



**Bjit**



## Exam on Swift OOP: 24 Nov 22

Total points **14/24**

Email \*

yeasirarefin@just.edu.bd

**0 of 0 points**

Class ID \*

30021

Name \*

Yeastir Arefin Tusher

Quiz Section

**14 of 24 points**

- ✓ Will anything go wrong with the following code snippet? Shortly explain \*4/4 your answer. If you think any error will occur in the code, then explain how it could be resolved as well.

```

1. enum SomeEnum {
2.     case a(Int)
3.     case b(String)
4. }
5. protocol SomeProtocol {
6.     associatedtype Element
7. }
8. extension SomeProtocol where Element: Equatable {
9.     func isEquatable() -> Bool {
10.         return true
11.     }
12. }
13. struct SomeStruct<T>: SomeProtocol {
14.     typealias Element = T
15. }
16. let something = SomeStruct<SomeEnum>()
17. something.isEquatable()

```

The associated type SomeEnum in the someStruct is not conforming to the Equatable protocol. We should conform the someEnum enum with Equatable protocol. Then the code will work perfectly.

- ✓ In the **struct** shown below, what kind of property is courseGPA? \*

**2/2**

```

1. struct Course {
2.     var courseCode : String
3.     var credits : Double
4.     var courseGPA: Double = 0
5.     var gradePoint : Double {
6.         didSet {
7.             courseGPA = gradePoint * credits
8.         }
9.     }
10.    init(courseCode: String, credits: Double,
11.         gradePoint: Double){
12.         self.courseCode = courseCode
13.         self.credits = credits
14.         self.gradePoint = gradePoint
15.     }

```

15. }

I. Computed property

II. Stored property ✓

III. Static property

IV. Optional property

✗ In the **struct** shown below, what kind of property is gradePoint? \*

0/2

```
1. struct Course {  
2.     var courseCode : String  
3.     var credits : Double  
4.     var courseGPA: Double = 0  
5.     var gradePoint : Double {  
6.         didSet {  
7.             courseGPA = gradePoint * credits  
8.         }  
9.     }  
10.    init(courseCode: String, credits: Double,  
11.        gradePoint: Double){  
12.            self.courseCode = courseCode  
13.            self.credits = credits  
14.            self.gradePoint = gradePoint  
15.        }  
16. }
```

I. Computed property ✗

II. Stored property

III. Static property

IV. Optional property

✓ What is the core objective of this code snippet? \*

2/2

```
struct Bank {  
    static var coinsInBank = 10_000  
    static func vendCoins(var numberOfCoinsToVend: Int) -> Int {  
        let number = min(numberOfCoinsToVend, coinsInBank)  
        coinsInBank -= number  
        return number  
    }  
    static func receiveCoins(coins: Int) {  
        coinsInBank += coins  
    }  
}  
  
class Player {  
    var coinsInPurse: Int  
    init(coins: Int) {  
        coinsInPurse = Bank.vendCoins(coins)  
    }  
    func winCoins(coins: Int) {  
        coinsInPurse += Bank.vendCoins(coins)  
    }  
    deinit {  
        Bank.receiveCoins(coinsInPurse)  
    }  
}
```

a. Whenever the player is removed from the game, its coins are returned to the bank. ✓

b. Whenever the player is removed from the game, its coins are not returned to the bank.

c. Struct and classes can be worked together

d. None of the above

✓ What will be the output of this program? \*

2/2

```
class Singer {  
    var name = "Taylor Swift"  
    var genre = "Pop"
```

```

}

var singer = Singer()
print(singer.genre)

var singerCopy = singer
singerCopy.name = "Eminem"
singerCopy.genre = "Rapper"

print(singer.genre)
print(singerCopy.name)

```

Pop  
Rapper  
Taylor Swift

Rapper  
Eminem

Option 1

Option 2

Pop  
Rapper  
Eminem

Rapper  
Eminem  
Pop

Option 3



Option 4

✓ What will be the output of the following code snippet? \*

2/2

```

1. struct StudentStruct {
2.     var id: Int
3.     var name: String
4. }
5. func changeId(student: StudentStruct?) {
6.     if var student = student {
7.         student.id = 0
8.     }
9. }
10. var s1 = StudentStruct(id: 1, name: "Will")
11. changeId(student: s1)
12. print(s1.id)

```

I. 0

II. 1



III. nil

IV. nan

✓ What will be the output of the following code snippet? \*

2/2

```

1. class StudentClass {
2.     var id: Int
3.     var name: String
4.     init(id: Int, name: String) {
5.         self.id = id
6.         self.name = name
7.     }
8. }
9. func changeId(student: StudentClass?) {
10.     if var student = student {
11.         student.id = 0
12.     }

```

```
13.     }
14.     var sl = StudentClass(id: 1, name: "Bill")
15.     changeId(student: sl)
16.     print(sl.id)
```

I. 1

II. 0 ✓

III. nil

IV. nan

✗ What's wrong with the following code snippet? Shortly explain your answer. If you think any error will occur in the code, then explain how it could be resolved as well.

\*0/4

```
1. enum DataError {
2.     case nullError
3. }
4. func doSomething(with data: Data?) throws {
5.     guard let _ = data else {
6.         throw DataError.nullError
7.     }
8. }
9. let data = Data()
10.    doSomething(with: data)
```

This snippet will throw an DataError.nullError. Because the initialiser Data() class is not defined in the snippet. It is nil

✗ What's wrong with the following code snippet? \*

0/2

```
1. class Parent {
2.     var pId: Int = 100
3. }
4. class Child: Parent {
5.     var cId: Int = 200
6. }
7. var obj1 = Child()
8. var obj2 = obj1 as Parent
9. obj2.cId = 404
```

A. In line #8, we must use as? or as! operator instead of as, otherwise we'll get ✗ an error

B. Parent class has no member called 'cId' ✓

C. Objects of Child class cannot be typecast as an object of Parent class ✗

D. obj2 is immutable, therefore value of cId cannot change

E. There is nothing wrong with this code snippet

✗ In the struct shown below, what kind of property is area? \*

0/2

```
1. struct Rect {
2.     var height : Double?
3.     var width : Double?
4.     var area: Double? {
5.         if let height, let width {
6.             height * width
7.         }
8.         return nil
9.     }
10. }
```

- I. Computed property
- II. Stored property
- III. Static property
- IV. Optional property

✖

This form was created outside of your domain. - [Terms of Service](#) - [Privacy Policy](#)

Google Forms