# Unity Tricks

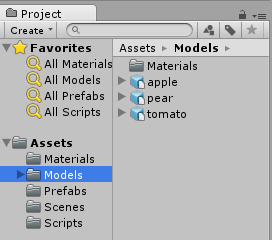
## Design scene

### Layout

Trước khi làm việc cần lựa chọn layout cho phù hợp với công việc sắp tới sẽ làm. Ví dụ để tạo không gian 3d nhiều thành phần con nên chọn layout “Tall”. Kiểu này có không gian làm việc với scene rộng. Còn nếu thiết kế giao diện thì nên chọn “2 by 3”, kiểu này cho phép vừa làm vừa xem kết quả hiển thị trên màn hình thực sẽ như thế nào.

### Folder tree

Đối với tài nguyên trong game cần tạo riêng từng thư mục để quản lý. Ví dụ:



Cần thống nhất về cách chia thư mục. Ở đây chia theo kiểu tài nguyên. Các đoạn mã scripts thì phải đưa vào thư mục scripts…

### Hierarchy

Các tài nguyên khi được đưa vào scene cũng cần phải sắp xếp gọn để dễ quản lý. Unity hỗ trợ lựa chọn đối tượng bằng cách click trực tiếp trên scene nhưng khi các đối tượng chồng chéo lên nhau và có những đối tượng “invisible” thì việc sắp xếp và lựa chọn đối tượng trên hierarchy là cần thiết.

Xây dựng scene bình thường, trước khi số lượng object trở nên quá nhiều cần quyết định phương án gộp các object tương tự nhau. Ví dụ có thể chia thành kiểu object tĩnh (không có scripts) và object động. Hoặc có thể xếp những object nằm cùng vị trí trên scene lại thành một. Chủ yếu là để có thể lựa chọn nhanh chóng trên hierarchy.

### Prefab

Việc tạo prefab không chỉ cần thiết cho coding mà nó còn rất có ích trong việc design scene. Khi thiết kế scene có nhiều những object giống hệt nhau (duplicate) thì việc chọn object đó rồi duplicate ra là rất nguy hiểm. Vì khi duplicate thì 2 object đó hoàn toàn khác nhau. Khi có thay đổi thì sẽ phải thay đổi từng object đó, rất mất thời gian. Cách để giải quyết là tạo prefab cho object cần duplicate nhiều lần.

Việc sử dụng prefab dùng chung thuộc tính xem tại: <http://docs.unity3d.com/Manual/Prefabs.html>

### Transform

Hạn chế việc kéo thả, scale bằng tay. Việc bằng tay chỉ làm ước lượng khi có cảm giác hợp lý rồi thì nên chỉnh transform bên inspector. Nghĩa là sau khi chỉnh tới kích thước ok thì nhìn sang bên inspector làm tròn các thông số bên đó. Việc này có lợi cho coding sau này khi các con số đều là số nguyên.

### GUI

Khi sắp xếp các thành phần GUI trên giao diện, cần chú ý đến tọa độ của các thành phần, tất cả đều xoay quanh gốc tọa độ (0, 0). Nếu cần thiết kế đặt 2 nút song song với nhau, thì khi căn chỉnh được 1 nút rồi, thì đổi dấu của tọa độ nút thứ nhất sẽ được nút thứ 2 song song.

Khi gộp nhóm. Trước hết nên đưa object parent về tọa độ (0,0). Và các tỉ lệ scale là (1,1,1).

### Scene

Hạn chế việc tạo nhiều scene trong object. Chia các scene khác nhau có thể dễ hơn cho việc design nhưng lúc code sẽ vất vả hơn, vì lúc chuyển scene, các object cũ sẽ bị xóa hết. Ta lại phải tính đến việc bảo lưu các giá trị khi đổi scene.

Chỉ thiết kế các scene khác nhau khi không còn cách nào khác.

### Keyboard shortcut

Sử dụng keyboard shortcut rút ngắn thời gian lựa chọn tool bằng chuột, thao tác sẽ nhanh gọn hơn. Các phím tắt thường dùng trong design.

|  |  |
| --- | --- |
| **Phím** | **Chức năng** |
| Ctrl+Shift+N | Tạo một object mới |
| Alt+Shift+N | Tạo object mới là con của object đang chọn |
| Alt+Chuột trái | Xoay hướng nhìn |
| Alt+Chuột phải | Zoom view |
| Alt+Shift+Chuột phải | Zoom với tốc độ nhanh hơn |
| F | Zoom vào object đang được chọn |
| Q-W-E-R | Tương ứng View-Translate-Rotate-Scale |
|  |  |

## Coding

### GameManager.cs

Trong phần script luôn cần một script tên “GameManager.cs”.

Scripts này không đưa vào object nào, nó chỉ chứa các thành phần tĩnh (static) là các thông số của game như level, score…

Các biesn trong GameManager không được để public. Tất cả biến đều phải viết hàm get,set. Việc sử dụng hàm get,set sẽ dễ dàng debug khi có lỗi logic và dễ thêm các chức năng khác liên quan.

### Public variable

Các biến để public trong code có thể được chỉnh sửa trực tiếp trên inspector. Khi để các biến public cần chú ý khi thay đổi giá trị bằng tay trên code, Unity sẽ ưu tiên giá trị được định nghĩa trong inspector chứ không phải trong code. Nên các biến giá trị (integer, float) nếu chưa cần thiết thì hạ chế để public. Vì khi sửa một số sẽ phải trở ra giao diện Unity để sửa trong code, việc này đôi khi gây nhầm lẫn.