게임 개발 기초 상식 | 2D 플랫폼의 이해

1. 2D 세계의 구성

Bitmap 과 Vector

**Bitmap**

* 화면 표현 방법을 점(pixel)마다 색상(aRGB)정보를 사용하는 방식
* Jpg, gif, png, bmp, dds 등 2D 게임에서 대부분 사용
* 확대 축소시 이미지가 뭉개진다.
* 앨리어스 현상(계단현상)이 발생한다.

**Vector**

* 화면 표현 방법을 좌표(Point)와 좌표 간의 연결을 **연산**하여 사용하는 방식
* 확대 축소가 자유롭고 용량의 변화가 없다는 점이 장점
* 연산부하가 발생한다.
* 리소스 제작이 어렵다.

**색 표시 방법**

* RGB~ 각각 8bit 총 24bit 1600만 가지의 색 / 투명색 없음
* aRGB~ 각각 8bit 총 32bit / 투명색 존재

1. 레이어

레이어란? | 2D 이미지의 층

레이어 구성의 목적

* UI표시, 게임 세계의 구성, 이펙트, 이미지 조합을 통한 재활용 및 용량 절약

1. 레이어 분해 예시

레이어 분해 예시

* 고전 게임 Street Fighter 2의 Ken 스테이지를 분해

1. KO UI
2. 노란색 UI
3. 빨간색 UI 및 점수 UI 및 텍스트들
4. 정박용 못
5. 캐릭터 + 파괴가능 오브젝트
6. 바닥면
7. 큰배 + 관객
8. 성조기
9. 바다, 작은배, 산, 하늘
10. 레이어는 대략 다음과 같이 구성됩니다.

* 원경
* 근경
* 땅
* 구조물 등 고정적 오브젝트
* 캐릭터/몬스터/아이템 등 유동적 오브젝트
* 각종 이펙트 효과
* 날씨나 시간 등의 표시
* 커서
* UI

1. 레이어는 동적으로 바뀌기도 합니다

* 특정 상황에서만 생성되거나 사라지기도 하고
* 뒤에 있는 캐릭터가 앞으로 나오거나 앞에 있는 캐릭터가 뒤로 가거나 하는 것처럼 레이어의 순서가 바뀌기도 하고
* 특정 상황에서 반투명, 흑백 등으로 처리되기도 함

1. 맵 / 배경

실제 게임플레이가 이루어지는 배경

* RPG, RTS, 시뮬레이션 등 많은 장르에서 효율성을 이유로 **타일 맵**을 많이 사용
* 대개 단위 타일은 게임 플레이를 구성하는 최소 단위의 오브젝트들이 위치
* 타일 이미지와 속성이 1:1로 대응되는 게임도 있지만 이미지와 속성을 별개로 설정하는 게임도 있다.

**타일이 가지고 있는 정보** | 지형 속성, 이동 가능, 높이, 제어 가능 여부 등

* 타일을 점유한 오브젝트는 상태, 방향, 충돌 등을 체크

2D 오브젝트의 구성

1. 스프라이트

**이미지 출력 전반을 관리하는 클래스** | 각 프레임 별로 정보가 들어감

게임 내 모든 애니메이션이 되는 오브젝트의 효과적인 관리

이미지의 재 사용성을 높여 용량과 메모리의 관리를 용이하게 함

1. 스프라이트에서 관리하는 데이터

**이미지** | 캐릭터, 장비, 이펙트, 상태 아이콘 등

* 캐릭터를 기준으로 장비, 이펙트 등의 추가 이미지가 붙는 좌표를 지정

**기준점** | 시점에 따라 기준점 설정

* 예) 탑뷰 : 좌상, 쿼터뷰 : 아래 중앙, 사이드뷰 : 우하 등을 참고점으로 한 임의의 지점

**애니메이션** | 이미지 조합 및 제어(반전, 회전, 확대축소, 색 변환 등)을 통해 데이터 최소화

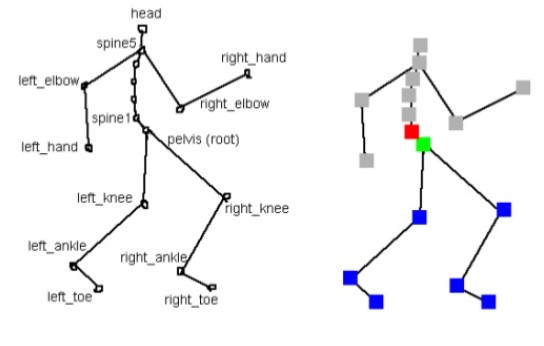
* 출력순서, 딜레이, 반복 구간 설정 등을 통해 다양한 애니메이션 표현

**충돌박스** | 공격을 받는 충돌 박스 or 공격을 하는 충돌 박스의 설정

1. Skeletal Animation

계층 구조를 가진 상호 연결된 뼈대(Skeleton or rig)를 이용해 애니메이션을 하는 기법

* 뼈대가 연결된 점은 **위치**와 **방향 정보**를 가짐
* 프레임 단위로 연결점의 정보를 저장하기도 하지만 대개 물리 엔진을 이용해 **키 프레임** 간의 전환을 함으로써 데이터 용량도 절약하고 다양한 상황도 만들어 냄



1. 팔레트

초등학교 미술 시간 때 썼던 팔레트 맞음

* 팔레트 색상 변환을 통해 이미지의 재 사용성을 높이는 기법
* RTS나 스포츠 게임의 팀 색깔, 레벨이 다른 몬스터 등에서 활용

1. 아바타

아바타 시스템은 기본적으로 레이어, 스프라이트, 팔레트 시스템의 총합