## Kotlin Exam

\*\*Instructions:\*\*

- This exam consists of both theory and practice questions.

- The exam duration is 2 hours.

- Answer all the questions.

- For theory questions, provide your answers in the space provided.

- For practice questions, write the code in the space provided.

\*\*Theory Questions:\*\*

1. What is Kotlin? How is it different from Java?

- Your Answer:

2. What are the key features of Kotlin?

- Your Answer:

3. Explain the following terms in Kotlin:

- a. Null Safety:

Null Safety dùng để đảm bảo nếu như giá trị của biến đó là null thì chương trình vẫn chạy được và không bị crash hay sao ý ạ 😐

- b. Data Classes:

Data class là class có giá trị khởi tạo của đối tượng được ghi bên cạnh tên của Class đó. Đối với data class thì giá trị khởi tạo của đối tượng phải có ít nhất là một giá trị. Và khai báo data class thì phải có chữ data được trước tên class đó

- c. Coroutines:

4. What are the primary differences between `val` and `var` in Kotlin?

- Val là kiểu dữ liệu mà sau khi biến được gán giá trị thì không thể thay đổi được giá trị trong quá trình xây dựng và thực thi chương trình

- Var là kiểu dữ liệu mà sau khi biến được gán giá trị thì biến đó có thể thay đổi được giá trị trong quá trình xây dựng và thực thi chương trình

5. Explain the difference between functions and higher-order functions in Kotlin.

- Funtion là một hàm để để thực hiện một chức năng nào đó của chương trình. Một hàm có thể có tham số truyền vào hoặc không tùy thuộc vào chức năng của hàm đó. Một hàm sẽ bao gồm tên hàm, giá trị tham số truyền vào và kiểu dữ liệu trả về của hàm đó. Một hàm có thể trả về giá trị Unit

- Higher-order function là một hàm mà giá trị tham số truyền vào của nó có thể là một hàm khác hoặc giá trị trả về là một hàm khác

\*\*Practice Questions:\*\*

1. Write a Kotlin class named `Student` that has the following properties: `name` (String), `age` (Int), `roomNumber` (Int), and `gradeLevel` (Int). Provide a function named `displayInfo()` that prints the student's information.

```kotlin

// Your code here

```

2. Write a Kotlin class named `Room` that has a property named `number` (Int) and a property named `capacity` (Int). Provide a function named `isFull()` that returns true if the room is full (i.e., the number of students in the room is equal to the capacity).

```kotlin

// Your code here

```

3. Write a Kotlin program that creates an instance of the `Room` class and checks if the room is full.

```kotlin

// Your code here

```

4. Create a superclass named `Resident` with a function named `checkIn()` that prints "Resident checked in." Create a subclass named `Student` that inherits from `Resident` and overrides the `checkIn()` function to print "Student checked in." Create an instance of the `Student` class and call the `checkIn()` function.

```kotlin

// Your code here

```

5. Write a Kotlin interface named `Paymentable` with a function named `makePayment()` that calculates and returns the total payment for a dormitory resident. Create a class named `Student` that implements the `Paymentable` interface and provides its own implementation of the `makePayment()` function. Create an instance of the `Student` class and call the `makePayment()` function.

```kotlin

// Your code here

```

6. Write a Kotlin class named `Dormitory` that has a property named `residents` (a list of `Student` objects). Provide a function named `getTotalOccupancy()` that calculates and returns the total number of students in the dormitory.

```kotlin

// Your code here

```

7. Create an instance of the `Dormitory` class and demonstrate the usage of the `getTotalOccupancy()` function.

```kotlin

// Your code here

```

8. Explain the concepts of inheritance and polymorphism in Kotlin.

Tính kế thừa là các lớp con có thể kế thừa, sử dụng và override lại các tính chất, method của lớp cha

Tính đa hình là từ một phương thức cùng tên ta có thể có nhiều các phương thức khác nhau ở tham số truyền vào kiểu giá trị trả về. Điều này thể hiện qua tính chất overloading và overriding

9. What is an abstract class in Kotlin? How does it differ from an interface?

Lớp trừu tượng trong Kotlin là một lớp dùng để khai báo các tính chất của một đối tượng. Lớp trừu tượng giống như một khung để khởi tạo các đối tượng có cùng chung tính chất

Interface dùng để khai báo các phương thức của các đối tượng sẽ implement nó. Trong kotlin interface cho phép các method trong nó được khai báo cụ thể thay vì chỉ có tên method như Java. Tuyệt cà là vời

* Lớp trừu tượng dùng để biết đối tượng kế thừa sẽ có những tính chất, thuộc tính gì. Interface sẽ cho biết đối tượng implements sẽ có các method gì

10. Explain the concept of encapsulation in OOP and how it is implemented in Kotlin.

Tính đóng gói dùng giới hạn các phạm vi truy cập của các thuộc tình hay phương thức của một đối tượng. Điều này giúp hạn chế việc dữ liệu bị thay đổi khi nhiều lớp có thể truy cập vào phương thức hay thuộc tính

11. Describe the purpose and usage of the `super` keyword in Kotlin.

Super dùng trong trường hợp lớp con kế thừa lại các thuộc tính của lớp cha. Để không phải viết lại những cái có ở lớp cha thì ta dùng từ khóa SUPER

---