

# Tableview cell multiple check mark selection to add cell data into single array Swift 4.2?

Asked 4 years, 8 months ago   Modified 4 years, 8 months ago   Viewed 955 times

🌟 Part of [Mobile Development](#) Collective



0



My scenario, I have loaded my JSON data into tableView with help of Codable. Here, I have added my tableView cell multiple check mark select and deselect. Now, If I am selecting tableView cell I can able to get cell data but I want to add within one array, same if I am unselecting cell It should remove from the array. Selected cell data I am moving to another ViewController. I would like to know how to do that.



My Code



```
// method to run when table view cell is tapped
func tableView(_ tableView: UITableView, didSelectRowAt indexPath:
IndexPath) {
    print("You tapped cell number \(indexPath.row).")

    if let cell = tableView.cellForRow(at: indexPath as IndexPath) {
        if cell.accessoryType == .checkmark {
            cell.accessoryType = .none
        } else {
            cell.accessoryType = .checkmark
            let item = users[indexPath.row]
            print(item) // here printing cell selection data
        }
    }
}
```

My Cell Selection current output

```
You tapped cell number 1.
User(userId: "121", active: 1, name: example_table.Name(firstname: "jack",
lastname: "m"))
You tapped cell number 2.
User(userId: "122", active: 1, name: example_table.Name(firstname: "rose",
lastname: "h"))
You tapped cell number 3.
User(userId: "123", active: 1, name: example_table.Name(firstname: "makj",
lastname: "i"))
```

[ios](#) [swift](#) [uitableview](#)

Share Follow

asked Feb 19, 2019 at 18:21



[devmikle](#)

105 2 11

## 3 Answers

Sorted by: Highest score (default)



So it sounds like you want to be able to have an array that contains all of the selected Users. Then what you would do is have an array like this instantiated in the class declaration:

0

```
var users:[Users] = []
```



Now what you should be doing is leveraging swift's protocols in order to handle the heavy lifting for you; meaning, when removing a previously selected user from the Users array, you shouldn't need a for loop, but something more familiar: contains.

```
extension User: Equatable {
    static func == (lhs: User, rhs: User) -> Bool {
        return lhs.userId == rhs.userId
    }
}
```

So now you can call this when removing or adding, for example:

```
var thisUser = User(userId: "123", active: 1, name:
example_table.Name(firstname: "makj", lastname: "i"))

if users.contains(thisUser) {
    users.removeAtIndex(users.indexOf(thisUser))
} else {
    //Add the user to the array here for example
    users.append(thisUser)
}
```

Share Follow

answered Feb 19, 2019 at 18:38



Jay

2,621 1 16 28



Don't use an extra array, add the *selected* information to your model.

0

```
struct User : Codable {
    var isSelected = false
    // ... other members
}
```



Assuming the data source array is declared as `users`, set the checkmark depending on the `isSelected` value in `cellForRowAt`

```

override func tableView(_ tableView: UITableView, cellForRowAt indexPath:
IndexPath) -> UITableViewCell {
    let cell = tableView.dequeueReusableCell(withIdentifier: "Cell", for:
indexPath)
    let user = users[indexPath.row]
    cell.accessoryType = user.isSelected ? .checkmark : .none

    ...
}

```

In didSelectRowAt just toggle isSelected and reload the row

```

override func tableView(_ tableView: UITableView, didSelectRowAt indexPath:
IndexPath) {
    users[indexPath.row].isSelected.toggle()
    tableView.reloadRows(at: [indexPath], with: .none)
}

```

To get all selected users just filter the array

```
let selectedUsers = users.filter{ $0.isSelected }
```

Share Follow

answered Feb 19, 2019 at 19:56



vadian

277k

30

358

365



You can try

0

```

var selectedArr = [Item]()
var users = [Item]()

```



```

func tableView(_ tableView: UITableView, didSelectRowAt indexPath: IndexPath) {
    print("You tapped cell number \(indexPath.row).")
    let item = users[indexPath.row]
    if let cell = tableView.cellForRow(at: indexPath) { // you can also omit
the if let and force-unwrap as in this case cell will never be nil
        if cell.accessoryType == .checkmark {
            cell.accessoryType = .none
            selectedArr.remove(where: { $0 == item })
        } else {
            cell.accessoryType = .checkmark
            selectedArr.append(item)
        }
    }
}

```



Then inside cellForRowAt

```
let cell = ///  
let item = users[indexPath.row]  
cell.accessoryType = selectedArr.contains(item) ? .checkmark : .none
```

Also make sure model named `Item` conforms to `Equatable`

given

```
for (index, element) in item.enumerated() {  
    print("Item \(index): \(element)")  
}
```

that gives

Item 0: 121, Item 1: 122, Item 2: 123

Then it's an array of Ints , so do

```
let res = item.map{ "\( $0)" }.joined(separator: ",")
```

Share Follow

edited Feb 19, 2019 at 20:44

answered Feb 19, 2019 at 18:32



[Shehata Gamal](#)

**99.2k** 8 67 89

---

I can able to get the values like Item 0: 121, Item 1: 122, Item 2: 123, using `for (index, element) in item.enumerated() { print("Item (index): (element),") }` Now how to change like `id = 121,122,123`  
@Sh\_Khan – [devmikle](#) Feb 19, 2019 at 19:02

---