

```
tapgesture
import UIKit
```

```
class ViewController: UIViewController {
    @IBOutlet weak var cityImage: UIImageView!
```

```
    @IBOutlet weak var Label: UILabel!
```

```
    override func viewDidLoad() {
```

```
        //tap gesture bir dugme kullanmayacagi icin
        onu acilis pozisyonuna almak gerekiyordu o yuzden
        override altina yazdik ki image kullanıcı ile
        interaction yapma ozelligi enable olsun.
```

```
        cityImage.isUserInteractionEnabled = true
```

```
        //ardindan, gesture i bizim app in icinde
        var etmek icin, ona tasavvuri bir sabit verdik, ya
        bu ozellik sen varsin ve adin tapCity dedik.target
        self cunku bu nitelik kendisinde olacak. Selector?
        seklinde cikiyor ama biz basina # koyuyoruz.
        boylece selector a evet deyip ona yon verioyoruz,
        o yon ise yine kendisi yani
        (ViewController.tapPic)
```

```
        let tapCity =
```

```
        UITapGestureRecognizer(target: self, action:
        #selector(ViewController.tapPic))
```

```
        // once, resim interaction yapabilsin dedik,
        ikinci adimda bu interaction in tapGesture
        olmasina karar verdik simdi ise kazanilan bu
        kabiliyeti cityImage kullansin diyoruz . hatta
        satir cok acik. diyor ki, city image a gesture
        ekle o gesture ise baska birsey degil tarif edip
        yarattigim tapCity gesture u. (addaction,
        addsubview, present komutlarinin hazirlik
        asamalari var bu sekilde.. alerts notunda birinci
        sekmede de isaret etmisiz.)
```

```
cityImage.addGestureRecognizer(tapCity)  
}
```

//eger yukaridaki isler basarili yapiliyorsa
resim tiklaninca tapPic fonksiyonu olarak
istedigim sey you tapped diye yazmasi (simdilik...
buraya daha kompleks bir komut yaziyoruz yandaki
sekmede.. resmi degistir fonksiyonu, boolean a
gore bir duzenek)

```
@objc func tapPic() {  
    print("you tapped")
```

```
}
```