Course overview (1)

* Objects and classes
* Understanding class definitions
* Object interaction
* Collections –Grouping objects
* More sophisticated behaviour –libraries
* Designing classes
* Well-behaved objects -testing, maintaining, debugging
* Inheritance
* Polymorphism
* Extendable, flexible class structures
* Building graphical user interfaces
* Handling errors
* Designing applications

Fundamental concepts

1. object
2. class
3. attribute
4. method
5. parameter
6. return value

Objects and classes

* 1. objectsrepresent ‘things’ from the real world, or from some problem domain: the red car down there in the car park
  2. the green circle at position (230, 90) on the canvas
  3. classesrepresent all objects of a kind: car
  4. circle

Attributes

* 1. Many object instancescan be created from a single classcircle1, circle2
  2. An object has attributes: values stored in fields
  3. The class defines what fieldsan object has, but each object stores its own set of values(the stateof the object)all circles have diameter, xPosition, yPosition, colour
  4. circle1hasdiameter 68: circle2hasdiameter 40

Methods and parameters

1. Objects have operations which can be invoked (Java calls them methods).
2. Methods may have parameters to pass additional information needed to execute.

Return values

* 1. None of the methods in the figures project return a data value: they have voidreturn typesmethods may return a result via a return value.
  2. Such methods have a non-void return type.
  3. More on this in the next lecture.

Source code

1. Each class has source code (Java code) associated with it that defines its details (fields and methods)
2. The source code is commented, to summarise what each method should do
3. Comments are used to automatically generate documentation

Summary

1. Java is an object-oriented language, used for development of a wide range of applications
2. BlueJis an introductory teaching environment for Java
3. Demo: classes, objects, attributes, methods, parameters, return values

Classes: Main concepts to be covered

1. fields
2. encapsulation (private, public)
3. constructors
4. parameters
5. assignment statements