**CHƯƠNG 1: CÁC KIẾN THỨC CƠ BẢN**

1. Có 2 loại dữ liệu:

* Định sẵn – kiểu giá trị
* Đối tượng – kiểu tham khảo

1. Array jagged: là một mảng mà các phần tử là các mảng khác.
2. Phương pháp phân tích từ trên xuống: Phân rã class lớn thành n class con nhỏ hơn. Mỗi class nhỏ hơn lại phân rã thành m class nhỏ hơn nữa…
3. Namespace: Là một không gia tên theo dạng phân cấp.
4. Assambly: là phương tiện đóng gói vật lý nhiều phần tử.
5. Mỗi assembly có thể chứa nhiều phần tử nằm trong các namespace khác nhau. Ngược lại mỗi namespace có thể chứa nhiều phần tử mà về mặt vật lý chúng nằm trong assamly khác nhau

**CHƯƠNG 2 : CÁC KHÁI NIỆM CHÍNH CỦA LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG**

1. Chương trình = tập hợp các đối tượng độc lập, tương tác với nhau khi cần thiết để hoàn thành nhiệm vụ của chương trình.
2. Tính chất cơ bản của lập trình hướng đối tượng:

* Tính trừu tượng
* Tính đóng gói
* Tính đa hình
* Tính thừa kế

1. Đặc điểm:

* Chương trình chỉ chứa 1 loại thành phần : Đối tượng
* Các đối tượng có tính độc lập cao =>dễ nâng cấp bảo trì
* Không thể tạo ra dữ liệu toàn cục cho chương trình

1. Đối tượng: gồm 2 thành phần chủ yếu:

* Tác vụ
* Thuộc tính

1. Đặc tả một đối tượng bao gồm:

* Interface: góc nhìn sử dụng
* Class: góc nhìn hiện thực

1. Kiểu trừu tượng – Interface:

* Interface là tập hơp các điểm nhập mà bên ngoài có thể giao tiếp với đối tượng
* Biến đối tượng là biến tham khảo => không chứa trực tiếp đối tượng, chỉ chứa thong tin để truy xuất đối tượng
* Biến đối tượng kiểu interface có thể tham khảo đến nhiều class miễn sao nó hỗ trợ interface.

1. Class: định nghĩa chi tiết hiện thực đối tượng

* Thuộc tính
* Tác vụ

1. Tính bao đóng: Che giấu mọi chi tiết hiện thực đối tượng. Cung cấp các từ khóa private, protected, internal, public để xác định tầm vưc truy xuất.
2. Tính thừa kế:

* Cho phép đa thừa kế đối với Interface, nhưng chỉ đơn thừa kế đối với class
* Đối tượng của class con >= đối tượng class cha
* Đối tượng của class con hoàn toàn có thể đóng vai trò của đối tượng class cha nhưng ko ngược lại.

1. Tính bao gộp: Một đối tượng có thể chứa nhiều đối tượng khác.
2. Thông điệp : là phương thức giao tiếp duy nhất giữa các đối tượng. Gồm 3 thành phần:

* Tham khảo đến tên đối tượng
* Tên tác vụ muốn gọi
* Danh sách tham số (truyền hoặc nhận)

1. Tính đa hình: cùng 1 lệnh gửi thông điệp đến đối tượng thông qua cùng 1 tham khảo, nhưng với từng thời điểm thực thi sẽ kích hoạt từng tác vụ khác nhau ở các đối tượng khác nhau.
2. Tổng quát hóa: Sinh tự động các class có cấu trúc giống nhau.

**CHƯƠNG 3: INTERFACE VÀ CLASS**

1. Định nghĩa một class:

***Class MyClass : BaseClass, I1, I2, I3***

* BaseClass: class cha thừa kế
* I1,I2,I3: Các interface thừa kế

1. Một class bao gồm:

* Thuộc tính vật lý của đối tượng
* Tác vụ chức năng, các toán tử chức năng.
* Các thuộc tính giao tiếp luận lý
* Các đại diện hàm chức năng
* Các sự kiện
* Indexer của class

1. Thành phần statistic: Thành phần kết hợp với class không bị nhân bản cho từng đối tượng.

**CHƯƠNG 4: VÒNG ĐỜI ĐỐI TƯỢNG VÀ SỰ TƯƠNG TÁC GIỮA CHÚNG**

1. Constructor: Mỗi lần đối tượng đc tạo ra (new) máy sẽ tự động constructor của class tương ứng.
2. Mặc định khi gọi constructor của cha chạy nó sẽ gọi constructor không có tham số.
3. Destructor:

* VC không cung cấp hàm delete đối tượng khi không cần dùng nữa
* Việc xóa đối tượng được hệ thống tự động làm bằng Garbage collection (trình thu dọn rác).

1. Trong VC nếu sử dụng từ khóa **vitual** trong khi định nghĩa tác vụ thì tác vụ này sẽ được xử lý theo cơ chế liên kết động.
2. Constructor và destructor là 2 tác vụ đặc biệt chỉ được gọi bởi hệ thống. Nên ta không thể dung từ khóa vitual cho nó.

**CHƯƠNG 5: XÂY DỰNG GIAO DIỆN**

1. Class cấu thành chươn trình dùng giao diện đồ họa được chia làm 2 nhóm.

* Các class miêu tả các đối tượng giao diện
* Các đối tượng miêu tả chức năng

1. Mọi phần tử giao diện đều là các cửa sổ. Một ứng dụng có thể có nhiều cửa sổ, nhưng trong 1 thời điểm chỉ có một số ít cửa số hiển thị để làm việc với người dùng.
2. Tính chất của đôi tượng giao diện:

* Có thuộc tính giống như đối tượng bình thường.
* Name: giá trị của thuộc tính này sẽ trở thành biến tham khảo đến đối tượng.
* Các thuộc tính xác định vị trí, kích thước.
* Các thuộc tính xác định tính chất hiển thị
* Các thuộc tính xác định hành vi
* Các thuộc tính liên kết dữ liệu database: data binding

1. Quy trình viết ứng dụng chạy bằng VC:

* Nắm bắt yêu cầu => xác định chức năng
* Phân tích chức năng => xậy dựng class tương ứng
* Thiết kế chi tiết class: Phân loại class thành 2 loại : Giao diện – Giải thuật.
* Hiện thực: thiết kế trực quan – tạo hàm xử lý sự kiện.

CHƯƠNG 6: TƯƠNG TÁC VỚI NGƯỜI DÙNG

1. Sự tương tác gồm 2 hoạt động chính:

* Chờ dư liệu hoặc chờ nhận lệnh của người dung
* Hiển thị kết quả

CHƯƠNG 7: GHI ĐỌC DỮ LIỆU

1. VC cung cấp 3 đối tượng để phục vụ việc ghi đọc file:

* FileStream
* BinaryWrite
* BinaryReader

1. Tác vụ BinaryWriter có 14 biến thê 1 tham số để ghi đc 14 kiểu dữ liệu định sẵn.
2. Để hỗ trợ việc đọc ghi nội dung của đối tượng VC đề nghị sửa dụng kỹ thuật serialization.

CHƯƠNG 8: XÂY DỰNG CLASS TỔNG QUÁT HÓA

1. Tổng quát hóa nhằm mục đích tối thiểu hóa công việc của người lập trình.

CHƯƠNG 9: TẠO ĐỐI TƯỢNG CÁ NHÂN HÓA

1. Gồm 3 loại phổ biến:

* User Control
* Inherited Control: gần giống control có sẵn
* Owner-drawn Control: giống y như control có sẵn nhưng giao diện thì khác. Thừa kế class có sẵn và toàn bộ chức năng rồi override tác vụ OnPaint để vẽ lại bộ mặt mới.

CHƯƠNG 10: TRUY XUẤT DATABASE