

Choose a template for your new target:

iOS watchOS tvOS macOS Cross-platform

 Filter

Application Extension

Extension

Preview Extension

Extension

Extension



Thumbnail
Extension



Today Extension

Test



iOS UI Testing
Bundle



iOS Unit Testing
Bundle

Application



Cancel

Previous

Next



A screenshot of Xcode showing a Swift test file named `window_shopperTests.swift`. The code defines a test class `window_shopperTests` that inherits from `XCTestCase`. It includes standard setup and teardown methods and two test functions: `testExample` and `testPerformanceExample`.

```
8
9 import XCTest
10
11 class window_shopperTests: XCTestCase {
12
13     override func setUp() {
14         super.setUp()
15         // Put setup code here. This method is called before the invocation of each test method in the class.
16     }
17
18     override func tearDown() {
19         // Put teardown code here. This method is called after the invocation of each test method in the class.
20         super.tearDown()
21     }
22
23     func testExample() {
24         // This is an example of a functional test case.
25         // Use XCTAssert and related functions to verify your tests produce the
26         // expected results.
27     }
28
29     func testPerformanceExample() {
30         // This is an example of a performance test case.
31         self.measure {
32             // Put the code you want to measure the time of here.
33         }
34     }
35 }
```

The screenshot shows the Xcode interface with the following details:

- Project Navigator:** Shows the project structure for "window-shopper". The "Wage.swift" file is selected in the "Model" folder.
- Editor:** Displays the contents of the "Wage.swift" file. The code is as follows:

```
//  
//  Wage.swift  
//  window-shopper  
//  
//  Created by Mark Price on 6/19/17.  
//  Copyright © 2017 Devslopes. All rights reserved.  
  
import Foundation  
class Wage {  
    class func getHours(forWage wage: Double, andPrice price: Double) -> Int {  
        return Int(ceil(price / wage))  
    }  
}  
15
```

The code defines a class named "Wage" with a static function "getHours" that takes two double parameters: "wage" and "price", and returns an integer representing the number of hours.

The screenshot shows the Xcode interface with the 'Wage.swift' file open in the editor. The code defines a class 'Wage' with a static method 'getHours'. The 'Identity and Type' inspector on the right shows the file's properties: Name is 'Wage.swift', Type is 'Default - Swift Source', Location is 'Relative to Group', and Full Path is '/Users/markprice/Dropbox (Devslopes)/Devslopes Team Folder/Courses and Lessons/iOS-11-course/Section3/window-shopper/window-shopper/Model/Wage.swift'. The 'Target Membership' section is highlighted with a green border and contains two entries: 'window-shopper' and 'window-shopperTests', both with checkboxes checked.

```
1 //  
2 //  Wage.swift  
3 //  window-shopper  
4 //  
5 //  Created by Mark Price on 6/19/17.  
6 //  Copyright © 2017 Devslopes. All rights reserved.  
7 //  
8  
9 import Foundation  
10 class Wage {  
11     class func getHours(forWage wage: Double, andPrice price:  
12         Double) -> Int {  
13         return Int(ceil(price / wage))  
14     }  
15 }
```

Add target membership for tests

The screenshot shows the Xcode interface with a project named "window-shopper". The left sidebar displays the project structure, including "Model/Wage.swift", "Views/CurrencyTextField.swift", "Controller/MainVC.swift" and "AppDelegate.swift", "Main.storyboard", "Assets.xcassets", "LaunchScreen.storyboard", and "Info.plist". The "window-shopperTests" group contains "window_shopperTests.swift" and "Info.plist". The "Products" section is also visible.

The main editor area shows the content of "window_shopperTests.swift". The code includes standard setup and teardown methods:

```
override func setUp() {
    super.setUp()
    // Put setup code here. This method is called before the
    // invocation of each test method in the class.
}

override func tearDown() {
    super.tearDown()
}
```

It also includes a placeholder test function:

```
func testExample() {
    // This is an example of a functional test case.
    // Use XCTAssert and related functions to verify your tests
    // produce the correct results.
}
```

A green rounded rectangle highlights the following test function:

```
func testGetHours() {
    XCTAssertEqual(Wage.getHours(forWage: 25, andPrice: 100) == 4)
    XCTAssertEqual(Wage.getHours(forWage: 15.50, andPrice: 250.53)
        == 17)
}
```

Pro Tip

Good functions accept input and
return output

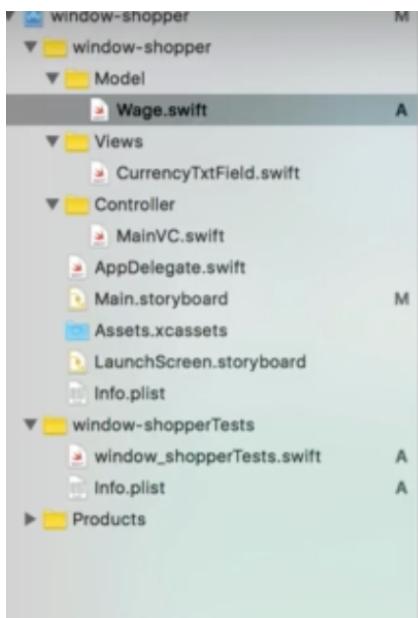
Bad functions set global variables
and rely on other functions to
work

Pro Tip

Always run (and pass) your unit tests before you:

- 1) Create pull requests
- 2) Deploy to device
- 3) Send app to beta testers
- 4) Submit apps to App Store!!!

```
window-shopper > window-shopperTests > window_shopperTests.swift > M testGetHours()  
12  
13     override func setUp() {  
14         super.setUp()  
15         // Put setup code here. This method is called before the invocation of each test  
16         // method in the class.  
17     }  
18  
19     override func tearDown() {  
20         // Put teardown code here. This method is called after the invocation of each  
21         // test method in the class.  
22         super.tearDown()  
23     }  
24  
25     func testExample() {  
26         // This is an example of a functional test case.  
27         // Use XCTAssert and related functions to verify your tests produce the correct  
28         // results.  
29     }  
30  
31     func testGetHours() {  
32         XCTAssert(Wage.getHours(forWage: 25, andPrice: 100) == 4)  
33         XCTAssert(Wage.getHours(forWage: 15.50, andPrice: 250.53) == 17)  
34     }  
35
```



```
1 //  
2 //  Wage.swift  
3 //  window-shopper  
4 //  
5 //  Created by Mark Price on 6/19/17.  
6 //  Copyright © 2017 Devslopes. All rights reserved.  
7 //  
8  
9 import Foundation  
10 class Wage {  
11     class func getHours(forWage wage: Double, andPrice price: Double) -> Int {  
12         return Int(round(price / wage))  
13         //|return Int(ceil(price / wage))  
14     }  
15 }  
16 }
```

Now try
changing the
logic of the
function

**== 17 test
fails because
function now
return 16 for
that
calculation**

```
        results.  
26    }  
27  
28    ✘ func testGetHours() {  
29        XCTAssertEqual(Wage.getHours(forWage: 25, andPrice: 100) == 4)  
30        XCTAssertEqual(Wage.getHours(forWage: 15.50, andPrice: 250.53) == 17) ✘ XCTAssertTrue f...  
31    }  
32  
33    ◊ func testPerformanceExample() {  
34        // This is an example of a performance test case.  
35        self.measure {
```

```
--  
27  
28  
29     func testGetHours() {  
30         XCTAssertEqual(Wage.getHours(forWage: 25, andPrice: 100) == 4)  
31         XCTAssertEqual(Wage.getHours(forWage: 15.50, andPrice: 250.53) == 17) ✘ XCTAssertTrue f...  
32     }  
33  
34     func testPerformanceExample() {  
35         // This is an example of a performance test case.  
36         self.measure {
```

```
Test Suite 'Selected tests' started at 2017-06-19 17:01:57.311  
Test Suite 'window-shopperTests.xctest' started at 2017-06-19 17:01:57.311  
Test Suite 'window_shopperTests' started at 2017-06-19 17:01:57.311  
Test Case '-[window_shopperTests.window_shopperTests testGetHours]' started.
```

```
4  
16
```

```
/Users/markprice/Dropbox (Devslopes)/Devslopes Team Folder/Courses and Lessons/iOS-11-course/  
Section3/window-shopper/window-shopper/window-shopperTests/window_shopperTests.swift:30:  
error: -[window_shopperTests.window_shopperTests testGetHours] : XCTAssertTrue failed -
```