

Swift

Protokół – wymaganie istnienia property

```
protocol CoMaByć {  
    var ileKół: Int {get} // {set get}  
    var ileOpón: Int {get}  
}
```

W definicja protokołu nie precyzujemy o jaki rodzaj property nam chodzi

wymaganie istnienia setera i getera

```
class Pojazd: CoMaByć{  
    var ileKół: Int  
    var ileOpón: Int {  
        get{  
            return ileKół+1  
        }  
    }  
    init(ile: Int){  
        ileKół = ile  
    }  
}
```

property alokowane: {set get} właściwie nie są potrzebne

property obliczane: {set get} są właśnie dla tego rodzaju property

Swift

Protokół – wymaganie istnienia metody

```
protocol CoMaByć{
    var ileKół: Int {get} // {set get}
    var ileOpón: Int {get}
    func jakiePaliwo() -> String
}

class Pojazd: CoMaByć{
    var ileKół: Int

    var ileOpón: Int {
        get{
            return ileKół+1
        }
    }

    func jakiePaliwo() -> String{
        return "???"
    }

    init(ile: Int){
        ileKół = ile
    }
}

class Rower: Pojazd{
    override func jakiePaliwo() -> String{
        return "mięśnie"
    }
}
```

```
var mojRower: Pojazd = Rower(ile: 2)
print(mojRower.jakiePaliwo())
```

wymaganie istnienia metody

Przykład wystąpienia polimorfizmu

implementacja metody (musi być)

„nadpisanie” metody wirtualnej