

## TELUS AI - Programming - Dec 2024

1. Bob is working on iOS Development using SwiftUI. He wants to add control through which a user can select from a set of mutually exclusive values. Which of the given options can he use in the app view?

1. Slider
2. Picker
3. ColorPicker
4. DatePicker

2. You can create recursive enumerations in Swift. An instance of this is as follows:

```
enum List<T> {  
    case node(T, List)  
}
```

This code on execution shows a compilation error. How do you fix it?

1. By adding the `direct` keyword
2. By adding the `implicit` keyword
3. By adding the `indirect` keyword
4. The error cannot be resolved

3. What will be the output of the following code?

```
struct Rectangle {  
    var width = 0  
    var height = 0  
}  
class Square {  
    var fig1 = Rectangle ()  
    var sheer = false  
    var sidel = 0.0  
    var figname: String?  
}  
let home = Rectangle(width: 45, height: 35)  
var school = home  
school.width = 48  
print (school.width)  
print (home.width)
```

- |          |          |
|----------|----------|
| 1. 48 45 | 3. 48 45 |
| 2. 45 48 | 4. 48 48 |

4. What is the output of the following code snippet?

```
var linear = ["one", "two", "four", "seven"]
linear.sort { (x: String, y: String) -> Bool in
    return x < y
}
print(linear)
```

1. ["one", "two", "four", "seven"]
2. ["seven", "two", "four", "one"]
3. ["four", "one", "seven", "two"]
4. ["two", "seven", "one", "four"]

5. John is developing an iOS app using SwiftUI. He decided to use a geometric view state with a container view. Also, that should define its inner content as a function of its own coordinate space. Then which of the following options can he choose for this requirement?

1. GeometryReader
2. GeometryProxy
3. GeometryEffect
4. Anchor

6. Identify the error in the code given below:

```
func Product(_ x: Int, y: Int) -> Int { return x*y }
var ans = Product(x:5, y:6 )
```

1. No error
2. Extraneous argument label
3. Missing argument label
4. Use of unresolved identifier

7. Bob is working on iOS Development using SwiftUI. He wants to add a selector to his UI through which a user should be able to select a value from a bounded linear range of values. Which of the given options can he use to add to his view?

1. Slider
2. Toggler
3. Stepper
4. Picker

8. Ben has started building an iOS application using Swift. He has an application view that contains an image as a background. He wishes to make that image-background blurry. Which of the following options given below will help him accomplish the aforementioned task?

1. Using "Opaque: true" with image function will make it blurry
2. Using class CACornerRadius
3. Using UIKit stuff via UIViewRepresentable protocol
4. Using UIBlurEffect with UIVisualEffectView

9. In company XYZ, some employees have been assigned to departments. Other employees are in a queue and have not been assigned a department yet. If you have to write a piece of Swift code that conveys this information successfully, what would you write?

1.

```
if let name = employee.department#.name {  
    println("Department name is \(name)")  
} else {  
    println("Department is yet to be assigned")  
}
```

2.

```
if let name = employee.department.name {  
    print("Department name is \(name)")  
} else {  
    print("Department is yet to be assigned")  
}
```

3.

```
if let name = employee.department?.name {  
    print("Department name is \(name)")  
} else {  
    print("Department is yet to be assigned")  
}
```

4.

```
if let name = employee.department!.name {  
    print("Department name is \(name)")  
} else {  
    print("Department is yet to be assigned")  
}
```

10 .Which of the following pieces of code will create mutable instances of the variable "test"?

1.

```
let Test: Int=10
```

2.

```
struct NumberTest {  
var test = 10  
}
```

```
let Instance = NumberTest()
```

3.

```
class TestClass {  
var test = 10  
}
```

```
let Instance = TestClass ()  
Instance.test = 9
```

4.

```
let Test: String="Computer"
```