CARD

Class

Collaborators  
object references

Responsibilities  
· methods  
+ variables

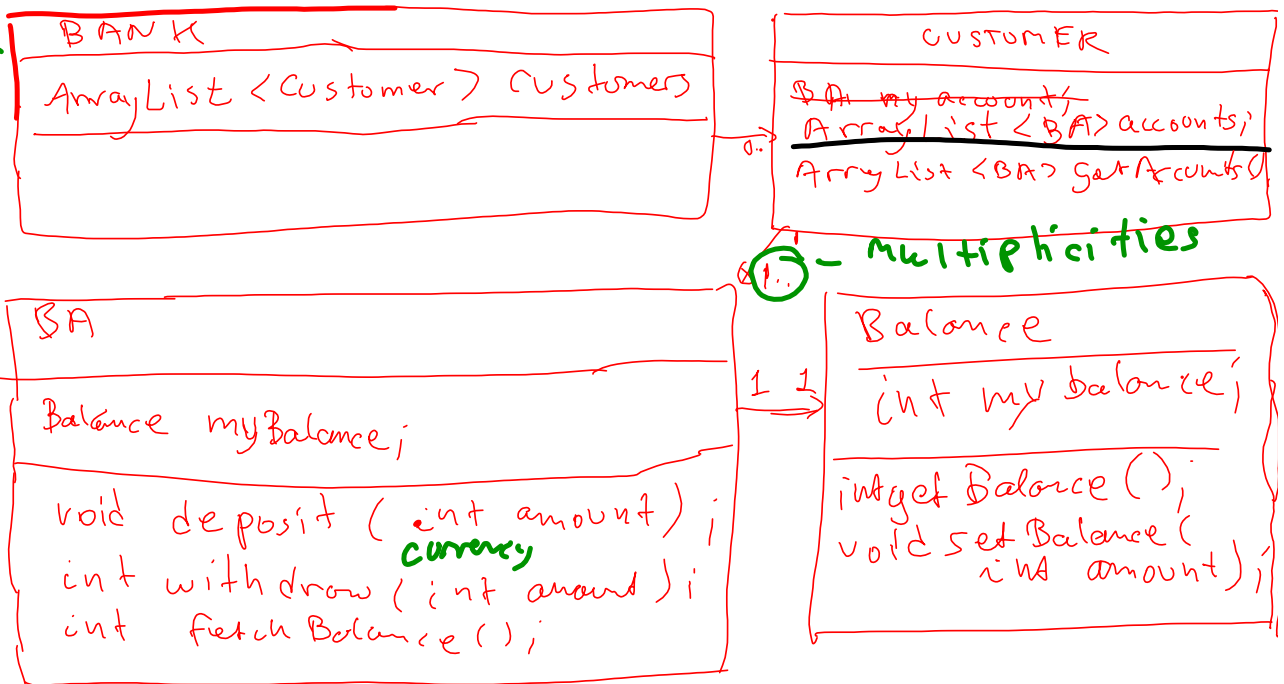
Problem Description

↓  
CRC cards

↓  
UML Class Diagram

↓  
write Java Classes

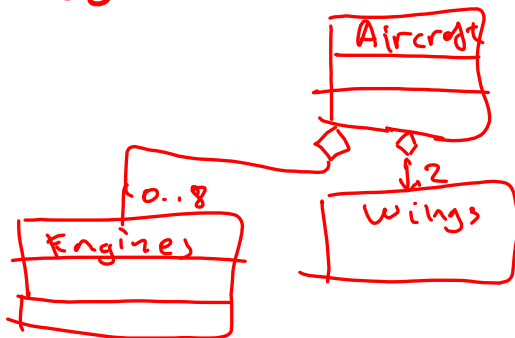
UML CLASS DIAGRAM



more on UML.

Association - Dependency - Collaborators

Aggregation - "strong" association



"has-a" relationship  
 Implemented by  
 an instance variable  
 often ArrayList  
 (if more than one)



Not more UML

~~Inheritance ; Is-a relation~~

~~Activity Diagram~~

⋮

## Enhanced Loops

```
double [] v = ...
double total = 0;
for ( double element : v ) {
    total += element;
}
```

Note:

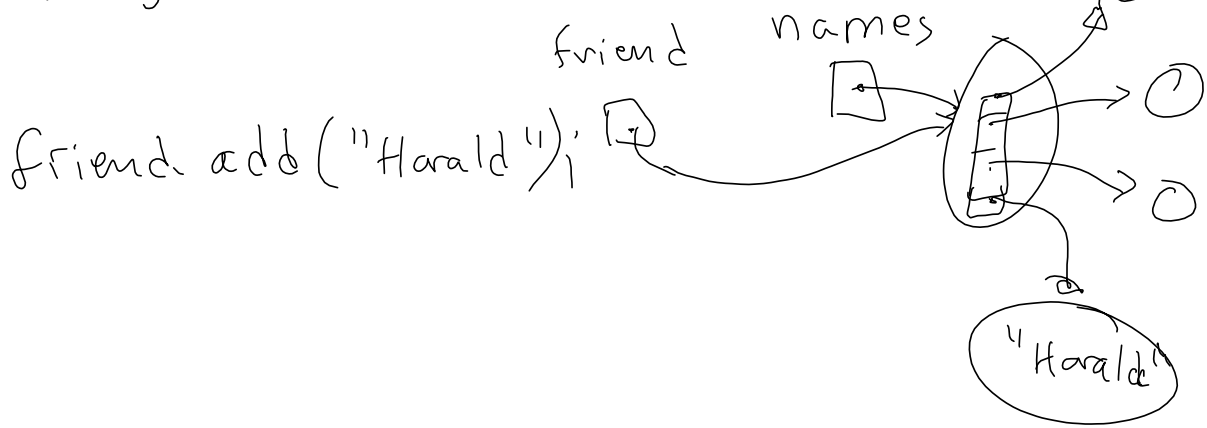
- Read only
- Index not needed

Something with  
Array List

Nice  
Short hand  
ITERATOR

## Note on Copying

ArrayList <String> names = ... ;  
ArrayList <String> friends = names



Choosing between  
Array

ArrayList ?

- Fixed
- efficiency

}

use Array

OTHERWISE

ArrayList