# Exam 2

Inheritance:

🡪 Forming new classes based on existing ones by **extending** them.

🡪 Superclass: parent class being extended

🡪 Subclass: Child class that inherits behavior from superclass

🡪 Why?

🡪 Allow one definition to work on multiple related types

🡪 Allow for specialization

🡪 What is polymorphism?

🡪 Refers to determining which program behavior to execute depending on data types. (Overloading)

🡪 how do we derive classes from one another (via extends)?

🡪 All classes derive from the Object Class

🡪 what are the rules about when and how we can derive classes?

🡪 1 parent can have many child classes (called sibling)

🡪 Every child class has exactly 1 parent

🡪 Can’t see data and method if it’s private

🡪 Must call parent’s constructor first (super)

🡪 If redefine, it’s hidden

🡪what are abstract methods and classes?

🡪 A placeholder in a class hierarchy that represents a generic concept

🡪 Can’t be instantiated

🡪 Can be extended

🡪 Has no body {…} 🡪 ; (>0 abstract method)

🡪 abstract class’s child must override abstract method

🡪 abstract method can’t be final/ static

🡪 Why: Allow us to establish common elements in a hierarchy that are too generic to instantiate

🡪 what are final variables, methods, and classes?

🡪 Final class can’t be extended

🡪 Final method can’t be override

🡪 what is overriding, and which version of a method will get used?

🡪 A child class can *override* the definition of an inherited method in favor of its own

🡪The child's method must have the same signature as the parent's method, but can have a different body

🡪 The type of the object executing the method determines which version of the method is invoked

Interfaces:

* what are they, and how do they relate to polymorphism or abstract classes?

🡪 A set of unimplemented methods called abstract methods (no fields are allowed)

🡪 Interface is a type

🡪 Can implement multiple of interface

* how do we use (implement) an interface?
* can we treat an interface like an object?
* why are they needed?
* what does it mean for a class to implement an interface?

🡪 what are the rules about extending interfaces or implementing multiple interfaces?

🡪 Class implementing interface has to provide implementations of all the methods found in that interface

* what if two interfaces share the same method names, or if a class has already implemented an interface's method?