**Nhóm 02**

**Bài tập: Điểm giống nhau và khác nhau giữa class, abstract class và interface.**

**\* Giống nhau:**

- Đều có thể được kế thừa bởi các Class.

- Đều có thể bao gồm các phương thức. Abstract Class và Interface còn có thể bao gồm các phương thức trừu tượng.

- Đều là các khái niệm quan trọng trong lập trình hướng đối tượng, giúp định nghĩa hành vi và cấu trúc cho đối tượng.

**\* Khác nhau:**

**1. Class**

- Là một kiểu cấu trúc dữ liệu dùng để định nghĩa các đối tượng. Class chứa các thuộc tính và phương thức mà đối tượng có thể sử dụng.

- Có thể khởi tạo đối tượng trực tiếp từ Class

- Mỗi Class chỉ có thể kế thừa tối đa một Class cha. Lớp con sẽ kế thừa các thuộc tính và phương thức của lớp cha, đồng thời có thể ghi đè hoặc mở rộng chúng.

Ví dụ:

A computer code with text

Description automatically generated with medium confidence

A close-up of a computer code

Description automatically generated

Class Dog kế thừa trực tiếp Class Animal nên Dog cũng sẽ có thuộc tính weight và sử dụng được phương thức eat.

**2. Abstract Class**

- Là một Class không thể được khởi tạo trực tiếp.

- Có constructor nhưng chỉ sử dụng khi class con gọi super() để khởi tạo các thành phần của abstract class

- Không thể tạo đối tượng từ abtract class.

- Có thể có các thuộc tính và phương thức mà các lớp con sẽ kế thừa.

- Phương thức có thể là phương thức trừu tượng hoặc phương thức thông thường. Trong Abstract class thì các phương thức trừu tượng chỉ có khai báo, còn phương thức thông thường thì có thể được định nghĩa.

- Khi một lớp con kế thừa từ một lớp trừu tượng thì tất cả các phương thức trừu tượng của lớp cha phải được cài đặt cụ thể ở lớp con.

- Lớp cha của một lớp trừu tượng có thể không phải là một lớp trừu tượng.

Ví dụ:

A computer code with text

Description automatically generated

A screenshot of a computer code

Description automatically generated A screen shot of a computer code

Description automatically generated

- Dog và Bird đều kế thừa Abstract class Animal và phải tự định nghĩa phương thức trừu tượng Move

**3. Interface**

- Có vai trò tạo ra phương thức, hành vi chung cho các lớp, tương tự Abstract class.

- Interface không phải một Class nhưng có cấu trúc tương tự như một Class, tuy nhiên chỉ chứa các hằng số và phương thức trừu tượng.

- Không thể có constructor

- Một Class có thể kế thừa nhiều Interface.

A close up of text

Description automatically generated Ví dụ:

A close up of a text

Description automatically generated

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer code

Description automatically generated

- Dog và Bird đều kế thừa 2 Interface Soundale và Edible và tự định nghĩa hai phương thức makeSound và eat cho riêng mình.

**4. Tổng kết**

|  |  |  |  |
| --- | --- | --- | --- |
| Tiêu chí | Class | Abstract Class | Interface |
| Khởi tạo | Có thể khởi tạo trực tiếp | Không thể khởi tạo trực tiếp | Không thể khởi tạo trực tiếp |
| Phương thức trừu tượng | Không có | Có thể có một hoặc nhiều phương thức trừu tượng | Tất cả phương thức mặc định là trừu tượng |
| Đa kế thừa | Không hỗ trợ đa kế thừa | Không hỗ trợ đa kế thừa | Hỗ trợ đa kế thừa(một class có thể implements nhiều interface) |
| Thành phần | Có thể có biến instance, phương thức thông thường và phương thức static | Có thể có biến instance, phương thức thông thường và trừu tượng | Chỉ chứa các hằng số và phương thức trừu tượng |
| Phạm vi truy cập | Các phương thức và thuộc tính có thể để public, private, protected, default. | Các phương thức và thuộc tính có thể để public, private, protected, default. | Các phương thức mặc định là public và abstract. |
| Tỉnh mở rộng | Chỉ kế thừa từ một class | Chỉ kế thừa từ một class/abstract class | Có thể implements nhiều interface |