

# Object oriented concepts

# Scope

- Classes and objects
- Information hiding
- Encapsulation
- Access modifiers

# Evolution

- 1960's – large software were complex to handle
- 1970 – procedural programming became popular
- 1990 – object oriented programming became widely used in 1990s

# Procedural Programming

The focus is

- on the design of the processing
- ways of passing arguments
- ways of distinguishing different kinds of arguments
- For example, Fortron, Pascal

# What is the problem?

- Natural way of structuring the world
  - Objects
  - actions



# Objects

- World around us is object oriented
- Objects
  - Animate (people)
  - Inanimate (table)
- Objects are characterized by **attributes** (name, height)
- Objects exhibit **behavior** (walk, talk)

# What is object oriented programming?

- A technique for programming
- Programming language that provides mechanisms that support the object oriented style of programming – for example C++

# User defined types

- C++ provides platform to create user defined types called classes (eg structures in C)
- The variables of these types are called the objects For example, for built in types like int, variables are declared
- Built-in types provide the building blocks for constructing new user-defined types (classes)



# Encapsulation

- Attributes (data) and operations (functions) are encapsulated in objects
- Objects have the property of information hiding

# Encapsulation

- The data components of a class are called data members. The function components of a class are called member functions
- It also means you should be able to use the object without knowing the details of how it is structured internally

# Abstraction

- The idea of abstraction is that a class represents an “abstract” version of an object
- A customer class presents the abstract idea of what a customer is
- A sale class represents an abstract idea of what a sale is

# A Simple OO Model

Objects:



**Joe**

Attributes:

Name  
Address  
Age  
Driver's license  
Number

Relationships:



owns



**Porsche**

Make  
Model  
ID number  
Body type  
Color



# Classes

- A class is a collection of similar objects; a class often defined as:
  - template
  - generalized description
  - pattern
  - “blueprint” ... describing a collection of similar items
- A class identifies ***properties (attributes)*** that belong to all objects of the class and ***behaviors (methods)*** of all objects of the class,  
Instantiating Classes

# Inheritance

- Inheritance is the ability to define classes that are extensions of other classes with new and/or specialized attributes and methods
- For instance class Dog inherits from class Animal, meaning that Dog has (inherits) all the attributes and the methods of Animal, and can redefine some of them and add new ones
- People say: Dog “is-a” Animal, Dog “extends” Animal, the class of Animals “contains” the class of Dogs, Animal “generalizes” Dog, ...

# Summary

- Procedural and object oriented programming
- Classes and objects
- Encapsulation
- Abstraction
- Inheritance

# Class assignment

- Discuss how following concepts applies to the notion of a watch
  - Object/class
  - Attribute
  - Behavior
  - Inheritance