Method overriding

* Method overriding is a language feature in which a class can provide an implementation of a method that is already provided by one of its parent classes.
* The implementation in this sub-class replaces (that is, overrides) the implementation in the parent class.
* When you define a method with the same name as that of a parent class, that new method replaces the inherited definition
* The new method must have the same return type and take the same number and type of parameters as the method you are overriding. Here’s an example:

**Example:**

@interface MyClass : NSObject {

}

- (int)myNumber;

@end

@implementation MyClass : NSObject {

}

- (int)myNumber {

return 1;

}

@end

@interface MySubclass : MyClass {

}

- (int)myNumber;

@end

@implementation MySubclass

- (int)myNumber {

return 2;

}

@end

* If you create an instance of MyClass and send it a myNumber message, it returns 1. If you create an instance of MySubclass and send it a myNumber message, it returns 2.
* The subclass’s method must have the same name and parameter list as the superclass's overridden method.
* In addition to completely replacing an existing implementation, you might want to extend a superclass’s implementation. To do this, you can invoke the superclass’s implementation using the superkeyword.

**Example:**

@implementation MySubclass

- (int)myNumber {

int subclassNumber = [super myNumber] + 1;

return subclassNumber;

}

@end

* Within a method definition, super refers to the parent class (the superclass) of the current object.
* You send a message to super to execute the superclass’s implementation of a method.
* In the above example, the implementation of myNumber by MySubclass simply adds 1 to whatever value is returned by the implementation of MyClass.

References:

* https://developer.apple.com/library/ios/documentation/General/Conceptual/DevPedia-CocoaCore/MethodOverriding.html