

```

import UIKit

//1.
class GameUser {

    let name: String
    var activity: Int = 0

    init(name: String) {
        self.name = name
    }
}

var user = GameUser(name: "Artemiy") {
    didSet{
        print("\(user.name)'s activity is \(user.activity)")
    }
}
user.activity += 1
//You wanted to print the user's activity each time it changes, but nothing is
    printed. Why? And what simple change can you provide to make that work?

//2. Tallinn Coordinates: (59.437222 lat, 24.745278 long)
struct Location {
    let latitude: Float
    let longitude: Float
}
/*
    I'm not defining any initialiser directly, can I still create an instance
        of Location with Tallin's coordinates without adding any additional
        code?
    If no, write a code that should be added?
    If yes, initialise the instance. How would you know which one of two
        floats passed will be used for latitude and which one for longitude?
*/

//3. Current estonian president is Kersti Kaljulaid.
class President {
    let name: String
    let surname: String
}
/*
    I'm not defining any initialiser directly, can I still create an instance
        of President with current president data without adding any additional
        code?
    If no, write a code that should be added?
    If yes, initialise the instance. How would you know which one of two
        strings passed will be used for name and which one for surname?
*/

//4. Initialise a teacher role:
struct User {

    enum Role {
        case student, teacher
    }
}

```

```

    let role: Role
    let name: String
}
//let teacherRole =

```

//5. Consider you're writing a code using a MVC design pattern only. Which of three roles will you define for Int?

```

//6.
enum Background {
    case color(UIColor)
    case clear
}
enum Winner: Int {
    case gold, silver, bronze
}

```

//What is the type of associated value that color case has? What is the raw value type in Winner enum?

```

//7.
class Server {

    let domain: String
    var port: Int

    weak var manager: ConnectionManager?

    init(domain: String, port: Int) {
        self.domain = domain
        self.port = port
    }
}

```

```

class ConnectionManager {
    var server: Server?
}

```

```

let connectionManager: ConnectionManager? = ConnectionManager()
let apple = Server(domain: "apple.com", port: 443)
let anotherApple = apple
anotherApple.port = 80
let google: Server? = Server(domain: "google.com", port: 443)
google?.manager = connectionManager
connectionManager?.server = apple
/*

```

In these question write how many references each object has after all the code was called
 google server object:
 apple server object:
 connection manager object:

```

*/

```

//8. What is wrong with this code? And how can you fix it (if I still want a property 'name')

```

enum Fruit {
    case orange, apple
    let name = "fruit"
}

```