

The screenshot shows a storyboard scene with three UI elements:

- A top text field labeled "Your hourly wage".
- A middle text field labeled "Item price".
- A large floating text view containing the number "32" and the word "hours".

An "Add New Alignment Constraints" popover is open at the bottom right, specifically for the "hours" text view. The popover contains the following options:

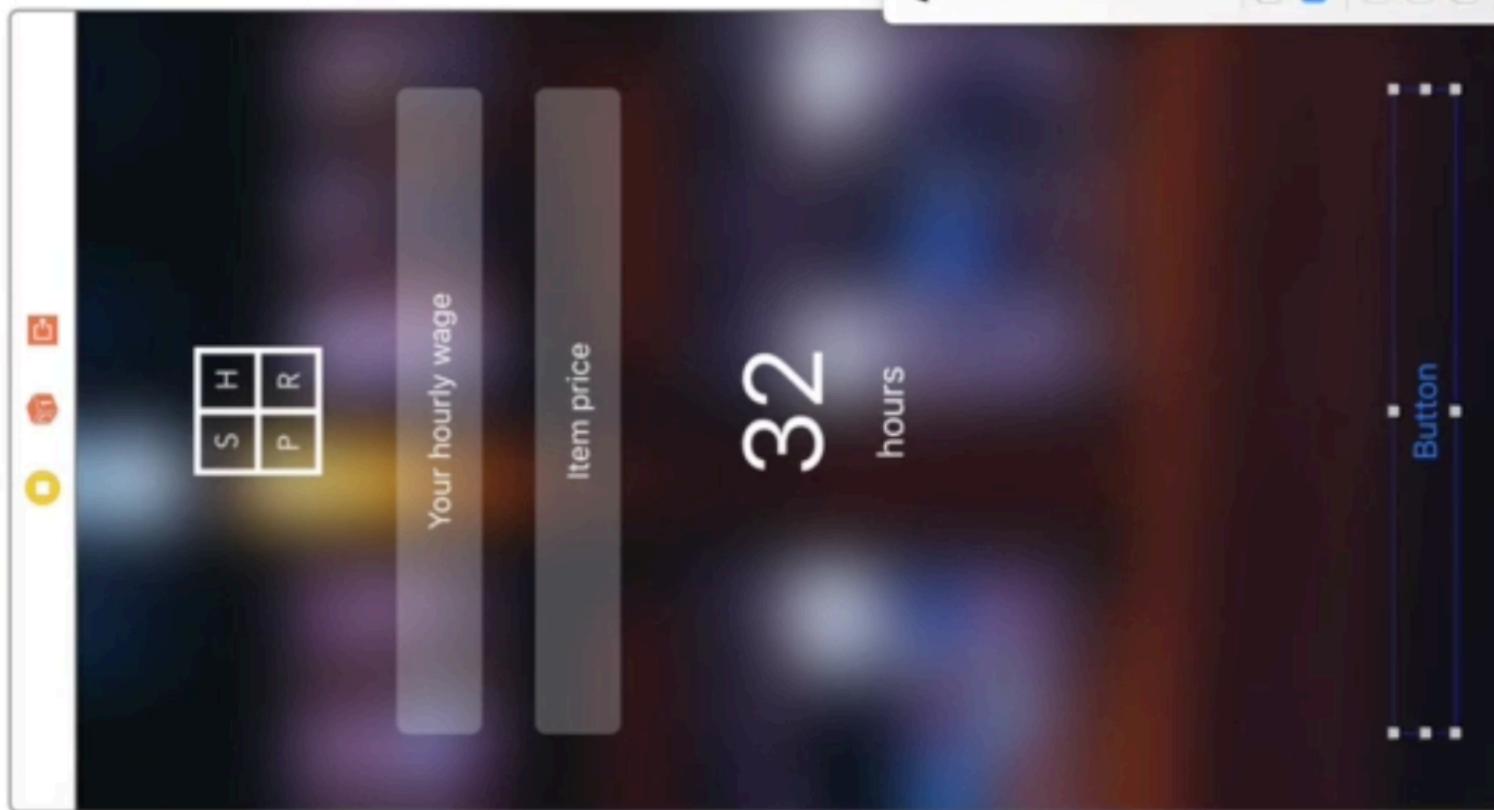
- Leading Edges
- Trailing Edges
- Top Edges
- Bottom Edges
- Horizontal Centers
- Vertical Centers
- Baselines
- Horizontally in Container** (checkbox checked, value: 0)
- Vertically in Container

At the bottom of the popover is a button labeled "Add 1 Constraint".

The right side of the screen shows the Xcode Attributes Inspector with various text-related settings like Lines, Behavior, Baseline, Line Break, and Autoshrink.

low-shopper: Succeeded | Today at 5:04 PM ▲ 4 ● 1

> Main.storyboard (Base) > MainVC Scene > MainVC > View > Button



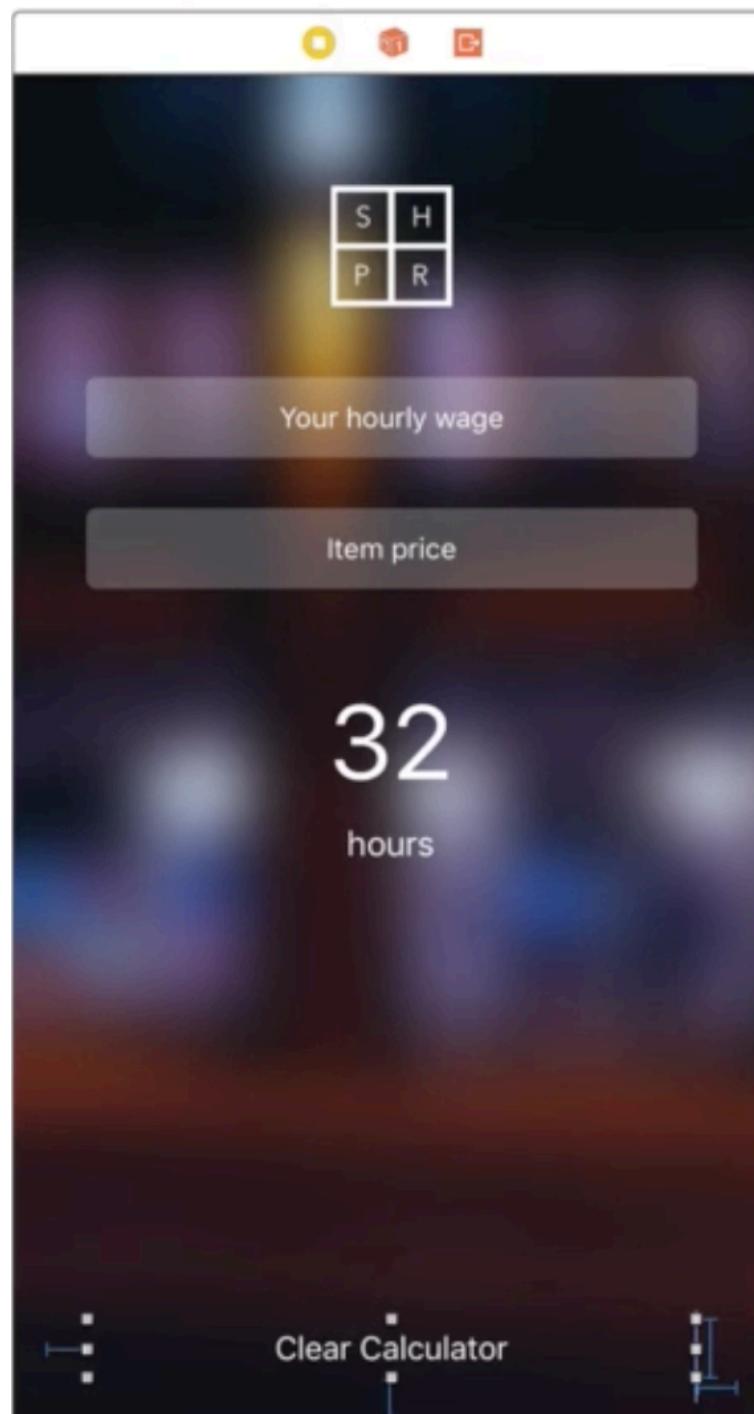
The screenshot shows the Xcode interface with the storyboard open. The top navigation bar includes icons for undo, redo, and search. The main area displays the storyboard scene with three buttons. A constraint editor overlay is visible for the top button, showing the following settings:

- Type**: Button
- State Config**: Deactivated
- Title**: Placeholder
- Font**: Placeholder
- Text Color**: Placeholder
- Shadow Color**: Placeholder
- Image**: Placeholder
- Background**: Placeholder
- Accessibility**: Placeholder
- Shadow Offset**: Placeholder
- Drawing**: Placeholder
- Line Break**: Placeholder
- Drag and Drop**: Placeholder

On the right side of the interface, there are several toolbars and panels, including:

- File
- Editor
- View
- Run
- Product
- Scheme
- Build
- Cloud

At the bottom, there are tabs for MainVC, Storyboard, and Class Definition.

**Button**

Type: System  
State Config: Default  
Title: Plain  
Clear Calculator  
Font: System 16.0  
Text Color:    
Shadow Color:   Default  
Image: Default image  
Background: Default Background image  
Accessibility:  Adjusts Image Size  
Shadow Offset: 0 0  
Width:   
Height:   
 Reverses On Highlight  
 Shows Touch On Highlight  
 Highlighted Adjusts Image  
 Disabled Adjusts Image  
Line Break: Truncate Middle  
Drag and Drop:  Spring Loaded

**Control**

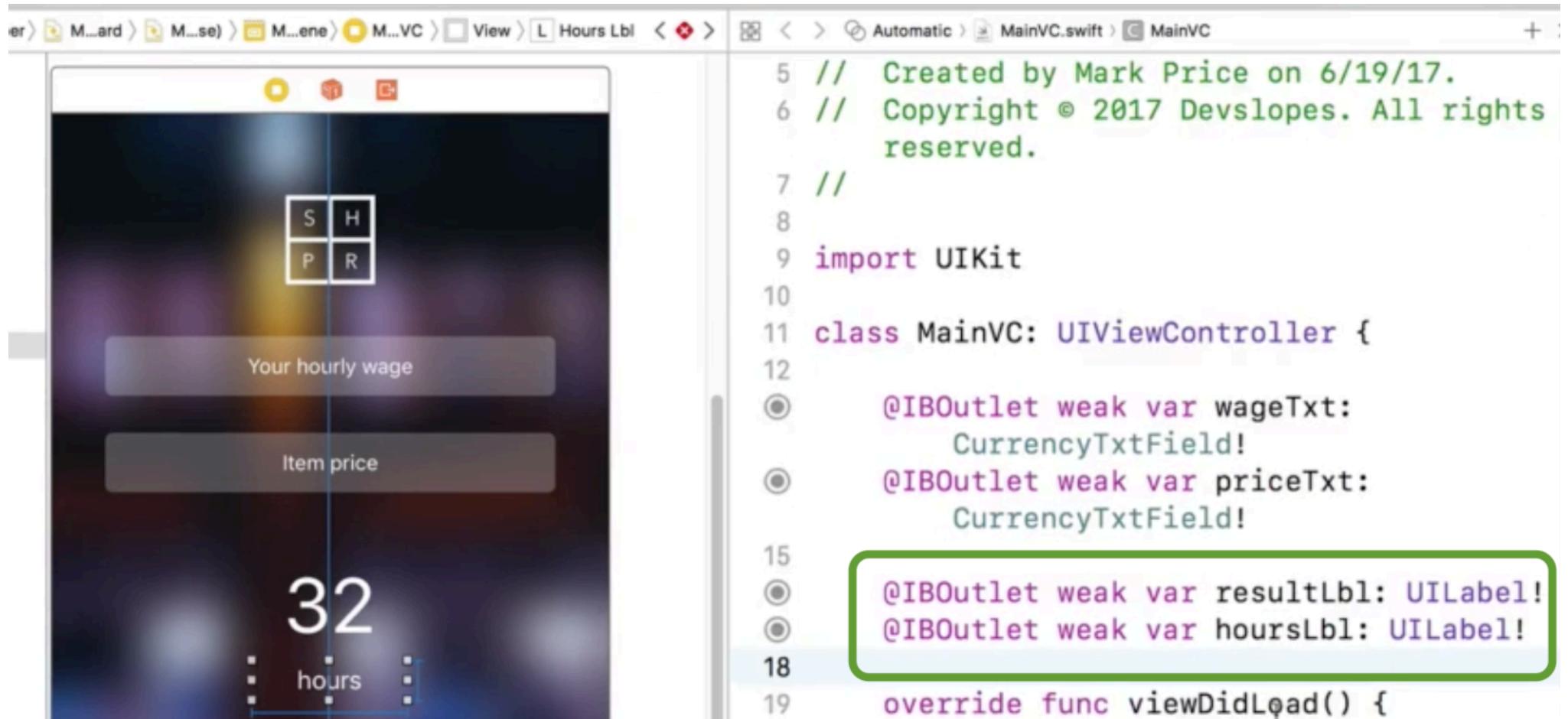
Alignment:                 
Horizontal  
                
Vertical  
State:  Selected  
 Enabled  
       

**Button** - Intercepts touch events and sends an action message to a target object when it's tapped.

**Item**

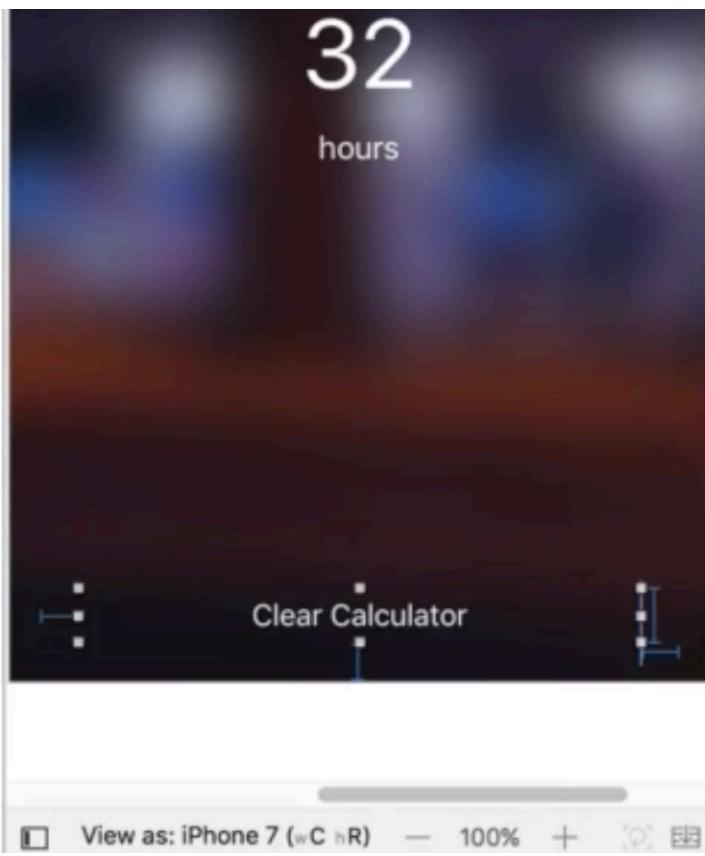
**Bar Button Item** - Represents an item on a UIToolbar or UINavigationItem object.

**Fixed Space Bar Button Item** -



The image shows a screenshot of the Xcode IDE. On the left is the storyboard, which displays a dark-themed view controller with several UI elements: a top bar with icons, a 2x2 grid button labeled S, H, P, R, two text input fields labeled "Your hourly wage" and "Item price", and a large result label at the bottom displaying "32". On the right is the code editor showing MainVC.swift. The code includes comments about the creation date and copyright, imports UIKit, defines a MainVC class, and declares outlets for the wageTxt, priceTxt, resultLbl, and hoursLbl labels. The last two outlet declarations are highlighted with a green rounded rectangle.

```
5 // Created by Mark Price on 6/19/17.  
6 // Copyright © 2017 Devslopes. All rights  
7 // reserved.  
8  
9 import UIKit  
10  
11 class MainVC: UIViewController {  
12  
13     @IBOutlet weak var wageTxt:  
14         CurrencyTextField!  
15     @IBOutlet weak var priceTxt:  
16         CurrencyTextField!  
17  
18     @IBOutlet weak var resultLbl: UILabel!  
19     @IBOutlet weak var hoursLbl: UILabel!  
20  
21     override func viewDidLoad() {
```



```
26 calcBtn.addTarget(self, action:  
27     #selector(MainVC.calculate),  
28     for: .touchUpInside)  
29  
30 wageTxt.inputAccessoryView =  
31     calcBtn  
32 priceTxt.inputAccessoryView =  
33     calcBtn  
34 }  
35  
36 @objc func calculate() {  
37     print("We got here")  
38 }  
39  
40 @IBAction func  
41     clearCalculatorPressed(_ sender:  
42     Any) {  
43 }
```

Section 6, Lecture 42

window-shopper > window-shopper > Controller > MainVC.swift > calculate()

```
24     calcBtn.setTitle("Calculate", for: .normal)
25     calcBtn.setTitleColor(white, for: .normal)
26     calcBtn.addTarget(self, action: #selector(MainVC.calculate), for: .touchUpInside)
27
28     wageTxt.inputAccessoryView = calcBtn
29     priceTxt.inputAccessoryView = calcBtn
30
31     resultLbl.isHidden = true
32     hoursLbl.isHidden = true
33 }
34
35 @objc func calculate() {
36     if let wageTxt = wageTxt.text, let priceTxt = priceTxt.text {
37         if let wage = Double(wageTxt), let price = Double(priceTxt) {
38             view.endEditing(true)
39             resultLbl.isHidden = false
40             hoursLbl.isHidden = false
41             resultLbl.text = "\(Wage.getHours(forWage: wage, andPrice: price))"
42         }
43     }
44 }
45
46 @IBAction func clearCalculatorPressed(_ sender: Any) {
47     resultLbl.isHidden = true
48     hoursLbl.isHidden = true
49     wageTxt.text = ""
50     priceTxt.text = ""
51 }
52 }
```

