# Objective C Categories

* You use categories to define additional methods of an existing class—even one whose source code is unavailable to you—without subclassing.
* Consider a case where you want to add a new method to your app.
* Let's say you have NSString but you hope you had a method that will capitalize the first letter of NSString.
* Categories add new methods to a class - without sub classing.
* You can add new methods to a class but you cannot add data to a class.

|  |
| --- |
| * #import "XYZPerson.h" |
|  |
| * @interface XYZPerson (XYZPersonNameDisplayAdditions) |
| * - (NSString \*)lastNameFirstNameString; |
| * @end |

* In this example, the XYZPersonNameDisplayAdditions category declares one additional method to return the necessary string.
* A category is usually declared in a separate header file and implemented in a separate source code file. In the case of XYZPerson, you might declare the category in a header file called XYZPerson+XYZPersonNameDisplayAdditions.h.
* Even though any methods added by a category are available to all instances of the class and its subclasses, you’ll need to import the category header file in any source code file where you wish to use the additional methods, otherwise you’ll run into compiler warnings and errors.
* The category implementation might look like this:

|  |
| --- |
| * #import "XYZPerson+XYZPersonNameDisplayAdditions.h" |
|  |
| * @implementation XYZPerson (XYZPersonNameDisplayAdditions) |
| * - (NSString \*)lastNameFirstNameString { |
| * return [NSString stringWithFormat:@"%@, %@", self.lastName, self.firstName]; |
| * } |
| * @end |