

[01강] RPG Maker VX Ace의 시작 (쓰꾸르 게임 만들기) - RPG Maker VX Ace 강좌

시작하기

2014/10/28 00:26

<http://blog.naver.com/biud436/220163747276>

RPG Maker VX Ace 왕초보탈출 1

"전문 지식이 없어도 누구나 쉽게 게임을 만들 수 있습니다"

강의 목차

[01강. RPG Maker VX Ace의 시작](#)

[02강. 맵과 오토타일](#)

[03강. 이벤트의 기본 조건](#)

[04강. 스위치와 셀프 스위치](#)

[05강. 변수와 조건분기](#)

[06강. 비밀 번호의 처리](#)

[07강. 게이지바 만들기](#)

[08강. 기본 전투의 처리 1편 \(기초편\)](#)

[09강. 기본 전투의 처리 2편 \(응용편\)](#)

[10강. 기본 전투의 처리 3편 \(심화편\)](#)

[11강. 퀘스트 만들기](#)

[12강. 데이터 베이스 - 애니메이션, 프레임 보완 기능, 플래시, 커먼 이벤트, 용어, 시스템 및 각종 옵션](#)

[13강. 리소스 : 스크립트 적용하기, 새로운 소재 불러오기 & 내보내기](#)

[14강. 스크립트의 기본 : 기본 문법과 스크립트의 활용법](#)

[15강. 게임 배포 - 사용한 소스에 대한 크레딧 만들기, 게임을 본격적으로 배포하기](#)

쭈꾸르 툴이란?

RPG Maker VX Ace 는 지난 2011년 12월 15일에 출시되었으며 게임 제작을 할 수 있는 일본 엔터브레인 사의 게임 제작 툴로 쭈꾸르(만들기 + 툴의 조합)라고도 부릅니다. 이 프로그램은 RPG를 만들 수 있는 프로그램이기 때문에 RPG와 관련된 리소스들과 기본 시스템 스크립트가 포함되어있으며 전문 지식이 없어도 누구나 쉽게 게임을 제작할 수 있다는게 큰 특징입니다.



▲ 쭈꾸르 툴 영문판의 모습

게임 엔진만 제공되는 제작 툴들은 게임을 만들려면 사용자가 직접 리소스를 만들거나 저작권법에 위반되지 않는 리소스를 찾아 나서는 작업부터 먼저 진행해야 합니다. 하지만 RPG 만들기 툴은 기본적으로 사용할 수 있는 리소스가 바로 제공되기 때문에 별다른 준비없이 바로 원하는 게임을 만들 수 있습니다. 그러나 멀티플랫폼 배포 기능 1을 제공하고 있지 않기 때문에 요즘 같은 모바일 시대에 뒤떨어지는 느낌이 있긴 합니다. 하지만 그 어떤 툴보다 쉽게 게임을 제작할 수 있다는 명제는 아직도 유효합니다.

- 맵 에디터 기능

타일 기반의 맵을 편집할 수 있는 강력한 맵 에디터를 제공합니다.

- 이벤트 커맨드 기능

이벤트 커맨드를 활용하면 전문 게임 프로그래밍 지식 2 이 없어도 게임 컨텐츠를 쉽게 만들 수 있습니다.

- 데이터베이스 기능

게임 데이터 파일을 편집할 수 있는 에디터입니다. (게임 디렉토리 내에 rvdata2 라는 확장자를 가진 파일을 만들어냅니다)

- 애니메이션 기능

키 프레임 마법 애니메이션을 제작할 수 있는 애니메이션 제작 도구입니다.

- 스크립팅 기능

XP 때 새로 추가된 게임 스크립팅 시스템이며 스크립트 언어를 사용하면 더 정교한 게임 제작을 할 수 있습니다.

- DLL3 확장 기능

스크립트 기능을 통해 DLL 파일로 만들어진 함수도 호출할 수 있습니다.

관련 강좌

[\[23강\] C++로 RPG Maker VX Ace 확장 DLL 만들기](#)

- 리소스 관리 기능

리소스(소재)를 불러오고 내보낼 수 있는 기능입니다.

- 테스트 플레이 기능

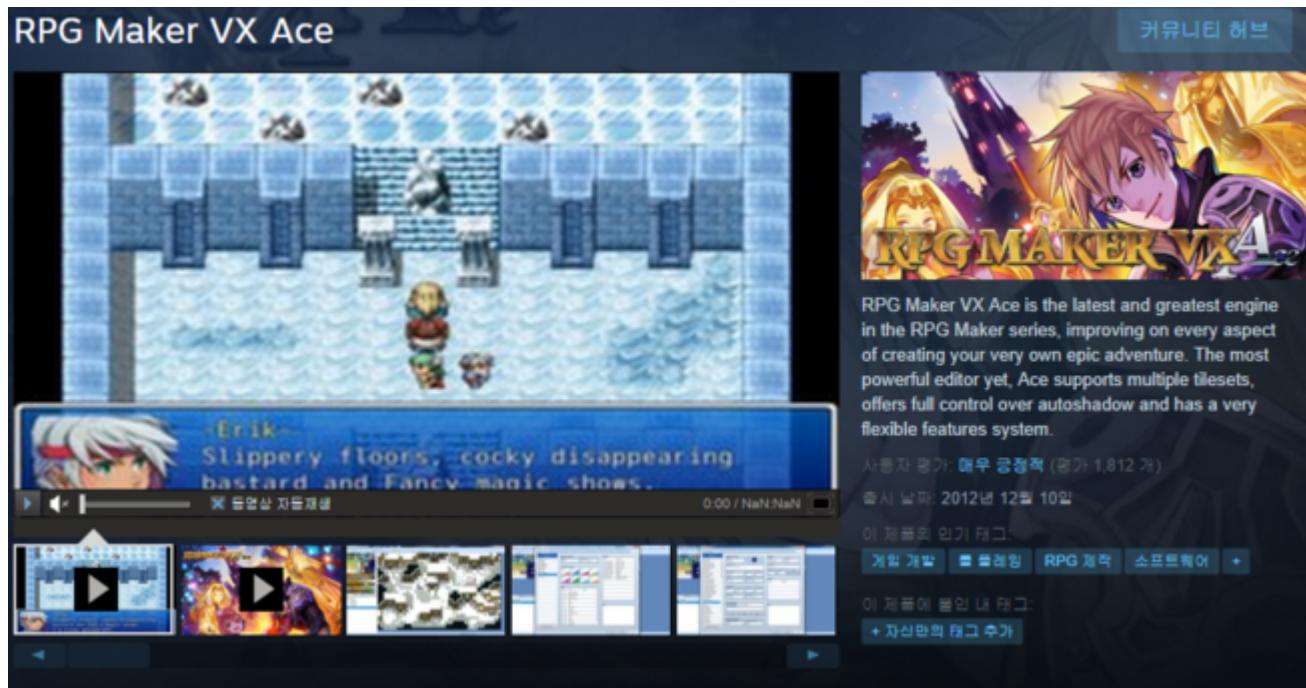
컴파일 작업 없이 게임을 바로 테스트 할 수 있습니다.

- PC 게임 배포 기능

PC 플랫폼에서 실행할 수 있는 설치 파일(일종의 압축 파일)을 만드는 기능입니다. 설치 파일을 실행하면 압축이 해제되고 게임 암호화 데이터가 들어있는 폴더가 만들어집니다.

정품 구매 경로 안내

RPG Maker VX Ace는 공식스토어와 스팀에서 판매되고 있으며 세일된 가격은 \$17.49입니다. 지원 가능한 언어에 포함되어있지 않은 우리나라는 영문판으로 플레이가 가능합니다. 스팀에서 자주 세일이 되고 있습니다. [RPG Maker Web이라는 공식 포럼의 페이스북 페이지](#)를 구독하면 세일 정보를 받아보실 수도 있습니다. 구매는 해외 결제가 가능한 비자 체크 카드 등이 있으면 쉽게 결제가 가능하며 디지털 다운로드 방식으로 컴퓨터에 다운 받을 수 있습니다. 이후에는 스팀 클라이언트를 통해 툴을 실행할 수 있게 됩니다.

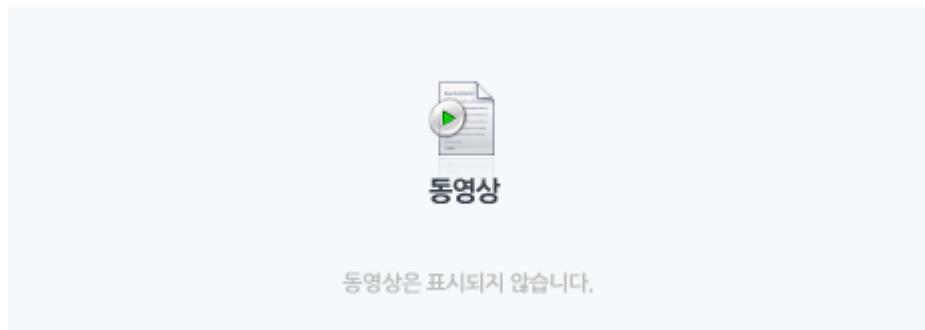


▲ 스팀

해외 결제가 가능한 결제 수단(마스터 체크 카드, 비자 체크 카드)이 있으면 스팀에서 정품을 구매할 수 있습니다. 세일 기간에 구매한다면 RPG Maker VX Ace를 -75% 세일된 가격 \$17.49에 구매할 수 있습니다.



해외 결제가 가능한 결제 수단이 있으면 스팀에서 정품을 구매할 수 있습니다.



- [RPG Maker VX Ace](#)
 - 판매처 : 스팀, 알피지 메이커 웹, 아마존, 험블번들, 번들스타, GMG, 누빔, GameStop, [데기게임즈](#)
 - 정가 : \$69.99
 - 발매일 : 2011년 12월 15일
 - 언어 : 영어, 프랑스어, 스페인어, 독일어, 이탈리아어, 포르투갈어(브라질)
 - 지원 OS : Microsoft® Windows® XP / Vista / 7 (32-bit/64-bit)
 - 루비 버전 : RGSS3 (Ruby 1.9.2)
- [RPG 츠쿠르 VX Value! + 일본풍 소재집](#)
 - 판매처 : 알피지 메이커 웹, 데기카
 - 정가 : 7,344엔 → 6,156엔
 - 발매일 : 2012년 09월 27일 (Value!+)
 - 언어 : 일본어
 - 루비 버전 : RGSS2 (Ruby 1.8.1)
- [RPG Maker VX](#)
 - 판매처 : 알피지 메이커 웹, 데기카
 - 정가 : \$59.99
 - 발매일 : 2007년 12월 27일
 - 언어 : 영어
 - 루비 버전 : RGSS2 (Ruby 1.8.1)



동영상

동영상은 표시되지 않습니다.

- [RPG 츠쿠르 XP VALUE!+](#)

- 판매처 : 데기카, 아마존 일본, 엔터브레인 스토어
- 정가 : 4,104엔 → 3,024엔
- 출시일 : 2012년 09월 27일 (Value!+)
- 언어 : 일본어
- 지원 OS : Microsoft® Windows® XP/Vista/7 64-bit OS is NOT supported
- 루비 버전 : RGSS1 (Ruby 1.8.1)

- [RPG Maker XP](#)

- 판매처 : [스팀](#), 알피지 메이커 웹, [아마존](#), 험블번들, 번들스타, [GMG](#), 데기카, [데기게임즈](#)
- 정가 : \$24.99
- 출시일 : 2004년 07월 22일(일본어)
2005년 09월 16일(영어)
- 언어 : 영어
- 지원 OS : Microsoft® Windows® XP/Vista/7 64-bit OS is NOT supported
- 루비 버전 : RGSS1 (Ruby 1.8.1)



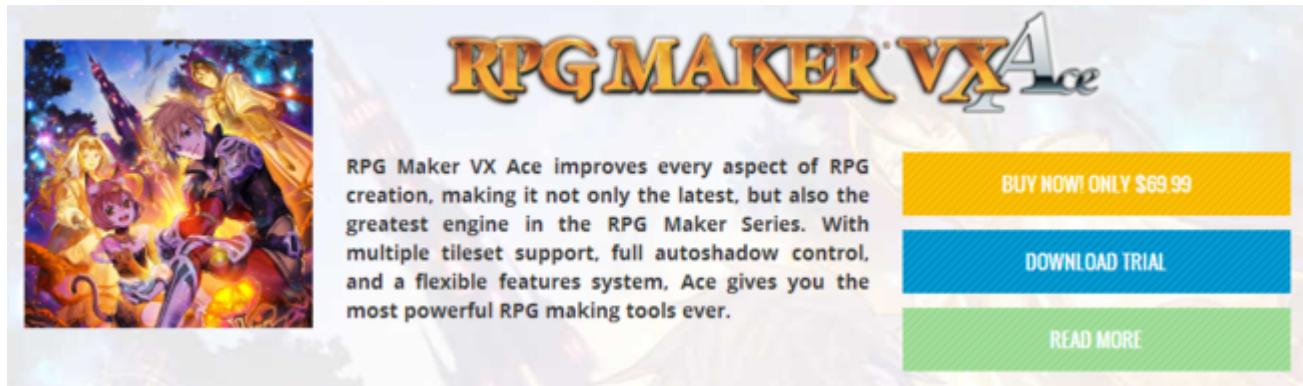
동영상

동영상은 표시되지 않습니다.

- [RPG Maker 2003](#)

- 판매처 : [스팀](#), [데기게임즈](#)
- 판매 가격 : \$19.99

- 출시일 : 2002년 12월 18일 (일본어)
2015년 04월 24일 (영어)
- 언어 : 영어
- 지원 OS : Microsoft® Windows® XP/Vista/7/8
- [RPG Maker 2000 Value!+](#)
 - 판매처 : 데기카, 아마존 일본, 엔터브레인 스토어
 - 정가 : 4,101엔 → 3,024엔
 - 출시일 : 2000년 4월 5일 (Original)
2012년 09월 27일 (Value!+)
 - 언어 : 일본어
 - 지원 OS : Microsoft ® Windows ® XP/Vista/7/8 일본어 (32bit/64bit 버전 OS 모두 지원)
-



▲ [RPG Maker Web 공식 스토어](#)

무료로 배포되고 있는 RPG Maker VX Ace Lite 버전의 설치

정품은 정가로도 구매할 수 있으나 \$69.99라는 가격에 부담이 될 수도 있습니다. 또한 한시적으로 할인이 되기 때문에 기간이 맞지 않으면 무작정 기다려야만 합니다. 그러나 RPG Maker VX Ace Lite라는 무료 버전도 존재합니다. Lite 버전은 위에서 소개한 공식 스토어와 스팀에서 다운로드 받을 수 있으며 기능에 제한이 있고 기간 제한은 없습니다. 기본 스트립트 에디터를 사용하지 못하게 막아놓았고 스크립트를 사용할 수 있는 부분이 굉장히 제한적입니다. 또한 데이터베이스 최대 갯수의 제한, 이벤트 생성 갯수의 제한 등이 있습니다. 물론 스크립트도 일종의 파일이기 때문에 외부에서 스크립트를 해석하여 조작할 수 있는 에디터로 스크립트를 조작할 수도 있고 한계를 돌파할 수도 있습니다. 하지만 이벤트 갯수에도 제한이 있기 때문에 게임 제작에 매우 불편함이 생길 것입니다. 정품을 구매하기 전까지 맛보기로 사용하기에는 안성맞춤입니다.



[RPG Maker VX Ace Lite 다운로드](#)

- RPG Maker VX Ace Lite
 - 다운로드 링크1 [[스팀](#)]
 - 다운로드 링크2 [[알피지 메이커 웹](#)]
 - Lite 버전은 기능이 제한되어있는 무료 버전입니다
 - 별도의 인증 후 사용할 수 있습니다.
 - 사용 기간은 제한이 없습니다
 - Lite 버전에서는 데이터베이스의 최대치를 늘릴 수 없습니다⁴
 - 스크립트 에디터 5를 사용할 수 없습니다.
 - 맵 하나에 커먼 이벤트와 이벤트를 10개까지만 만들 수 있습니다
 - 4개의 타일셋만 사용할 수 있습니다
 - 맵을 20개까지만 생성할 수 있습니다.

RPG Maker VX Ace 툴의 한글화

RPG Maker VX Ace는 공식적으로 한국어를 지원하지 않습니다. 그러나 스팀 버전은 전 세계에 소프트웨어를 판매하기 위한 틀이므로 번역을 손 쉽게 할 수 있도록 문자열 요소가 모두 DLL에 분리되어있습니다. 6개 국어(영어, 프랑스어, 스페인어, 독일어, 이탈리아어, 포르투갈어)는 개발사에서 공식 지원하고 있으나 한국어는 지원 대상이 아니기 때문에 패치가 필요할 수 있습니다. 비공식 패치 파일은 [이 글](#)을 통해 스팀 버전 패치 파일을 적용하면 패치가 가능합니다. 하지만 비공식 이기 때문에 업데이트에 취약하며 여러가지 문제가 생길 수 있으니 되도록이면 원문판을 그대로 쓰시기 바랍니다.

루비 스크립트 에디터

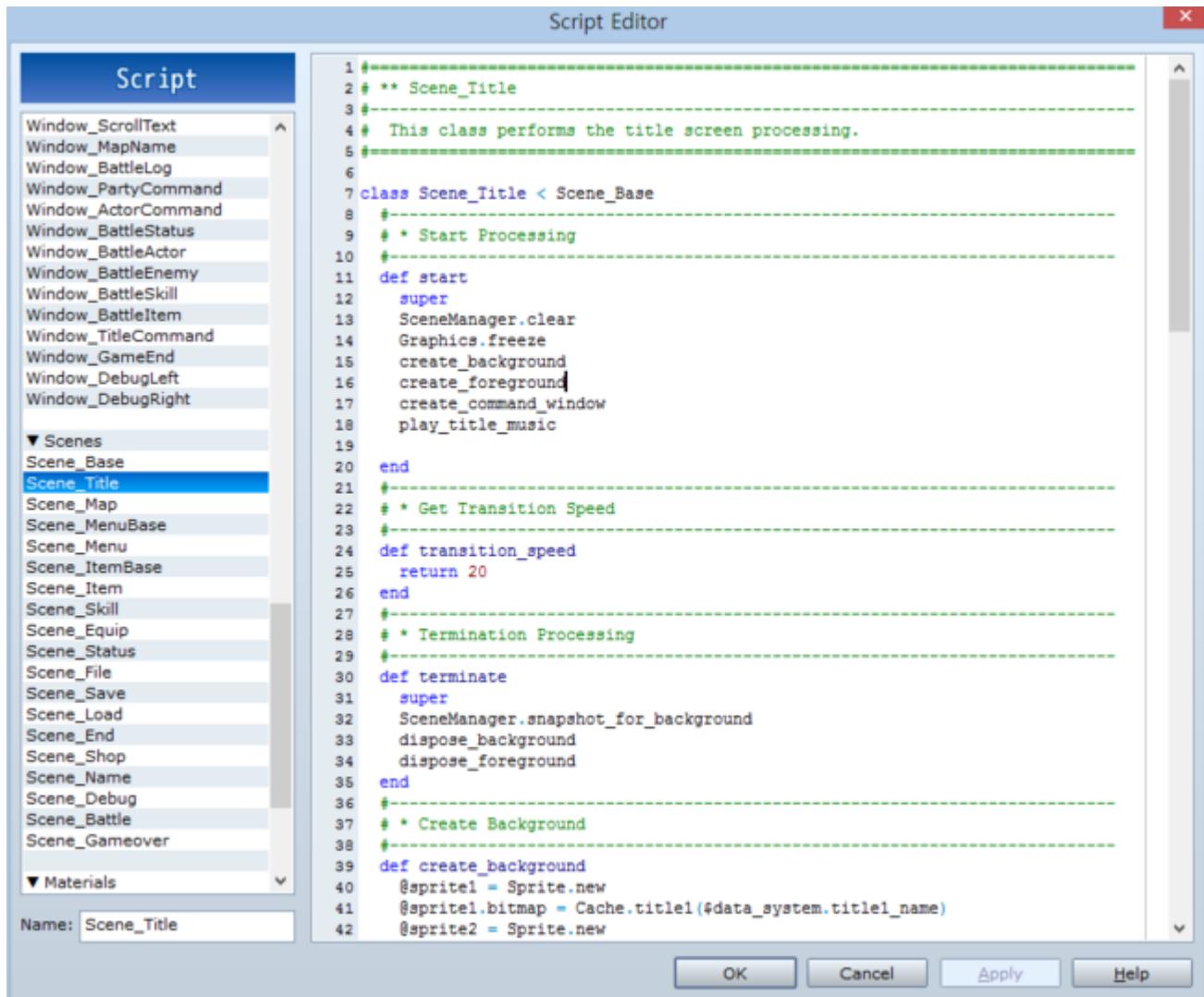
RPG Maker VX Ace에는 Ruby 스크립트를 작성할 수 있는 스크립트 에디터가 포함되어있습니다. RPG Maker VX Ace의 기본 게임 규칙들은 모두 Ruby 스크립트로 작성되어 있습니다. 따라서 Ruby 스크립트를 다룰 수 있다면 고급 제작자가 될 수도 있습니다.

Ruby는 일본인 프로그래머 Yukihiro Matsumoto 가 1993년도에 창시한 인터프리터 언어입니다. 문법은 파이썬이라는 언어와 비슷합니다. 루비는 루아 스크립트와 같이 C언어로 작성되어 있는 인터프리터 언어입니다. 인터프리터 언어는 컴파일을 거치지 않고 스크립트를 해석하여 실행하는 인터프리터만 포함되어있으면 구동이 가능합니다.

Ruby는 오픈소스이기 때문에 다른 소프트웨어에 포함하기 쉽고 라이브러리 DLL 를 만들어서 스크립트 언어로 활용할 수도 있습니다. RPG Maker VX Ace 도 그러한 형태로 라이브러리 파일이 제작되어있으며 루비 라이브러리가 정의되어있는 DLL에서 스크립트를 불러와서 처리합니다.

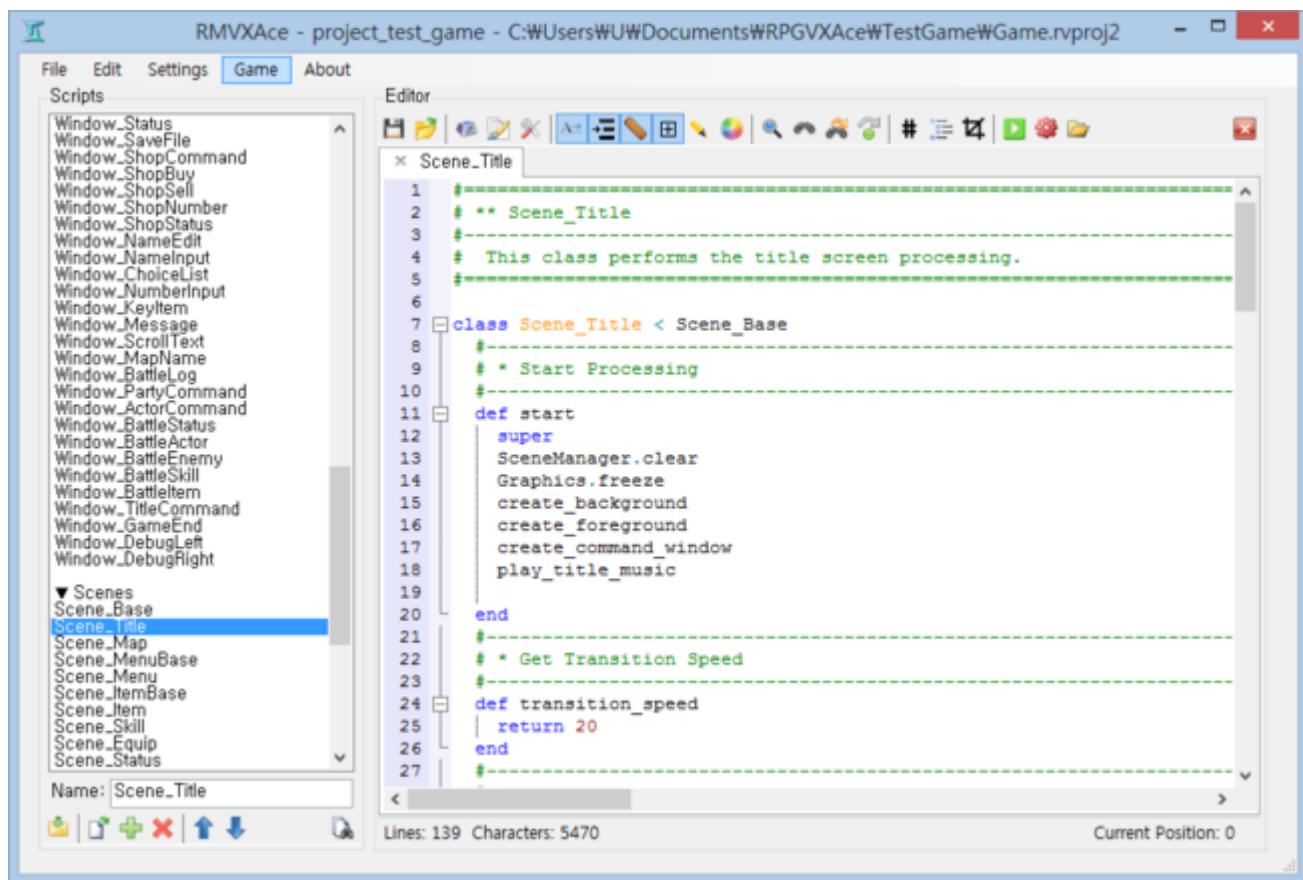
RPG Maker VX Ace에 포함된 루비의 버전은 1.9.2 버전입니다. 이 버전은 전작에 포함된 XP와 VX에 포함되어있는 1.8.1 버전보다 높은 버전입니다. 문법적으로 달라진 부분도 있지만 크게 달라지진 않았으며 포함되어있는 라이브러리도

같습니다. 그러나 툴마다 골격 구조가 달라 호환이 되진 않습니다. RPG Maker VX Ace는 그래픽, 오디오, 키보드 입력을 처리하는 모듈을 포함하고 있습니다. 이 모듈은 스크립트의 골격 구조만 익히면 매우 간단하게 사용할 수 있습니다. 모든 스크립트는 Data 폴더의 Script.rvdata2에 저장되어 있습니다. Game.exe에서 RGSS301.DLL에 포함된 루비 인터프리터를 통해 작성된 스크립트들이 해석되며 실행됩니다.



▲ 스크립트 에디터, 많은 부분이 이미 작성되어 있어서 게임 제작에 큰 도움이 됩니다

RPG Maker VX Ace에 포함된 기본 스크립트 에디터에는 함수 자동 완성 등의 인텔리센스 기능이 없기 때문에 불편한 느낌이 있어서 VXA 스크립트들을 편집할 수 있는 Gemini Editor라는 툴도 나와 있으니 참고하시기 바랍니다. 이 툴은 [여기서](#)에서 내려받을 수 있습니다.



▲ 제미니 스크립트 에디터, 자동 완성 기능을 사용할 수 있습니다(C#으로 제작됨)

소재 관리와 캐릭터 제작기

기본적으로 데이터베이스와 소재 관리창 그리고 스크립트 에디터가 포함되어있고 소재 관리를 통해 스팀에 출시된 공식 DLC들을 불러올 수 있습니다. 추가적으로 페이스칩과 캐릭터 그래픽을 만들 수 있는 캐릭터 제작기도 포함되어있습니다. 캐릭터 제작기는 미리 등록되어있는 칩셋을 통해 새로운 캐릭터들을 디자인 할 수 있는 기능이 있습니다. 페이스칩과 캐릭터 칩셋을 누구나 쉽게 제작할 수 있게 되어있습니다.

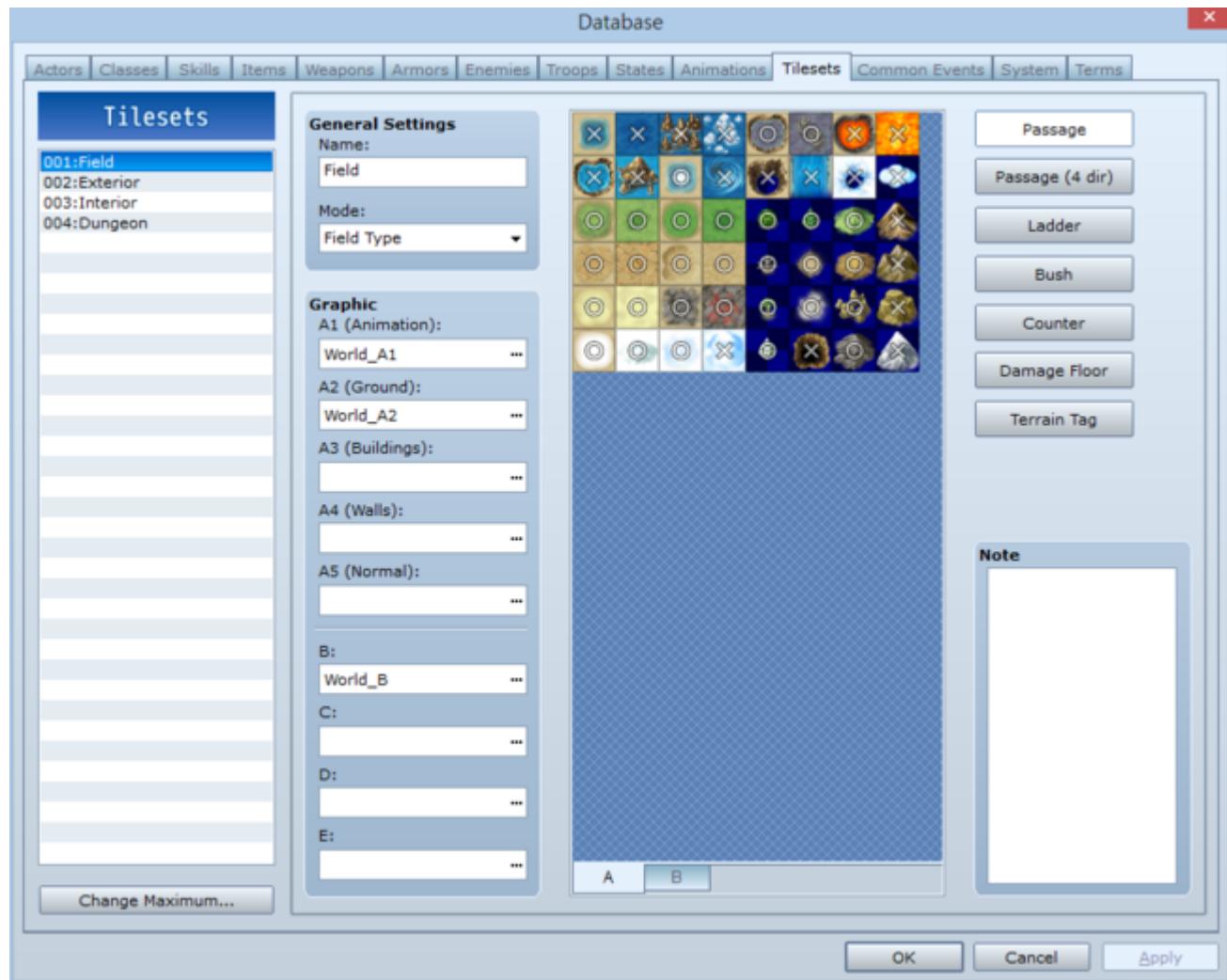
소재의 위치	소재의 이름	소재 규격	비고
Graphics/Characters	Characters	384 X 192	! : 문 (4방향 4프레임) \$: 특수 캐릭터, 이펙트, 큰 캐릭터를 사용할 때
Graphics/Faces	Face Graphics	384 X 192	96 X 96 4열 2행
Graphics/Battlers	Battle Graphics	544 X 296	
Graphics/Animations	Animations	192 X 192	
Graphics/Tilesets	A1	512 X 384	바다, 용암

	A2	512 X 384	일반 필드	
	A3	512 X 256	건물 오토타일	
	A4	512 X 480	벽 오토타일	
	A5	256 X 512	계단	8열 X 16행
	B, C, D, E	512 X 512	조형물	16열 X 16행
Graphics/Battlebacks1				
Graphics/Battlebacks2	Battle Backgrounds	580 X 444		
Graphics/Parallaxes	Parallaxes	제한 없음		
Graphics/Titles1,	Titles	544 X 416		
Graphics/Titles2				
Graphics/System	Balloon.png	256 X 320	8 패턴 X 10타입	8패턴 X 10타입
	IconSet.png	384 X 936	16열 X 16행(++)	24 X 24
	BattleStart.png	544 X 416	PNG	8Bit, GrayScale
	Shadow.png	제한 없음	비행선의 그림자	
	Gameover.png	544 X 416		
Graphics/Pictures	Window.png	128 X 128	영역 당 64 X 64	
	Pictures	제한 없음		
Audio/BGM				
Audio/BGS			OGG, WMA,	
Audio/ME			MP3, WAV, MID	
Audio/SE				
Movie		544 X 416	OGV6	

▲ RPG Maker VX Ace의 소재 규격입니다

데이터베이스 에디터

데이터베이스 에디터에서는 게임 내에서 필요한 데이터들을 쉽게 관리할 수 있습니다. 이 에디터를 통해 데이터를 관리하면 프로젝트 폴더 Data 폴더 내에 있는 데이터 파일들과 Game.ini 파일이 수정됩니다. 이 데이터 파일의 문자열들은 루비의 보안 관련 모듈로 암호화되어 저장됩니다. 내부 변수들은 도움말의 RPGVX Ace Data Structures의 세부항목에서 어떤 것들이 있는지 살펴 볼 수 있습니다.



▲ 데이터베이스 에디터

데이터베이스 한글화

위에서 데이터베이스는 파일을 조작하는 에디터라고 알려드렸습니다. 데이터베이스 에디터를 통해 데이터 파일을 조작할 수 있습니다. 그래서 반대로 생각하면 미리 한글로 바꿔놓은 파일들이 있다면 반대로 데이터베이스의 내용을 전부 바꿀 수도 있다는 소리가 됩니다. 데이터베이스 파일들의 내용들은 게임이 시작되면 스크립트 에디터에서 암호화된 내용을 문자열로 해석하여 \$data_로 시작되는 변수에 저장시킵니다. 데이터베이스 파일들은 총 13개이며 저장되는 파일들을 나열하면 아래와 같습니다.

폴더/<파일이름.확장자>	내용	기본 언어	패치 후
Data/Actors.rvdata2	주인공 설정집	영어	한국어
Data/Animations.rvdata2	애니메이션 설정집	영어	한국어
Data/Armors.rvdata2	방어구 설정집	영어	한국어
Data/Classes.rvdata2	직업 설정집	영어	한국어

Data/	커먼이벤트 설정집	영어	한국어
CommonEvents.rvdata2			
Data/Enemies.rvdata2	적 설정집	영어	한국어
Data/Items.rvdata2	아이템 설정집	영어	한국어
Data/Skills.rvdata2	스킬 설정집	영어	한국어
Data/States.rvdata2	상태 설정집	영어	한국어
Data/System.rvdata2	시스템 설정집	영어	한국어
Data/Tilesets.rvdata2	타일셋 설정집	영어	한국어
Data/Troops.rvdata2	적 그룹 설정집	영어	한국어
Data/Weapons.rvdata2	무기구 설정집	영어	한국어

▲ 위 파일들을 직접 폴더에 프로젝트를 새로 만들 때 마다 매번 덮어씌워줘야 합니다

1. 알만틀에서는 안드로이드, iOS, HTML5 등의 배포 기능을 지원하지 않습니다. 그래서 안드로이드 앱을 만들려면 직접 포팅 작업을 해야합니다. 루비 스크립트를 해석할 수 있는 플랫폼에 맞는 루비 인터프리터가 필요한데 인터프리터로는 주로 JRuby + Ruboto(Java), IronRuby(C#), JNI(NDK, 자바 네이티브 인터페이스)가 사용됩니다. 하지만 플랫폼에 맞는 모듈을 새로 만들어줘야 하기 때문에 이렇게까지 하는 사람은 거의 없다고 보면 됩니다.
2. 네이티브 언어로 보통 2D 게임을 만들 때 맵 에디터를 먼저 만들어야 하는데 이때 C, C++, Data Structure, C++ STL, Design Pattern, Win32API, MFC, DirectX 등의 지식이 필요합니다.
3. Dynamic-Link Library 의 약자로 윈도우에서 구현된 동적 라이브러리를 말합니다.
4. 데이터가 기록되어있는 주요 스크립트를 수정하면 데이터베이스 최대치도 확장할 수 있습니다. (관련 스크립트 이용 또는 확장 유тиль 사용)
5. C#의 기반 루비 인터프리터인 IronRuby로 만든 스크립트 편집기 'Gemini Editor'를 통해 스크립트 편집이 가능합니다.
6. OGV 확장자 인코딩 방법 - <http://www.rpgmakervxace.net/topic/15005-converting-movies-for-vx-ace-easy-free-no-script-no-watermark/>

[02강] 맵과 오토타일 - RPG 만들기 VX Ace 강좌

시작하기

2014/10/28 17:50

<http://blog.naver.com/biud436/220164425450>

RPG Maker VX Ace 왕초보탈출 2

"맵, 맵의 속성, 샘플맵, 던전의 생성, 좌표계, 오토타일, 데미지 플래그, 수풀, 카운터 플래그, 지형 ID, 지역 ID"

맵의 정의

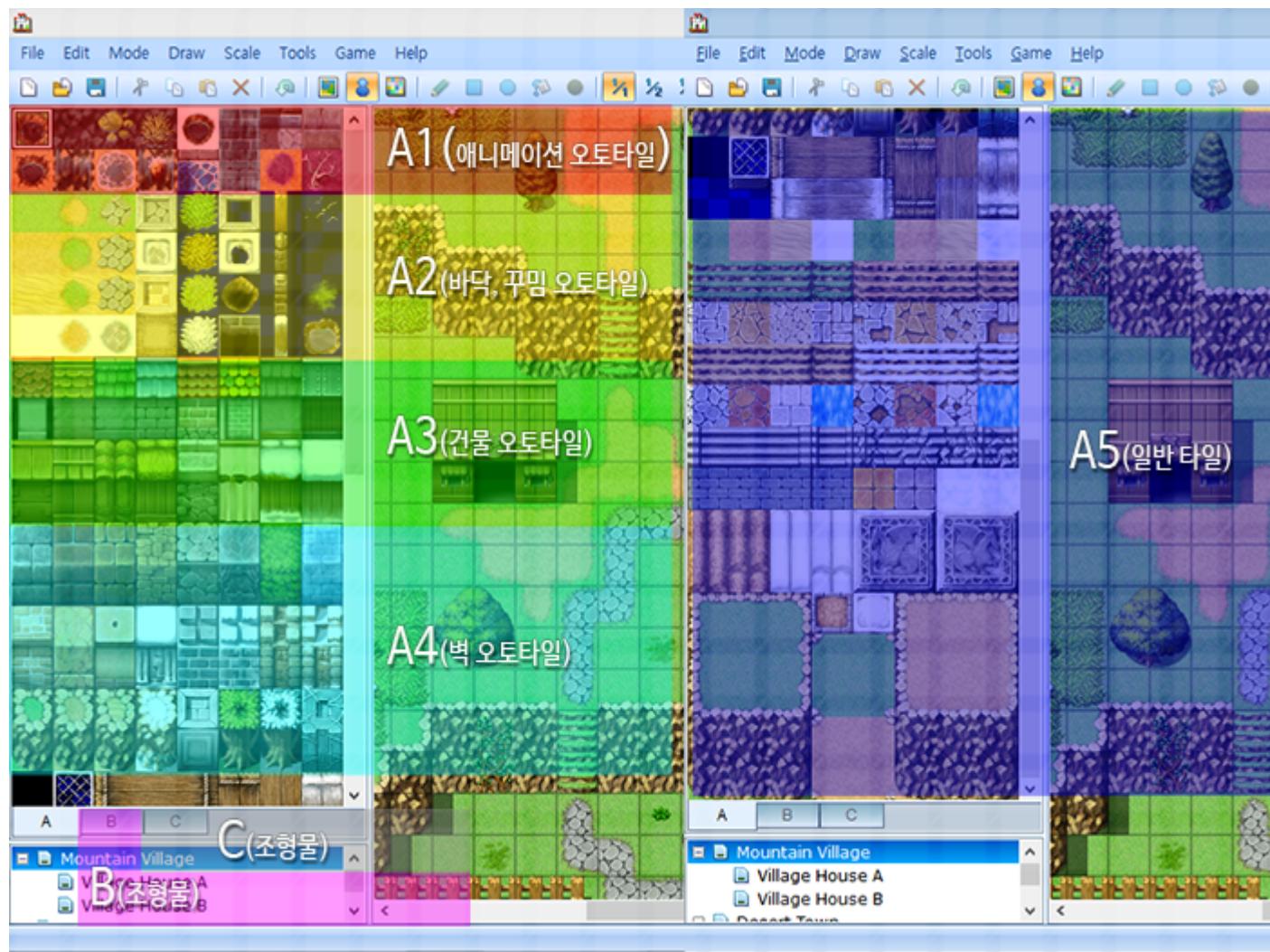
맵이라는 것은 게임 세계가 정의되는 공간입니다. 화장실에 있는 타일들과 같이 이 세계는 타일들의 집합이기도 합니다. 타일에는 다양한 정보들이 담겨있습니다. 통행 정보가 있을 수도 있고, 통행 시에 데미지를 받는 지 받지 않는 지에 대한 값들이 플래그라고 하여 16진수 형태로 저장되어있고 캐릭터를 비롯한 각종 이벤트들이 정의되어있는 공간이기도 합니다.

한 타일의 폭과 높이는 32픽셀이며 창의 크기에 비례하여 타일이 들어갈 갯수가 산정됩니다. 이렇게 맵에 들어갈 수 있는 타일들을 기점으로 X좌표, Y좌표를 산출할 수 있습니다. 타일은 제작자가 정의한 맵의 최대 크기만큼 들어가게 됩니다. 이렇게 맵에는 많은 타일들이 각 개 다른 정보들을 담고 있습니다.

RPG Maker VX Ace에서는 이렇게 정의된 각종 타일 정보들을 맵 데이터(MapXXX.rvdata2)로 만들어서 데이터 폴더에 저장시킵니다. 이렇게 만들어진 맵 데이터는 게임이 실행되면 스크립트 에디터의 Spriteset_Map 부분에서 타일들을 해석하면서 맵을 그래픽으로 표현하면서 우리가 알고 있는 게임 세상으로 만들어줍니다.

맵은 어떻게 이뤄지는가?

맵이라는 것은 타일이 어떻게 배치되는지에 대한 정보를 담고 있습니다. 맵은 타일을 배치하고 결합하는 작업에서부터 시작됩니다. 맵을 만들기 위해서는 RPG Maker VX Ace의 맵의 구조를 먼저 익혀야 합니다.



▲ 타일의 전개도

○ 필드와 조형물의 구분

RPG Maker VX Ace에는 필드와 조형물을 배치할 수 있는 공간이 단 하나뿐입니다. 이런 공간을 레이어라고 부르는데 RPG Maker VX Ace에는 이런 타일 그래픽들을 배치할 수 있는 레이어라는 공간이 표면 상으로 단 하나뿐이고 추가 레

이어가 없습니다. RPG Maker VX Ace에서는 타일 그래픽들을 배치하기 위해 그렇게 많은 레이어가 필요하지 않습니다. 하나여도 충분합니다. 그 이유는 타일들이 조합 특징에 따라 명칭 별로 나뉘어있기 때문입니다. 레이어는 따로 선택할 순 없지만 타일에 따라 그래픽을 알아서 알맞는 공간에 알아서 조합되면서 배치됩니다.

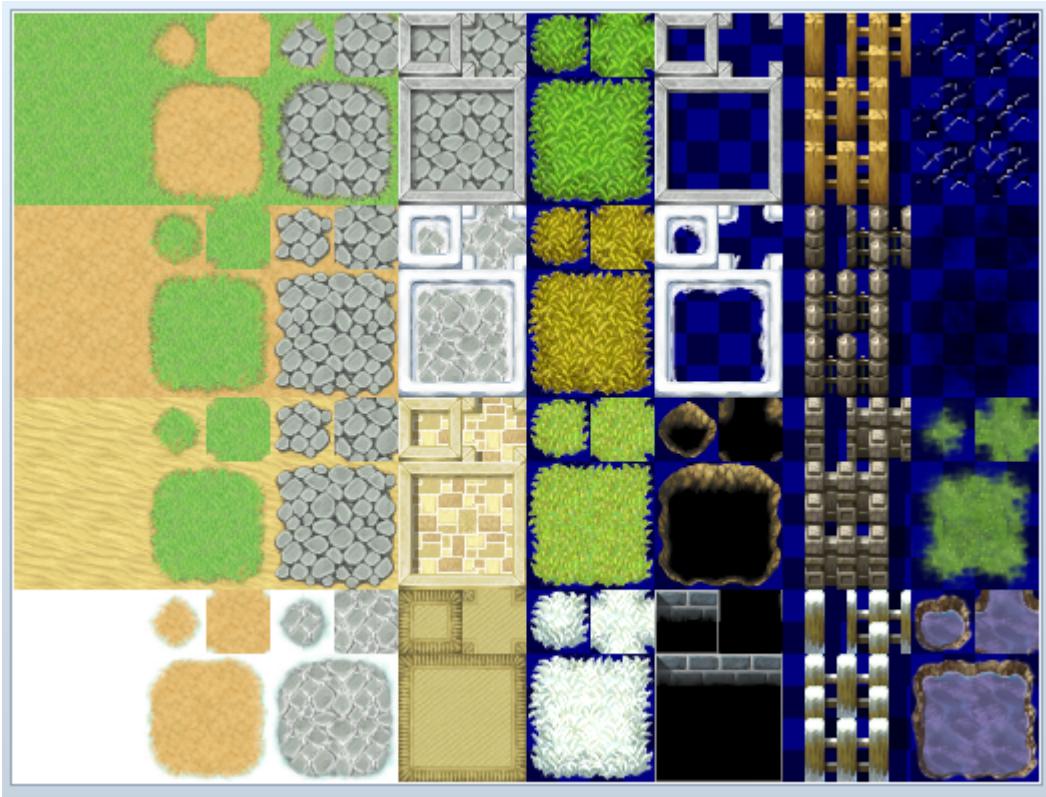
명칭	타일 분류	타일 형태	방향 설정	타일 특징
A1	오토타일	바다, 용암	불가능	애니메이션 타일입니다
A2	오토타일	땅	불가능	같은 타일이라도 투명한 영역이 있는 타일들과는 겹칠 수 있습니다
A3	오토타일	건물	불가능	지붕과 건물의 벽을 만들 수 있으며 건물이 없을 경우 따로 설정하지 않습니다. 지나다닐 수 없습니다
A4	오토타일	벽	불가능	벽을 표현하는 타일 그룹이며 벽 위를 지나다닐 수 있습니다
A5	고정타일	일반	가능	타일 한 칸을 차지하는 계단 같은 타일들입니다
B, C	고정타일	조형물	가능	조형물을 배치하고자 할 때 사용합니다

○ 오토 타일과 고정 타일

A탭의 A1 ~ A4 타일셋 부분에는 오토타일이라고 불리고 있는 특수한 타일들이 모여있습니다. 예를 들면 부드럽게 이어지는 길목과 담장 등을 제대로 이어가려면 많은 타일들이 필요합니다. 또한 방향도 다르고 형태도 달라져야 합니다. 그것을 일일히 배치하려면 힘이 들겠죠. 그래서 오토타일이라는 기능이 있습니다. 같은 형태지만 방향과 다른 타일들과의 경계에 따라 형태가 달라져야 하는 부분을 자동으로 툴에서 해당 형태에 맞는 그래픽을 찾아서 배치해주는 기능입니다. 타일이 배치되면 타일이 어떻게 이어져있든 간에 해당 타일의 형태에 맞게 자동으로 그래픽이 변경되면서 맞는 형태로 그래픽이 깔리게 됩니다.

▪ A탭의 A2 타일셋

A2의 타일셋은 땅을 그리기 위한 것으로 기본 그라운드 타일과 기본 타일들을 꾸밀 수 있는 타일들로 나뉘어져 있습니다.



기본 타일을 깔고 그 위에 길을 낼 때 유용하게 사용할 수 있습니다. 기본 타일 위에 기본 타일을 꾸며주는 담장을 그릴 수도 있습니다. 그렇게 배치한 오토타일은 하나로 합성되서 보이게 됩니다. 그런 일을 A2 칩셋이 담당합니다.

- B탭 ~ E탭

B탭 ~ E탭까지는 일반 조형물들이 위치하며 타일셋 원형은 아래와 같은 형태로 되어있습니다만 실제 배치 때에는 8열 X 32행으로 배치됩니다. 8열 X 32행은 256개의 조형물이 있다는 것을 나타냅니다



벽과 건물 등이 모여있는 A탭과는 달리 B~E탭의 조형물들은 투명한 부분이 있습니다. 나무, 탁자, 벽에 걸린 시계, 마른나뭇바닥 위에 덩그라니 놓여있는 침대와 같은 조형물들은 바깥이나 내부에서 꾸며주는 역할을 하기 때문에 일반적으로 홀로 사용되기 보다는 조형물이기 때문에 A탭의 타일들 위에 놓여있게 됩니다. 일반적으로 C탭까지 사용되지만, E탭의 타일셋까지 있다면 E탭까지도 등록해서 사용할 수 있습니다.



▲ A탭의 바닥타일 위에 B탭의 조형물들이 올라가 있는 모습

이러한 규칙들은 RPG Maker VX Ace 의 타일셋 방식에선 절대적인 규칙으로 작용할 수 밖에 없기 때문에 추후에 칩셋을 등록하더라도 이러한 규칙에 맞게 타일셋들을 만들어서 등록해줘야 합니다.

- **스크립트 내부에만 존재하는 레이어의 개념!**

사실 이러한 것이 가능한 이유는 스크립트 내부에서 레이어를 따로 자동으로 나누면서 계산하고 있기 때문입니다. A탭은 0번 레이어로 자동 지정되어 있고, B탭은 1번 레이어, C탭은 2번 레이어로 지정되며 건물의 그림자들과 지역 ID는 3번 레이어로 따로 지정되며 계산되고 있습니다. 이것은 타일의 성격을 구분하기 위한 스크립트 내부 기준인데 스크립트 내부에서 조차 레이어라는 기준이 없다면 불편함이 많이 따릅니다.

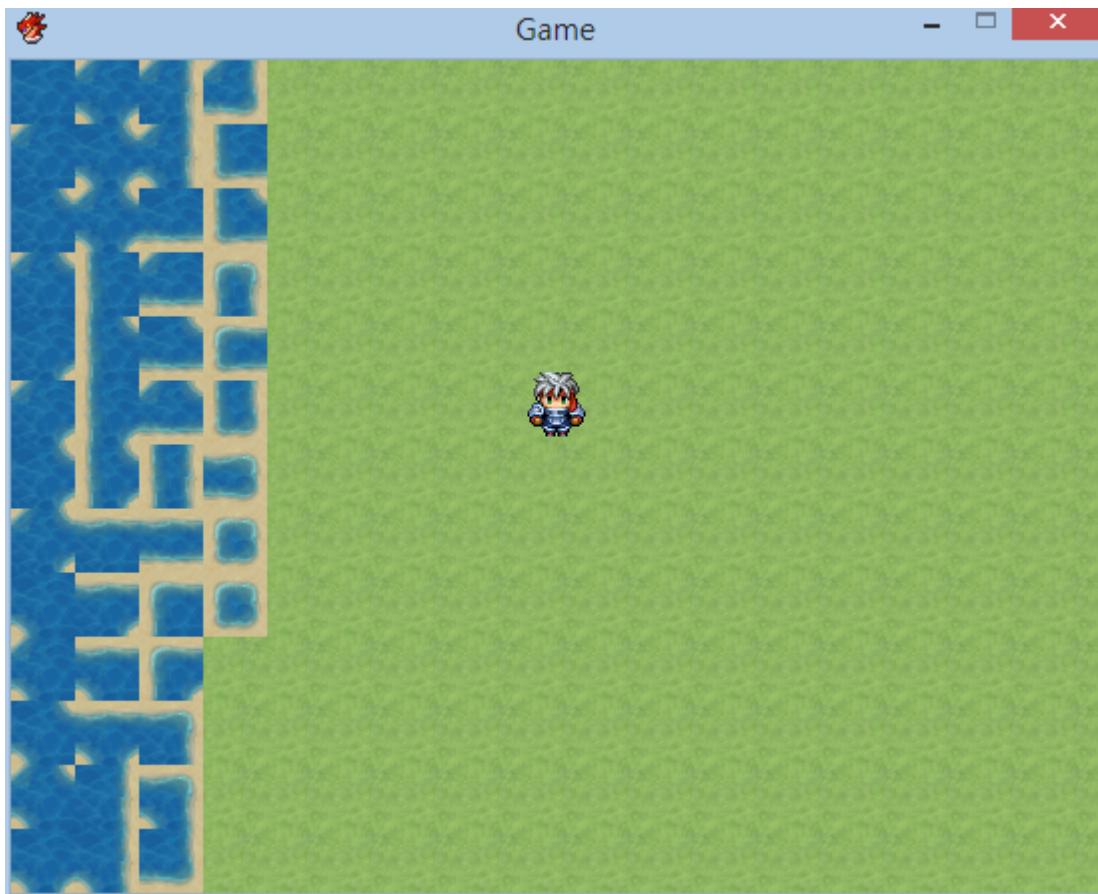
예를 들면 주인공이 서있는 타일 위에 나무가 있다는 것을 틀에서는 찾기가 힘들 수 있습니다. 그래서 내부적으로 0번 레이어(A탭)의 길 위는 지나갈 수 있지만 그 길 위에 1번 레이어(B탭)의 나무가 있으면 지나 갈 수 없다는 것을 쉽게 파악하기 위해 타일마다 레이어라는 번호를 붙여놓은 것입니다. 표면적으로 레이어가 없는 것처럼 보이지만 사실상 타일셋의 위치가 타일셋의 조합 성격에 따라 아예 고정되어있어서 레이어가 이미 나뉘어져있는 것과 마찬가지라고 볼 수 있습니다.

- **장점은 빠르고 쉬운 맵배치! 단점은 DLC의 유혹!**

전작인 RPG Maker XP 에서는 레이어를 나눠가면서 맵배치를 해야 했으므로 시간이 많이 소요되었습니다. 그러나 RPG Maker VX Ace에서는 레이어가 자동으로 나뉘어지기 때문에 나눌 필요가 없어서 시간이 단축되어 더 빠르게 게임을 만들 수 있게 된 것입니다. 편리하고 시간이 단축되어 좋아졌다고 생각하실 수도 있겠지만 사실 잊은 것도 있습니다. 타일셋을 정해진 형식에 따라 만들어야 하기 때문에 타일셋을 쉽게 만들 수가 없게 되었습니다. 그래서 RPG Maker VX Ace의 타일셋들은 전작들과는 달리 다채롭지 못합니다. 찾아보기가 힘든 것이 현실이고 RPG 만들기 2003 시절이 주류였던 시절처럼 다양한 칩셋이 있는 것도 아닙니다. 형식을 지켜서 만들어야 하고 은근히 찍어야 될 칩셋도 많아서 만들기가 까다롭고 더 어려워졌기 때문입니다. 그래서 개발사에서는 유료로 다운받을 수 있는 DLC 컨텐츠를 통해 타일셋을 많이 판매하고 있기도 합니다.

오토타일(Autotile)

오토타일은 주변 타일의 형태를 감지해서 형태에 알맞는 타일을 골라 자동으로 삽입해주는 기능을 가지고 있는 타일입니다. 주변 형태에 따라 만들 수 있는 타일의 경우의 수가 48가지나 되기 때문에 이런 기능이 존재하는 것입니다.



▲ 한 오토타일로 만들 수 있는 타일은 총 48개입니다.

- 오토타일이란 무엇인가?

오토타일은 타일의 경계 상태에 따라 폭과 높이가 16픽셀인 가장 자리 타일 4개가 규칙에 따라 서로 결합되면서 한 칸의 타일을 완성합니다. 이렇게 자동으로 결합되는 타일들을 오토타일이라고 부릅니다.

- 오토타일의 규칙



▲ 꾸밈 오토타일 중 책상 오토타일은 특별하게 아래 부분이 더 그려지게 되어있습니다

오토타일은 주로 기본 타일과 꾸밈 타일로 나뉘어집니다. 위 칩에서 가장 자리에 있는 타일 4개가 합성되면서 맵에 표현됩니다. 꾸밈 타일은 기본 타일을 꾸며주는 역할로 담장과 책상, 수풀등이 있으며 특히 책상을 표현하기 위한 오토타일은 특별하게 아래 부분이 위와 같이 더 그려진다는 점을 알아두시길 바랍니다.

맵 그리기

맵을 그릴 수 있는 도구가 상단 툴바에 비치되어있으며 누구나 알 수 있는 친숙한 아이콘으로 되어있기 때문에 별도로 설명하지 않아도 한 번 그려보면 무슨 기능을 하는 버튼인지 알 수 있을 것입니다. 그래도 궁금하신 분들을 위해 표를 첨부하였습니다.

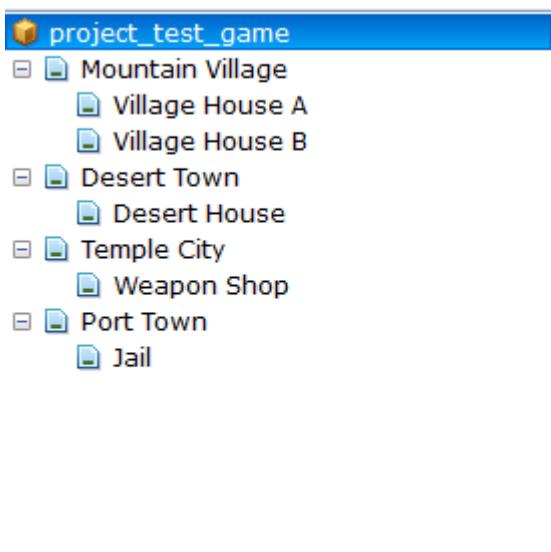
구분	아이콘	설명
Pencil		한 칸씩 타일을 그립니다
Rectangle		사각형으로 타일을 그립니다
Ellipse		원모양으로 타일을 그립니다
Flood Fill		전체를 현재 타일로 채웁니다
Shadow Pen		그림자를 그립니다

▲ 맵 타일을 배치할 수 있는 그리기 도구들

드래그를 하는 방식으로 맵 영역을 선택하여 부분 복사를 할 수도 있습니다.

맵의 트리

맵의 트리에서는 맵의 구조를 한 눈에 볼 수 있습니다. 맵의 트리를 통해서 마을이나 대륙 등을 그리고 맵의 구조를 설계할 수 있습니다. 풀더 탐색기처럼 되어있기 때문에 윈도우를 사용하는 분들이시라면 이미 익숙한 부분이기도 합니다.



▲ 한 눈에 볼 수 있는 맵의 구조

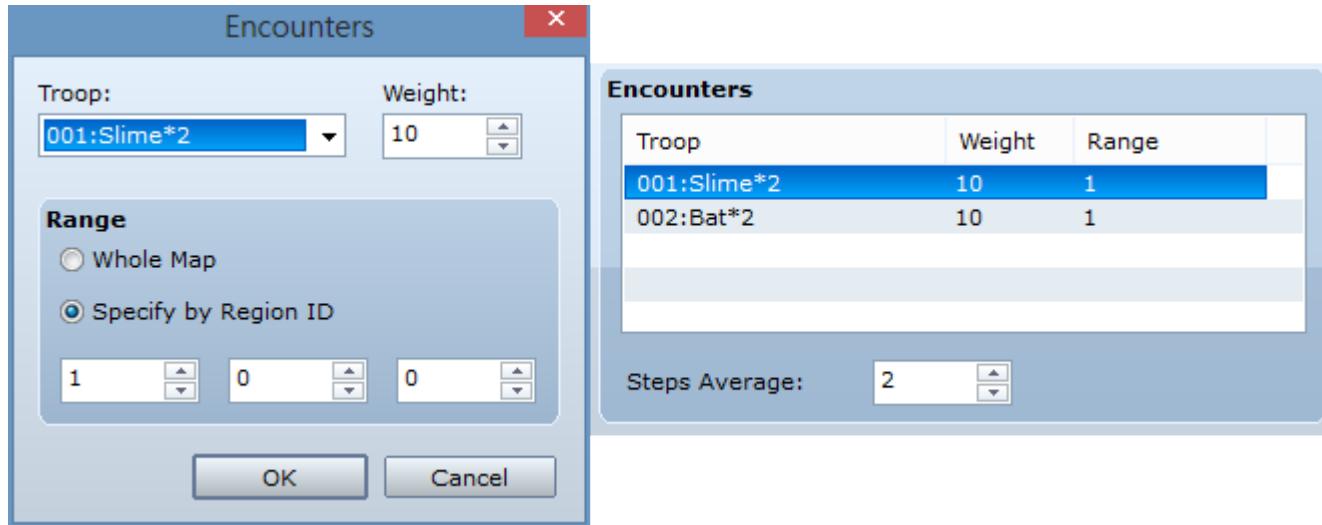
○ 맵의 속성(Map Properties)

윈도우를 사용하는 분들이라면 이미 폴더의 속성을 보는 방법을 모르는 분들이 없을 것입니다. 맵의 속성도 마찬가지 방법으로 맵의 속성들을 확인할 수 있습니다. 속성을 확인하게 되면 맵 구성에 필요한 다양한 정보를 설정할 수 있습니다.

구분	세부 명칭	설명
General Settings	ID	맵을 구별하기 위해 붙여진 고유의 숫자입니다.
	Name	맵의 이름으로 원하는대로 수정이 가능할 수 있습니다
	Display Name	게임에 표시될 맵의 이름을 적는 란입니다.
	Width	맵의 폭으로 가로 길이를 나타냅니다
	Height	맵의 높이이며 세로 길이를 나타냅니다
	No Loop	루프를 설정하지 않습니다
	Vertical Loop	세로 방향 루프를 나타냅니다
	Horizontal Loop	가로 방향 루프를 나타냅니다
	Both Loop	모든 방향의 루프를 나타냅니다
	Specify Battleback	던제 전투에서의 배경화면을 설정합니다
	Auto-Change BGM	맵의 BGM을 자동 재생합니다
	Auto-Change BGS	맵의 BGS를 자동 재생합니다
	Disable Dashing	달리기가 불가능한 맵으로 만듭니다
Parallax Background	Graphics	원경 그래픽을 설정할 수 있습니다
	Loop Horizontal	가로 방향으로 원경을 스크롤합니다
	Loop Vertical	세로 방향으로 원경을 스크롤합니다
	Auto Scroll	스크롤의 방향과 속도를 수치로 결정할 수 있습니다
	Show in the Editor	에디터에도 원경을 그립니다
Encounters		특정 지역이나 전체 맵에서 적을 만나 자동으로 전투가 발생합니다
Note		이 맵에 대한 설명을 적을 수도 있고 게임에는 영향을 미치지 않습니다

▪ Encounters

특정 지역이나 전체 맵에서 적을 만나 자동으로 던제 전투가 발생하게 하는 기능입니다.



▲ 맵 속성에서 엔카운터를 설정한 모습

Whole Map은 맵 전체 범위에서 적을 만나게 함을 의미하며 Specify by Region ID는 지역 ID라는 것을 설정하여 특정 지역에 있을 때에만 전투가 시작되게 함을 의미합니다.

Steps Average는 플레이어의 걸음 수를 나타내는데 2라는 수치는 평균적으로 두 걸음을 걷게 되면 전투가 발생해서 적을 만나게 됩니다.

- 중량(Weight)

이때 중요한 것은 중량(Weight)이라는 수치입니다. 이것은 어떤 적이 등장하는가를 고르기 위한 확률입니다. 이 방식은 확률을 구하는 방식과 같습니다. 같은 범위 내에 있는 모든 적 그룹의 중량을 더해서 확률을 구할 수 있습니다. 예를 들면, 슬라임(Slime) 그룹과 박쥐(Bat) 그룹은 중량이 각각 10입니다.

그렇다면 모든 그룹의 중량을 더하면 총 중량은 20이 됩니다. 총 중량에 대한 각 그룹의 확률을 구하면,

$(10/20) * 100\% = 50\%$ 확률로 슬라임과 박쥐를 번갈아 만나게 됨을 알 수 있습니다.

- 샘플맵의 생성

- 샘플맵이란?

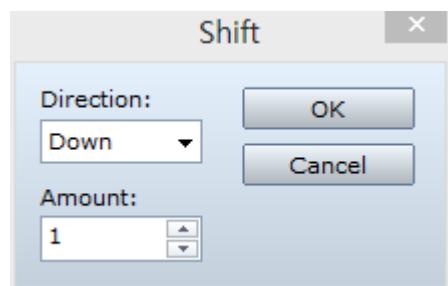
RPG Maker VX Ace의 개발자에서 미리 만들어놓은 맵을 불러올 수 있는 기능으로 스크린샷과 함께 RPG Maker VX Ace에서 어떤 맵을 어떤 형태로 다양하게 만들 수 있는지 그 배치 형태를 이 기능을 통해 미리 알아볼 수 있습니다.

- 샘플맵이 보관된 위치

샘플맵들은 미리 만들어진 맵 데이터로 RPG Maker VX Ace가 설치된 폴더에 샘플맵이란 폴더에 이미 저장되어있습니다.

- 쉬프트(Shift)

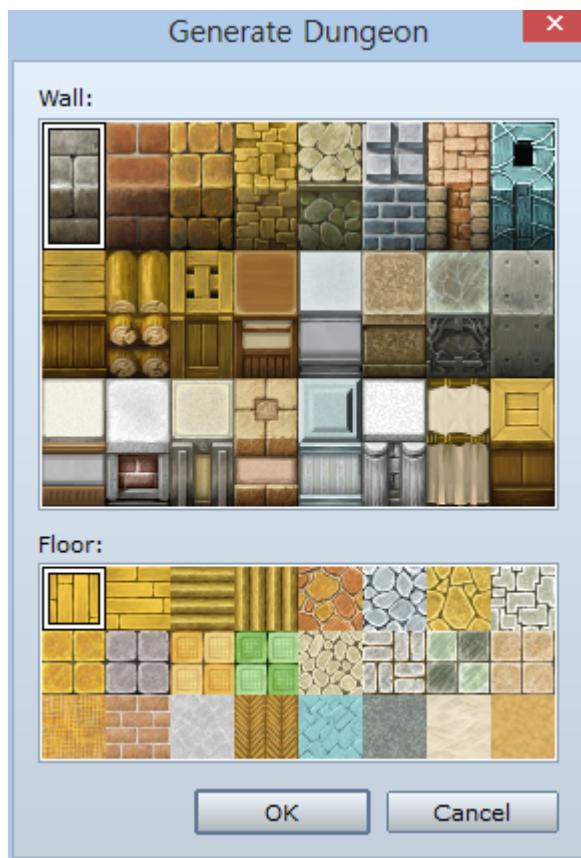
맵 전체를 특정 방향으로 옮겨놓을 수 있는 기능입니다.



▲ 특정 방향으로 맵을 옮길 수 있습니다

- 던전의 생성(Generate Dungeon)

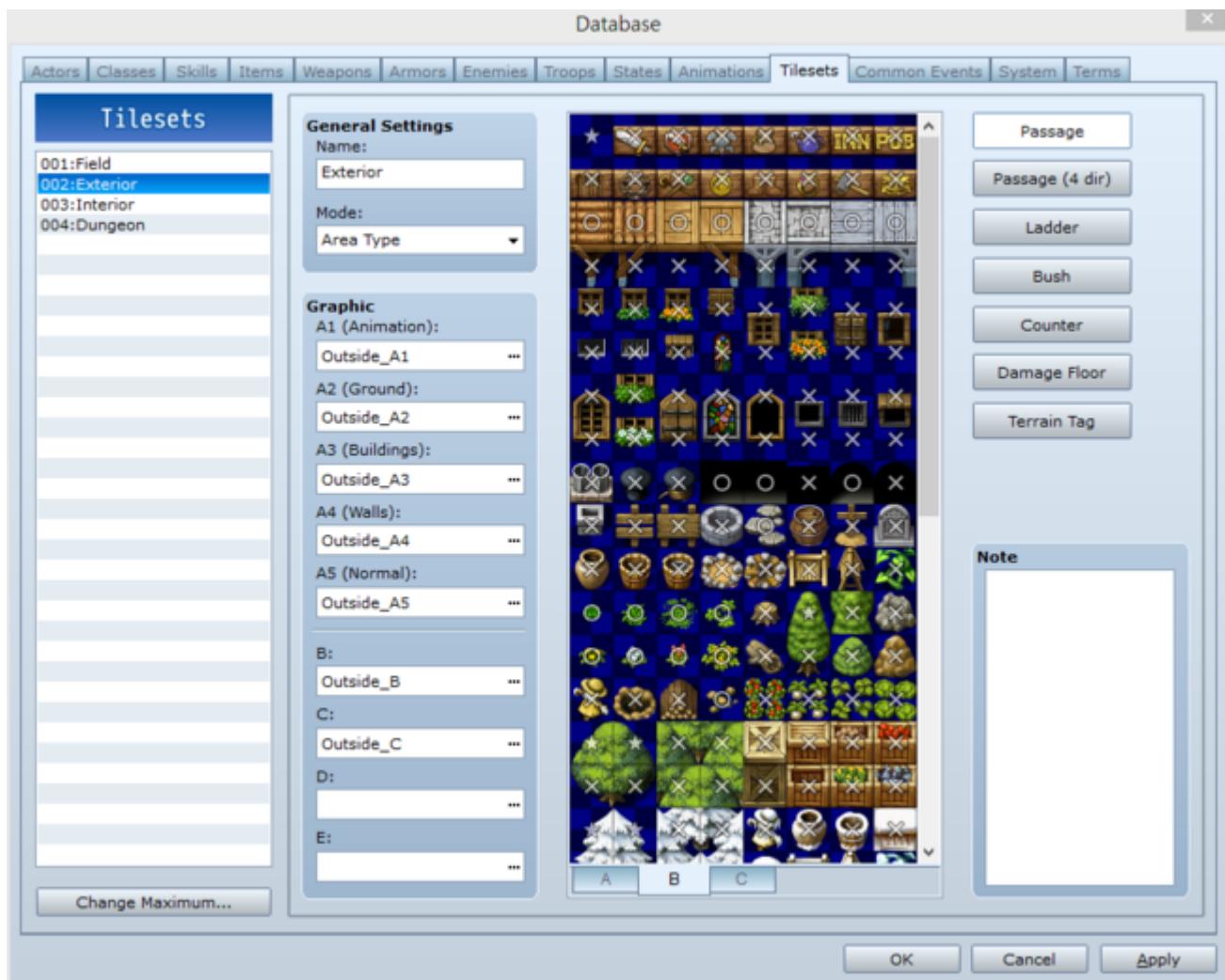
이미 생성된 맵에 벽과 바닥 타일을 지정하면 던전이 자동으로 생성되는 기능입니다.



▲ OK를 누르면 자동으로 던전 지형이 생성됩니다

타일셋의 설정

데이터베이스 에디터에서 Tilesets 탭을 누르면 타일셋을 설정할 수 있는 버튼이 보실 수 있습니다. 통행 설정은 다음과 같은 특수 기호로 표시되고 있습니다.

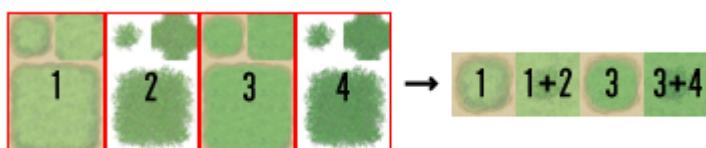


▲ 타일셋을 설정하고 있는 모습

- 모드

- Field Type

월드맵을 만들기 위한 모드로 이 모드에서는 꽉 채운 그라운드 타일로 가득합니다. 경계가 투명한 꾸밈 타일과 일반 그라운드 타일들이 합성된 상태로 나오게 됩니다



▲ 합성 과정의 전개도(도움말에서 발췌)

- Area Type

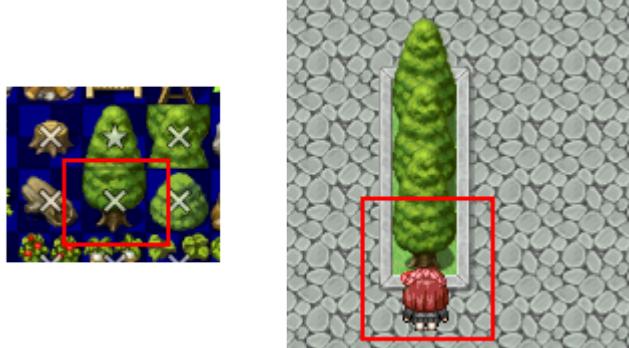
던전과 마을을 만들 수 있는 모드이며 이 모드에서는 한 타일을 꽉 채운 A2의 그라운드 타일들과 경계가 투명한 꾸밈 타일들을 합성할 수 있습니다.

- VX 호환 모드(VX Compatible)

VX와의 칩셋 호환을 위해 존재하는 모드로 이 모드에서는 꽉 채운 그라운드 타일로 가득합니다. 경계가 투명한 꾸밈 타일이 사라지게 되며 지형을 혼합하는 합성 기능을 지원하지 않습니다

- 통행 설정(Passage)

몇 가지 기호를 통해 타일의 통행 금지, 통행 가능, 그래픽 우선 순위를 설정할 수 있습니다.



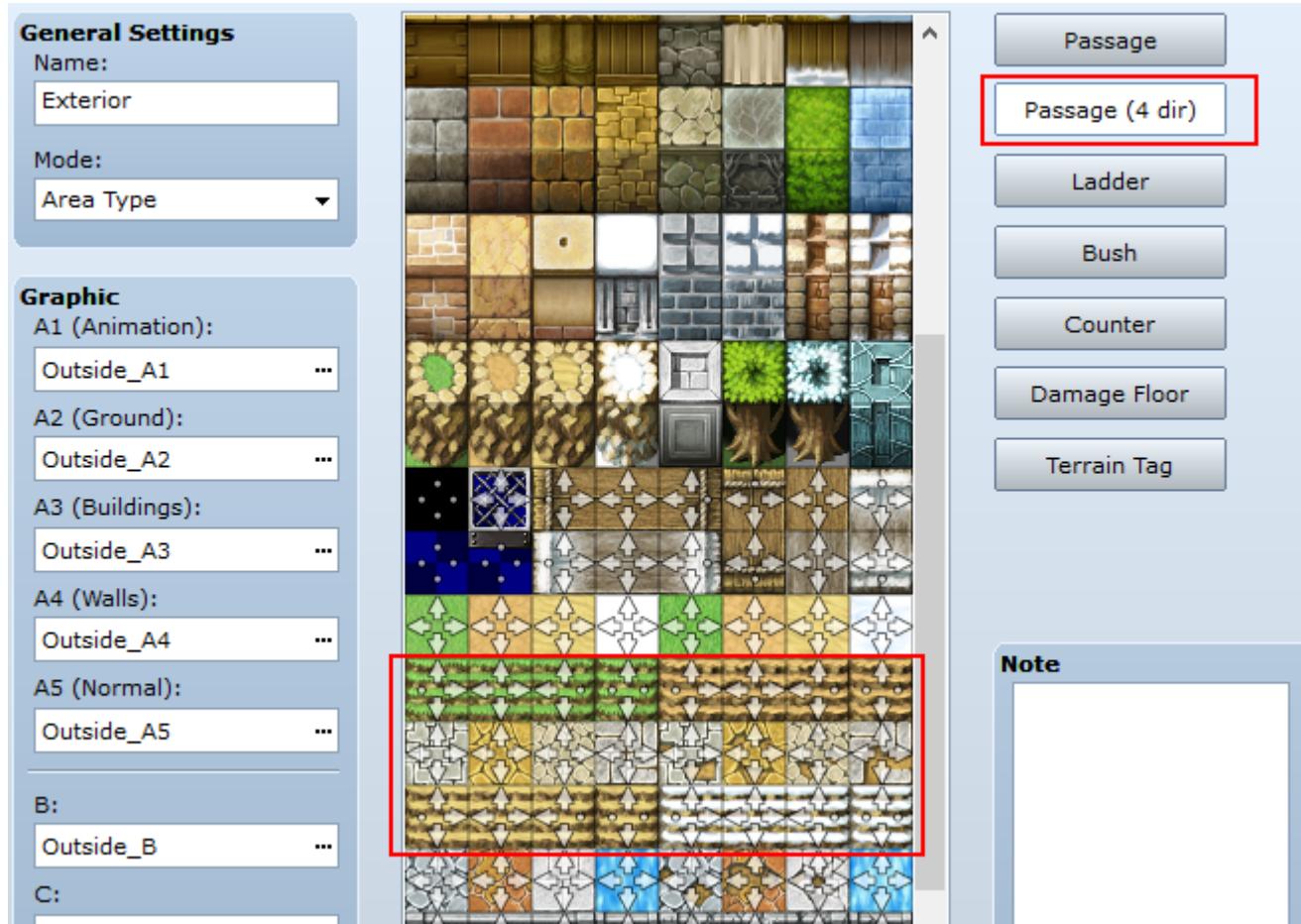
▲ 타일에 X 기호를 설정하면 통행이 불가능한 타일이 됩니다



▲ 타일에 ★ 기호를 설정하면 타일이 플레이어보다 더 앞쪽에서 그려집니다

- 4방향 통행 설정(Passage 4 Dir)

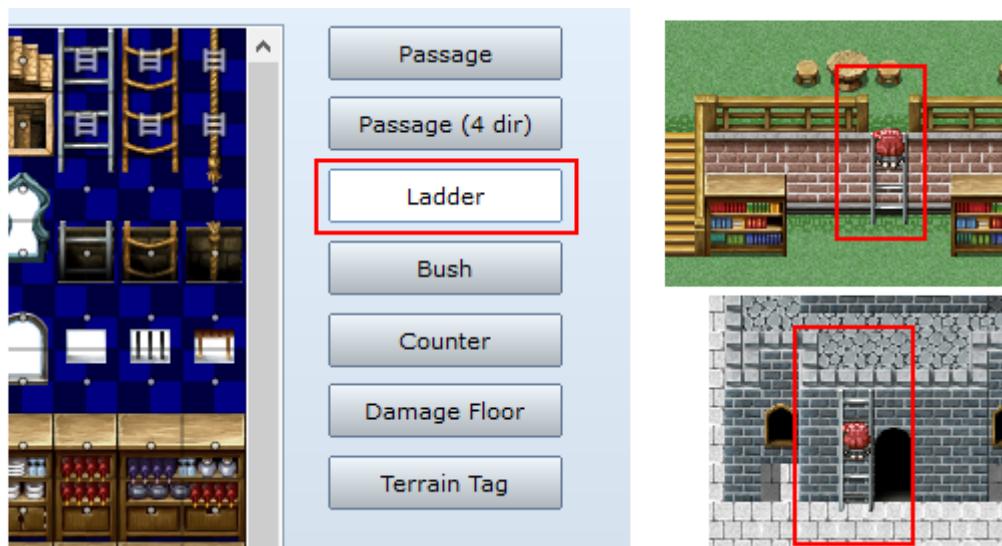
4방향 설정을 통해 타일의 가장 자리 부분의 이동 설정을 제한할 수 있습니다. 그러나 오토타일은 4방향 통행 설정을 할 수 없습니다.



▲ 계단이나 다리 같은 일방향으로 통행을 해야 하는 지형에 방향 제한을 걸어두면 좋습니다

○ 사다리 설정(Ladder)

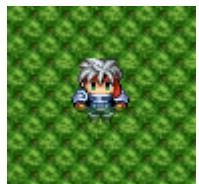
사다리 모양의 기호를 타일에 설정하면 플레이어가 이 타일을 사다리로 인식하여 자동으로 타고 올라가거나 내려가는 모션을 취하게 됩니다.



▲ 타일에 사다리 모양의 기호를 설정하면 사다리 타일이 됩니다

- 수풀 플래그(Bush)

수풀 플래그를 타일에 설정하면 플레이어의 몸체가 반 쯤 타일에 물에 잠기듯 잠기게 됩니다.



▲ 수풀에 잠겨있는 플레이어의 모습

- 카운터 속성(Counter)

카운터 속성 기호를 타일에 설정하면 아래 스크린샷과 같이 책상너머 한 칸 떨어진 곳의 NPC와 대화를 나눌 수 있게 됩니다.



▲ 카운터에 있는 NPC 이벤트가 실행됩니다.

- 데미지 바닥(Damage Floor)

데미지 플래그 기호를 타일에 설정하면 플레이어가 이 타일을 지나게 될 때 파티가 데미지를 받게 됩니다. 기본 데미지는 10이며 이 데미지 수치를 변경하려면 스크립트 에디터에서 Game_Actor 스크립트의 basic_floor_damage 메소드에 있는 수치값을 변경해줘야 합니다.



▲ Damage Floor 를 설정하면 체력이 10 낫게 됩니다

Script Editor

```

Script
Game_ActionResult
Game_BattlerBase
Game_Battler
Game_Actor
Game_Enemy
Game_Actors
Game_Unit
Game_Party
Game_Troop
Game_Map
Game_CommonEvent
Game_CharacterBase
Game_Character
Game_Player
Game_Follower
Game_Followers
Game_Vehicle
Game_Event
Game_Interpreter

Sprites
Sprite_Base
Sprite_Character
Sprite_Battler
Sprite_Picture
Sprite_Timer
Spriteset_Weather
Spriteset_Map
Spriteset_Battle

Windows

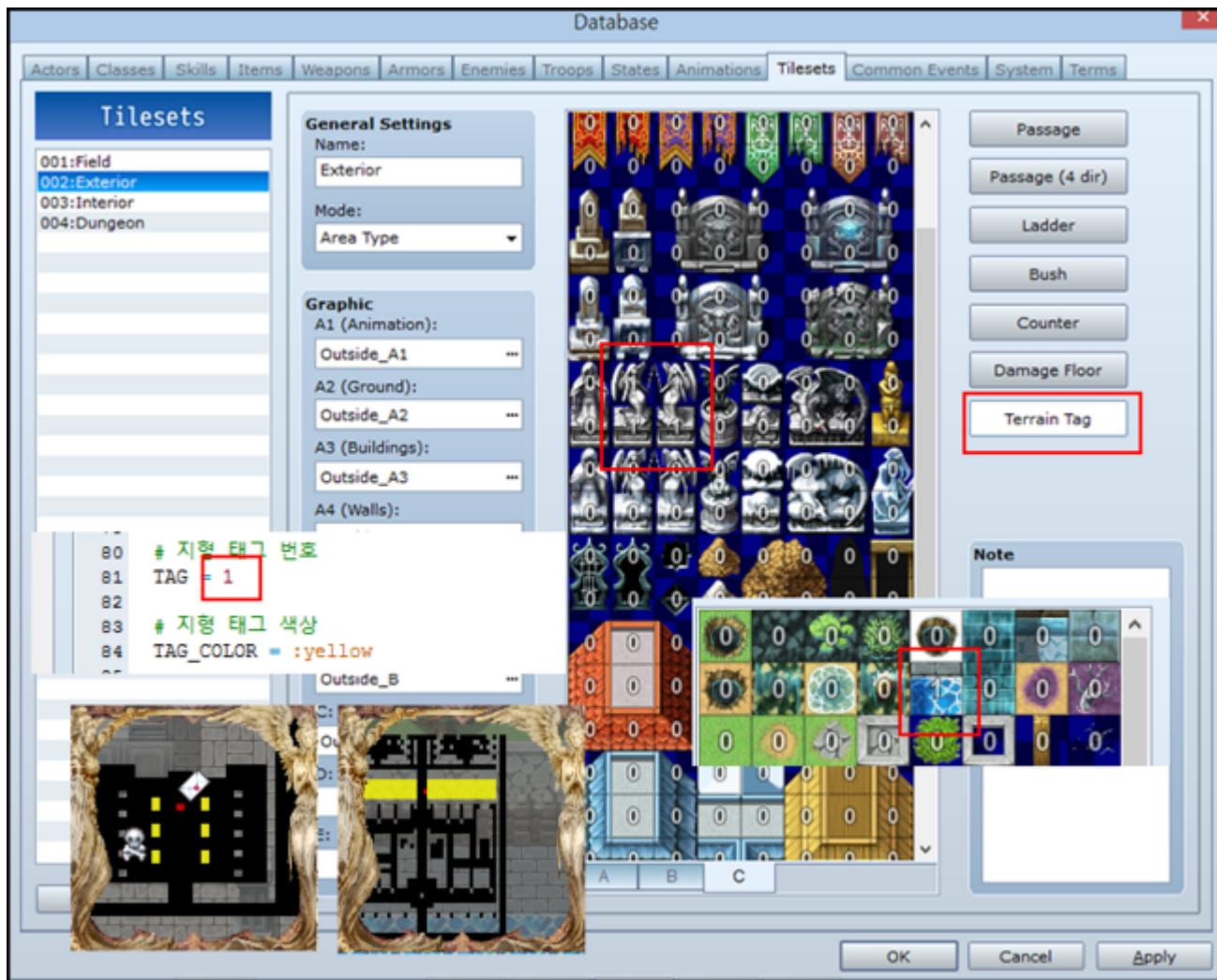
on_turn_end
perform_map_damage_effect if @result.hp_damage > 0
end
#
# * Determine Floor Effect
#
def check_floor_effect
execute_floor_damage if $game_player.on_damage_floor?
end
#
# * Floor Damage Processing
#
def execute_floor_damage
damage = (basic_floor_damage * fdr).to_i
self.hp -= [damage, max_floor_damage].min
perform_map_damage_effect if damage > 0
end
#
# * Get Base Value for Floor Damage
#
def basic_floor_damage
return 10
end
#
# * Get Maximum Value for Floor Damage
#
def max_floor_damage
$data_system.opt_floor_death ? hp : [hp - 1, 0].max
end
#
# * Execute Damage Effect on Map
#
def perform_map_damage_effect

```

▲ 데미지 수치는 위 스크린샷처럼 스크립트 에디터에서 수정할 수 있습니다

○ 지형 태그

특정 타일에 0번부터 7번까지의 번호를 새겨놓을 수 있습니다. 이렇게 새긴 번호는 나중에 배울 이벤트 에디터의 Get Location Info 명령으로 해당 타일에 새겨진 지형 태그 번호를 얻어올 수 있습니다.



▲ VXA 추천강좌 게시판에 있는 미니맵 시스템에서는 타일의 지형 태그값을 이용하여 미니맵을 그립니다

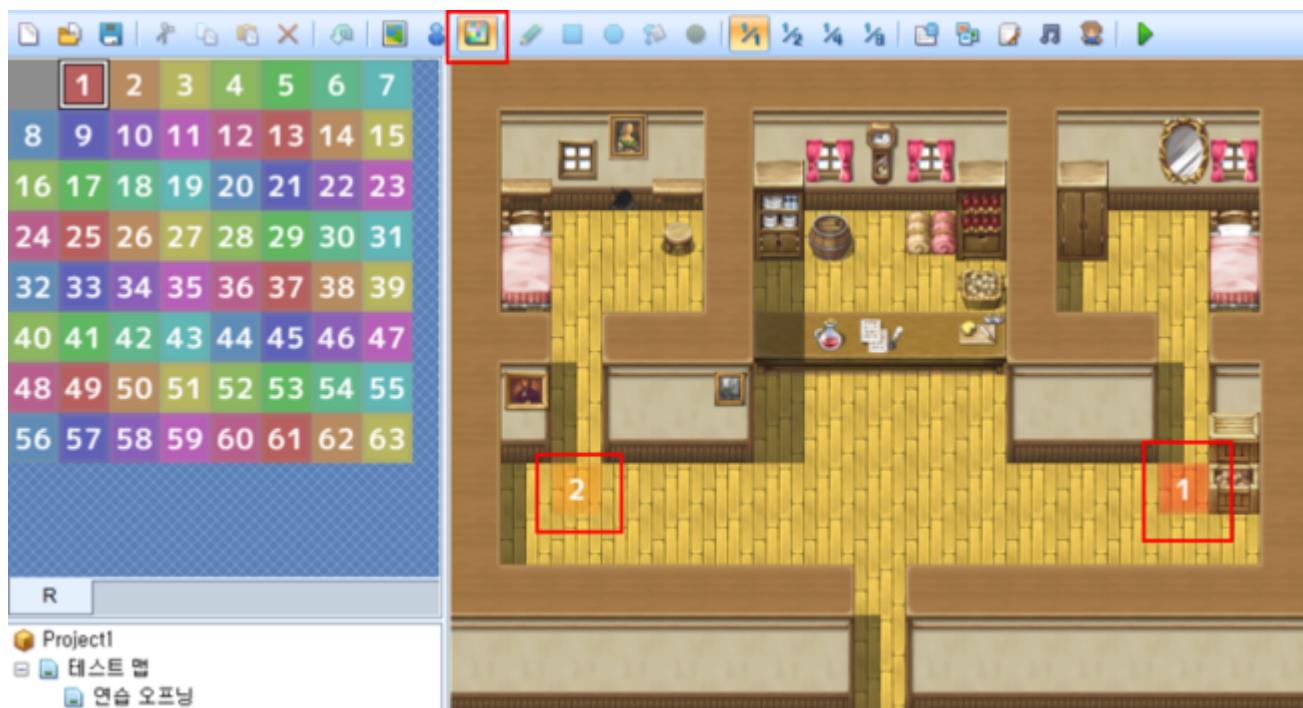
○ 지역 ID

지형 태그 이외에 맵 타일에 지역 ID를 맵 에디터에서 직접 지정할 수 있습니다. 이전에 학습한 엔카운터라고 부르는 적과의 만나는 시스템에서 사용되기도 합니다. 아래와 특정 위치에서만 아이템 사용하는 이벤트를 만들 때 더 유용하게 쓸 수 있는 기능입니다. 이 기능도 아직은 배우지 않았지만 Get Location Info 명령으로 해당 타일의 지역 ID를 가져올 수 있습니다.

관련 강좌

[\[07강\] 특정 위치에서만 아이템 사용하기](#)

[\[10강\] 캐릭터 선택 시스템 만들기](#)



▲ 특정 지역에서만 아이템 사용할 때도 지역 ID를 활용하게 됩니다

[03강] 이벤트의 기본 조건 - RPG 만들기 VX Ace 강좌

시작하기

2014/10/29 22:27

<http://blog.naver.com/biud436/220165759289>

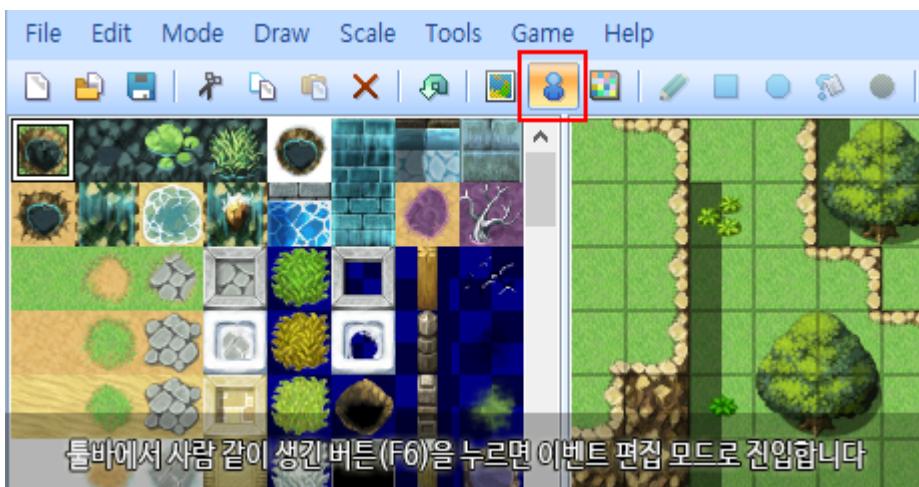
RPG Maker VX Ace

왕초보탈출 3

"이벤트의 기본 조건"

새 이벤트 만들기

RPG Maker VX Ace 라는 툴로 게임을 만들기 위해서는 먼저 이벤트를 만들고 이벤트를 조작할 줄 알아야 합니다.



오른쪽 타일 필드가 바둑판처럼 변했습니다. 더 이상 타일을 편집할 수도 없고 타일을 클릭할 수 있는 상태가 되었습니다. 타일을 편집할 수 있는 모드에서 이벤트를 만들기 위한 모드로 바뀌었기 때문입니다.



더블클릭하면 새로운 이벤트를 만들 수 있습니다

이벤트를 만들고자 하는 타일에 더블 클릭을 하게 되면 새로운 이벤트를 만들기 위한 편집창이 화면에 뜨게 됩니다. 새 이벤트라는 창에 떴다면 이제부터 강의를 따라올 수 있게 된 것입니다.

이벤트의 기본 조건

이벤트의 조작은 쪼꼬르 툴 역사에서 가장 오래된 강의이기도 하고 모든 버전의 툴이 거의 비슷한 형태입니다. 그만큼 쪼꼬르 툴에서 중요한 부분을 차지하고 있으며 포함된 기능이 상당히 많기 때문에 지난 강의처럼 래퍼런스 형식으로 진행하는 것은 어렵습니다. 그래서 게임 제작에 필요한 부분만 설명하면서 여러 단계로 나눠서 강의를 진행하도록 하겠습니다.

그 시작은 이벤트의 여러가지 기본 조건들입니다. 이벤트 기본 조건으로는 이벤트의 시작 조건, 그래픽 설정, 이동 커맨드, 우선순위, 이벤트의 실행조건 등이 있습니다.

New Event - ID:

Name: EV010	New Event Page	Copy Event Page	Paste Event Page
1			
Conditions <ul style="list-style-type: none"> <input type="checkbox"/> Switch ... is ON (1) <input type="checkbox"/> Switch ... is ON <input type="checkbox"/> Variable ... is <ul style="list-style-type: none"> or above <input type="checkbox"/> Self Switch ... is ON <input type="checkbox"/> Item ... exists <input type="checkbox"/> Actor ... exists 		Contents: @> [empty]	
Graphic (2) 		Autonomous Movement (3) <ul style="list-style-type: none"> Type: Fixed Move Route... Speed: 3: x2 Slower Freq: 3: Normal 	
Options (4) <ul style="list-style-type: none"> <input checked="" type="checkbox"/> Walking Anim. <input type="checkbox"/> Stepping Anim. <input type="checkbox"/> Direction Fix <input type="checkbox"/> Through 		Priority (5) <ul style="list-style-type: none"> Same as Characters 	
		Trigger (6) <ul style="list-style-type: none"> Action Button 	

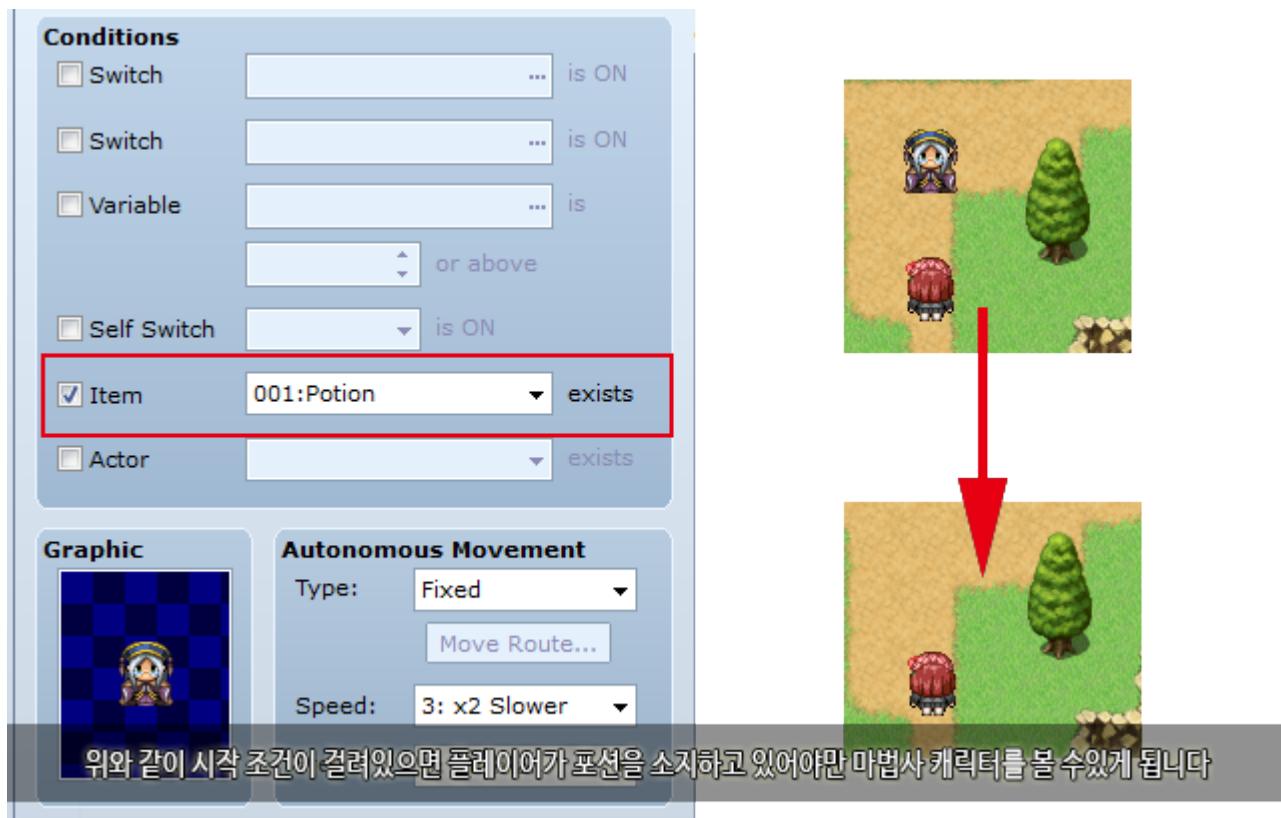
이번 강의는 위 스크린샷에 붙여진 번호 순으로 진행하겠습니다.

1. 이벤트의 시작 조건(Conditions)
2. 그래픽(Graphic)
3. 이동 조건(Autonomous Movement)
4. 옵션(Option)
5. 우선 순위(Priority)
6. 실행 조건(Trigger)

목록은 위와 같으며 시작 조건에 대해서 알아보도록 하겠습니다.

1. 이벤트의 시작 조건(Conditions)

이 부분에는 체크박스와 여러가지 형식의 리스트박스들이 위치하고 있습니다. 세부적으로 아직은 배우지 않은 항목이지만 시작 조건이라는게 어떤 기능인지 알아보겠습니다. 체크 박스의 많은 조건들 중 아무것도 체크하지 않으면 이벤트는 캐릭터는 게임이 시작되자마자 아래와 같이 맵에 보이는 상태로 존재하게 됩니다. 그러나 어떤 시작 조건이 걸려있다면 시작 조건이 맞지 않으면 캐릭터는 시야에서 보이지 않게 됩니다.



위에 체크박스만 보더라도 매우 간단하게 파악할 수 있는 기능입니다. 아직은 배우지 않았지만 셀프 스위치, 스위치, 변수와 같은 개념도 나중에 시작 조건으로 유용하게 사용할 수 있게 될 것입니다.

2. 그래픽(Graphic)

더블 클릭하면 캐릭터 파일들을 고를 수 있는 창이 뜹니다. 고를 수 있는 캐릭터가 상당히 많습니다. 여러가지 캐릭터들을 선택할 수 있는데요. 캐릭터를 하나 고르시면 되기 때문에 별로 설명을 드려야 할 부분이 없습니다. 선택할 수 있는 많은

캐릭터들은 쪼꾸르 툴에 기본적으로 포함되어 있는 데이터입니다. 쪼꾸르 툴에서는 이런 기본 소스가 있어서 게임을 쉽게 제작할 수 있습니다.

3. 이동 조건(Autonomous Movement)

이동 조건에는 이동 타입, 이동 루트의 설정, 이동 속도, 이동 빈도 등이 있습니다. 이동 타입을 보면 기본적으로 이동을 하지 않는 것으로 설정되어 있습니다. 타일 단위(그리드)로 이동합니다.

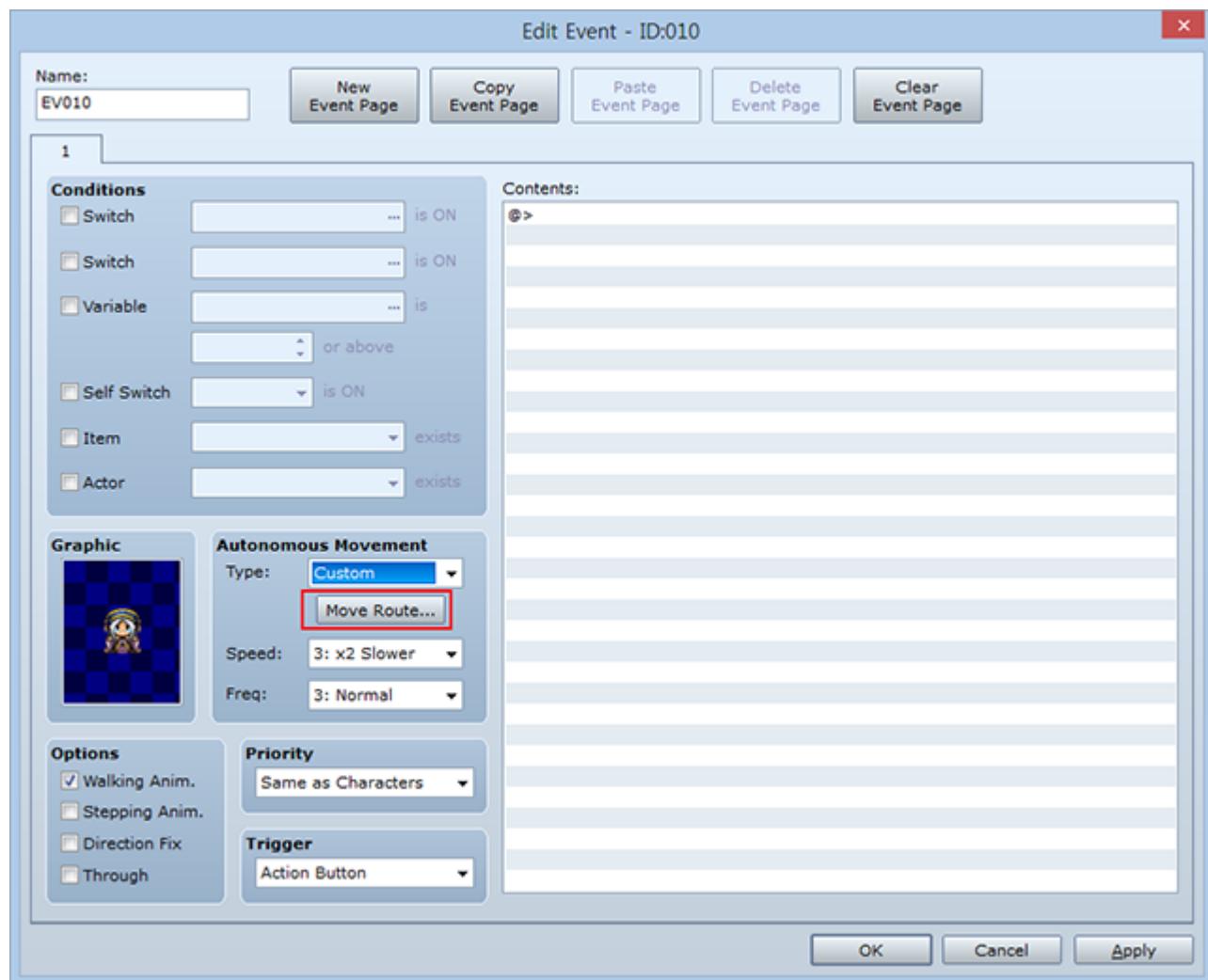


▲ 대표적으로 위의 2가지 이동 기능을 많이 사용합니다

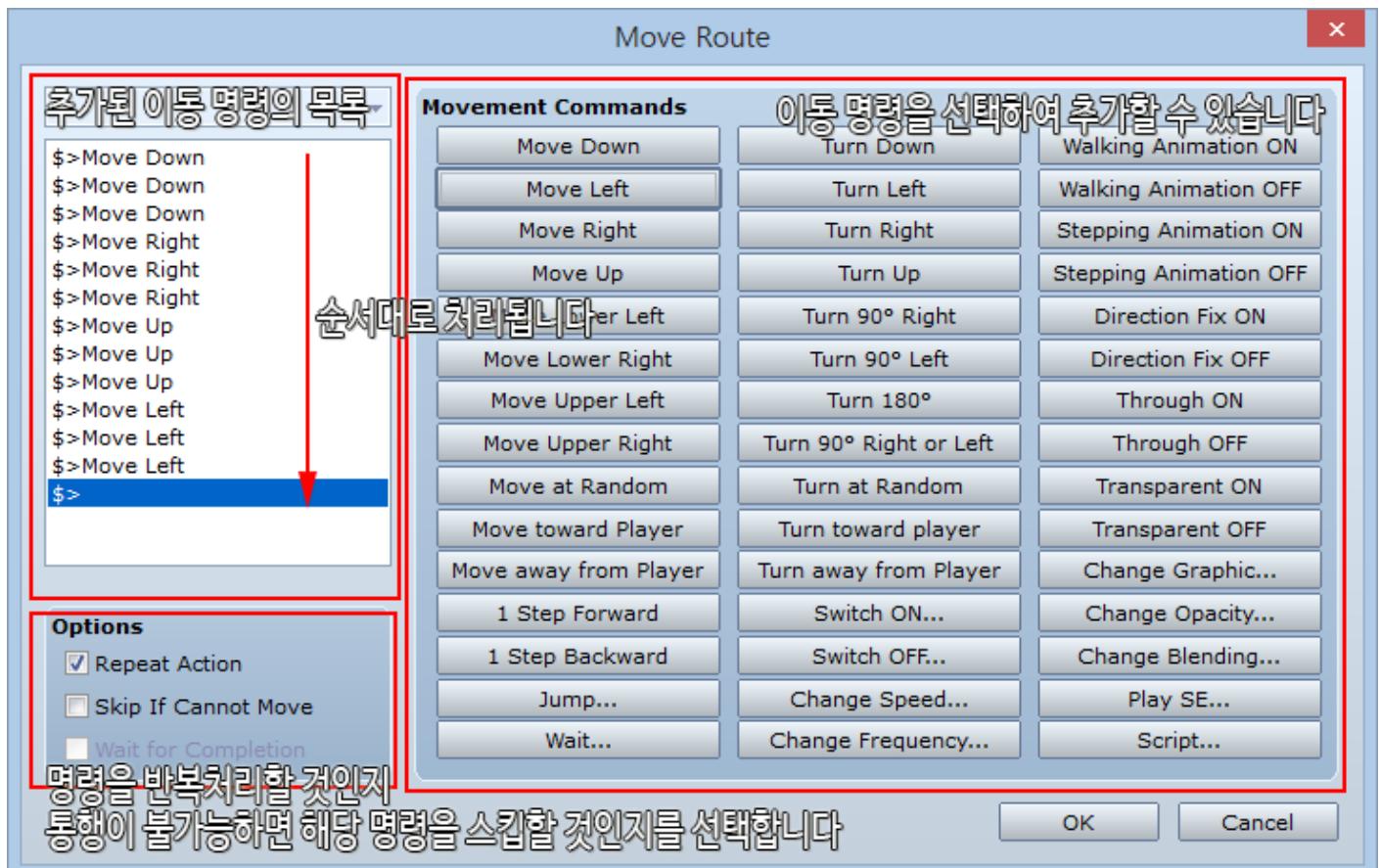
RPG Maker VX Ace는 타일 단위로 이동을 하는 시스템으로 '타일'에서 '타일'로 자동으로 건너가면서 이동 모션이 발생하게 되는데 그 다른 타일로 이동을 하는 속도를 이동 속도라고 합니다. 이동 속도(Speed)는 캐릭터가 다른 타일로 이동하는 속도를 나타내는 수치입니다. '오른쪽으로 한 칸 이동을 해라!'라는 명령이 떨어지면 캐릭터는 그 명령을 받고 오른쪽으로 한 칸 이동을 해야 하는데 이동 속도가 느리면 그 타일로 건너 가기 위한 이동 속도 자체가 느려집니다.



이동 빈도(Freq)는 이동 속도와는 달리 '오른쪽으로 한 칸 이동을 해라'라는 이동 명령의 처리 속도입니다. 캐릭터의 이동을 위한 명령들은 우리가 표를 예매하거나 식당에서 줄을 서고 있을 때의 줄처럼 어딘가에서 줄을 서며 대기 상태에 있습니다. 오른쪽으로 이동해라, 왼쪽으로 이동해라, 위쪽으로 이동해라와 같은 정보가 순서대로 쌓여있는데 이동 빈도가 높으면 짧은 시간에 더 많은 명령이 처리되면서 다른 타일로 위치를 옮겨가는 속도가 빨라집니다. 그러나 이동 빈도가 낮으면 명령이 처리되는 속도가 느려지면서 다른 타일로 이동되는 속도도 느려지게 됩니다.

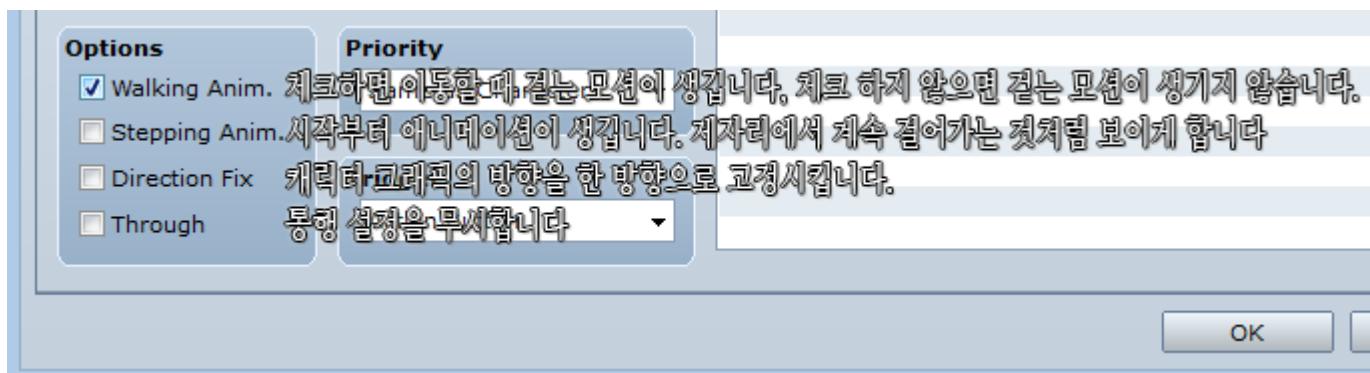


Custom은 제작자가 직접 이동 경로를 설정하는 기능입니다. 목록엔 램덤이나 플레이어에게 근접하는 기능만 있었지만 커스텀을 통해 이동 경로를 직접 만들 수 있습니다. 많은 이동 명령이 있어서 이 강의에 모두 포함하진 못하지만 몇 가지 특징만 소개하겠습니다.



4. 옵션(Option)

이벤트의 옵션은 캐릭터의 방향 고정, 통행 설정의 무시, 이동할 때 걷는 모션이 생기게 한다든가 처음부터 걷는 모션이 생기게 하는 등 부가 설정을 할 수 있는 기능입니다.



5. 우선순위(Priority)

플레이어와 이 이벤트에 설정된 그래픽과의 우선 순위를 말합니다. 우선 순위라는 것은 Z좌표를 기준으로 합니다. Z좌표라는 것은 2D 게임이지만 3D 게임처럼 바닥이 있다고 가정하면 바닥에서부터 얼만큼 떨어져있는지를 말합니다.

이 우선 순위는 플레이어를 기준으로 합니다. 플레이어의 Z좌표는 100입니다. 플레이어보다 Z좌표가 높으면 우선 순위가 높다고 말하고 낮으면 우선 순위가 낮다고 말하며 같으면 우선 순위가 같다고 표현합니다.



▲ 이벤트 우선순위를 이용하여 만든 다리(필드처럼 이용 가능)

Below Characters 는 우선 순위가 플레이어보다 낮은 캐릭터를 말합니다. Z좌표는 0입니다. 지형지물이 여기에 해당합니다.

Same as a Characters 는 우선 순위가 플레이어와 같은 상태를 말합니다. 플레이어와 같은 Z좌표로 설정합니다. 그래서 지나갈 수 없는 대상입니다. 사람이나 물건이 앞에 놓여 있으면 지나갈 수 없어야 정상이겠죠. 그럴 때 쓸 수 있는 우선 순위이며 엑스트라들이나 NPC들이 여기에 해당합니다.



▲ 우선 순위를 이용해 만든 박쥐(플레이어보다 높은 곳에 있습니다)

Above Characters 는 우선 순위가 플레이어보다 높은 상태를 말합니다. 플레이어보다 더 높은 위치인 Z좌표 200으로 설정합니다. 그래서 플레이어보다 더 높은 위치에서 그래픽이 그려집니다. 예를 들면 새라든지 지붕 등은 플레이어보다 더 높은 곳에서 위치해야 합니다. 그럴 때 설정할 수 있습니다.

6. 실행 조건(Trigger)

트리거는 이벤트의 실행 조건입니다. 캐릭터 그래픽을 설정하면 적절한 것으로 자동 선택되어집니다.

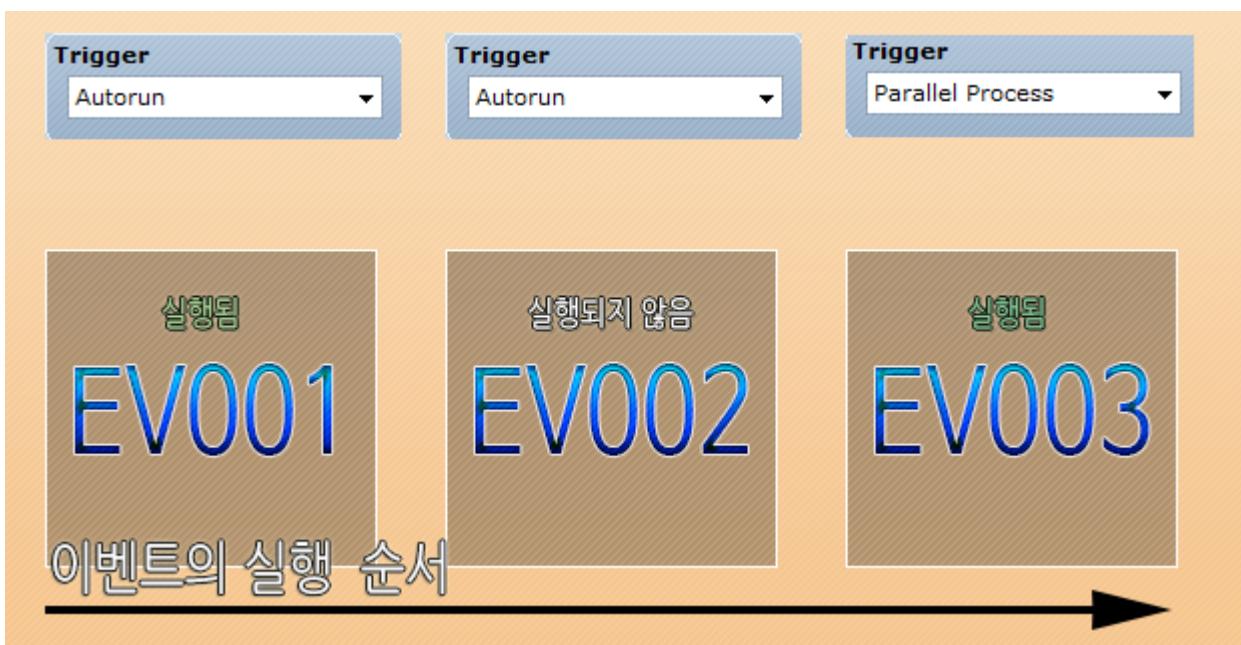
Action Button 은 결정키로 시작하는 방법입니다.



▲ 문 이벤트는 Player Touch 트리거로 플레이어가 터치해야 문이 열립니다

Player Touch 는 플레이어가 먼저 이벤트 쪽으로 다가가면 이벤트가 실행되는 방식입니다.

Event Touch 는 그 반대로 이벤트가 먼저 플레이어에게 다가갔을 때 실행되는 방식입니다.



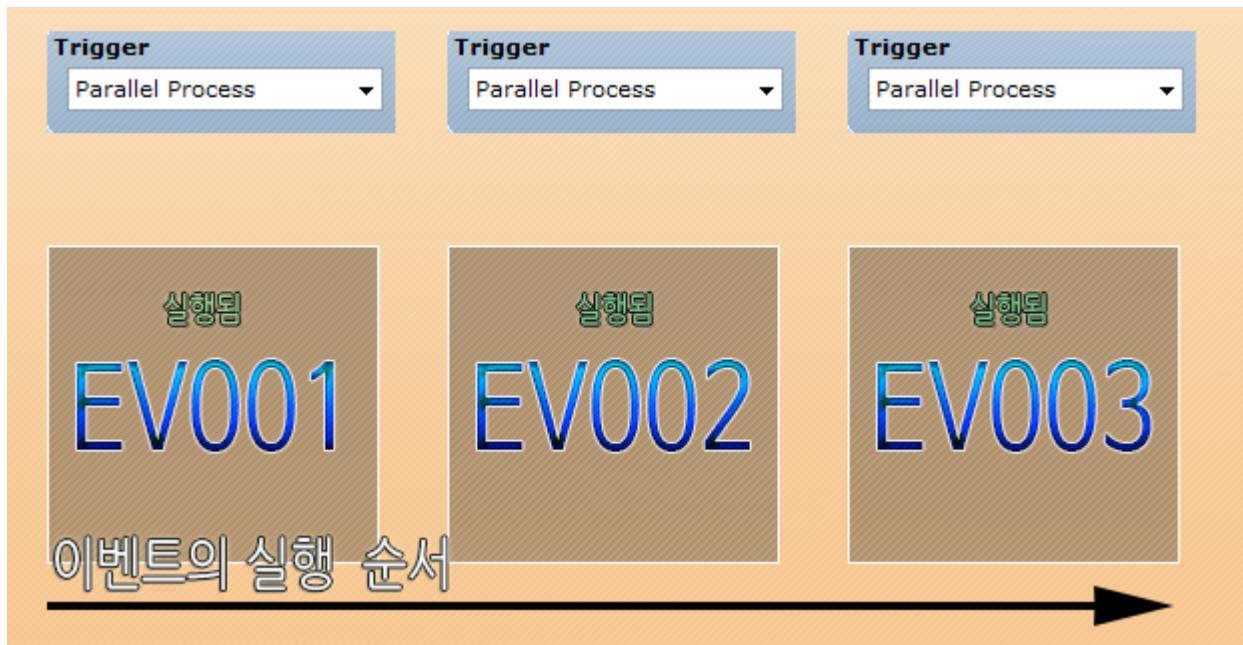
▲ 하나만 실행되는 Autorun(자동) 이벤트

Autorun(자동실행)은 처음부터 자동으로 실행되는 이벤트입니다. 이 자동 이벤트가 실행 중이면 플레이어가 멈추게 됩니다. 또한 트리거가 자동 실행, 병렬 처리가 아닌 Action Button, Player Touch, Event Touch 로 트리거가 설정된 이벤트들은 모든 움직임이 정지되며 락 상태에 빠집니다.

원래 이벤트는 매 프레임에 하나의 이벤트만 실행됩니다. 자동 이벤트는 매 프레임마다 실행되는 이벤트입니다. 그래서 변수 하나가 설정되어 있다하더라도 그것만 수행하느라 플레이어까지 멈춘 상태로 있게 되는 것입니다. 왜냐하면 게임에서 이 이벤트만 신경쓰고 있기 때문입니다. 그래서 자동 이벤트를 실행하고 종료하지 않으면 모든 캐릭터가 멈춘 상태로 있게 될 것입니다.

따라서 이 특징에 따라 자동 이벤트는 2개이상 설정될 수 없습니다. 설정했다하더라도 하나는 위와 같이 동작하지 않는 상태로 존재하게 됩니다. (2개 이상의 자동 이벤트가 실행되고 있다면 더 늦게 만들어진 자동 이벤트의 내용은 실행이 되지 않습니다. 그러나 움직임은 락 상태가 아니기 때문에 자동 이벤트로 만든 캐릭터는 움직임이 가능한 상태로 있습니다.)

그러나 다음 트리거인 병렬 처리(Parallel Process)는 별도로 실행되며 처리됩니다. 아래 스크린샷 보면 병렬 처리 이벤트의 특징을 알 수 있습니다.



▲ 병렬 처리 이벤트의 처리 상태

Parallel Process 는 병렬 처리라는 것으로 기본적으로 자동 실행되는 Autorun 이벤트와 같지만 차이점은 다른 이벤트에 영향을 주지 않는다는 점입니다.

대화 이벤트를 제외한 어떤 명령이 실행되고 있어도 플레이어를 비롯한 NPC와 캐릭터들이 멈추지 않습니다. 또한 병렬로 처리된 이벤트가 2개 이상 있다하더라도 하나가 멈추지 않습니다.

엄밀히 말하면 동시에 실행하는 것은 아닙니다. 알만툴에서는 이벤트를 빠른 이벤트부터 마지막 이벤트까지 순서대로 처리를 합니다. 컴퓨터에서는 거의 모든 것이 그렇습니다. 하지만 빠른 처리로 동시에 처리되는 것처럼 보일 뿐입니다.

병렬 처리도 마찬가지로 병렬 처리를 하게 되면 번갈아가면서 이벤트도 빠른 속도로 실행하여 처리하기 때문에 모든 병렬 이벤트가 동시에 실행되고 있는 것처럼 보이게 됩니다.

병렬 처리를 하면 병렬 이벤트가 실행되고 있는 도중에 다른 이벤트도 처리할 수 있습니다.

[04강] 스위치와 셀프스위치 - RPG 만들기 VX Ace 강좌

시작하기

2014/10/31 00:10

<http://blog.naver.com/biud436/220166911677>

RPG Maker VX Ace 왕초보탈출 4

"스위치와 셀프스위치"

스위치(Switch)에 대한 이론

현실 세계에서의 스위치는 무조건 두 가지의 값을 가지고 있습니다. 켜다(ON)와 끈다(OFF)를 나타내는 두 가지 상태가 존재합니다. 깜깜한 방 안에 따뜻한 불빛을 녹아들게 하려면 OFF 상태에 있는 스위치를 눌러서 ON으로 바꿔줘야 방 안에 빛이 생김을 여러분들은 이미 많이 해봐서 잘 알고 있습니다. 여러분들은 이미 스위치에 대해서 너무나 잘 알고 있습니다. 그래서 예를 들어보겠습니다.

전등 스위치를 켜면 전등이 켜져 방안에 불빛이 생기고, 끄면 전등이 꺼져서 방안이 어두워진다

텔레비전의 전원 스위치를 켜면 텔레비전이 켜지고, 끄면 텔레비전이 꺼진다

전자레인지의 가동 스위치를 켜면 전자레인지가 가동되기 시작하여 음식이 맛있게 데워진다. 스위치를 켜지 않았다면 아무일도 일어나지 않는다

게임 이벤트에서도 마찬가지로 스위치를 통해 어떤 일을 발생시킬 수 있습니다.

스위치1이 켜진 상태라면 체력이 회복되고, 꺼진 상태라면 아무일도 일어나지 않는다

스위치2가 켜진 상태라면 대화창이 더 이상 표시되지 않고 꺼진 상태라면 대화창이 표시된다

현실에서도 스위치를 통해서 어떤 일을 행하거나 하지 않거나 두 가지 상태를 스위치를 통해 컨트롤을 합니다. 즉, 스위치가 ON상태 일때 일어나는 일과 스위치가 OFF 상태일 때 일이 다릅니다.

초록색 글씨를 보시면 알겠지만 스위치에는 이름이 있었다는 것을 알 수 있습니다.

전등 스위치, 텔레비전 전원 스위치, 전자레인지의 가동 스위치처럼 말입니다. 이와 같이 모든 스위치에는 스위치를 식별할 수 있는 고유한 이름이 붙어 있습니다. 이것은 우리가 하게 다루게 될 이벤트의 스위치에도 적용되는 개념입니다.

이벤트에서는 스위치가 켜진 상태를 ON이라고 하고, 꺼진 상태를 OFF라고 부릅니다.

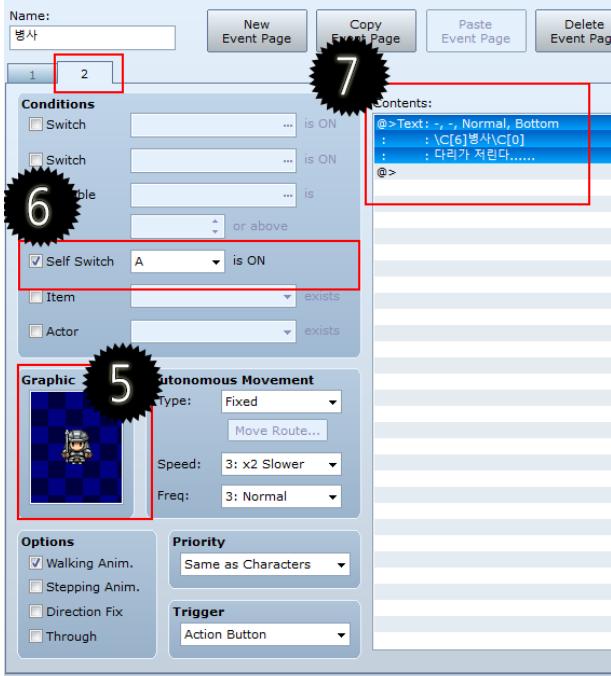
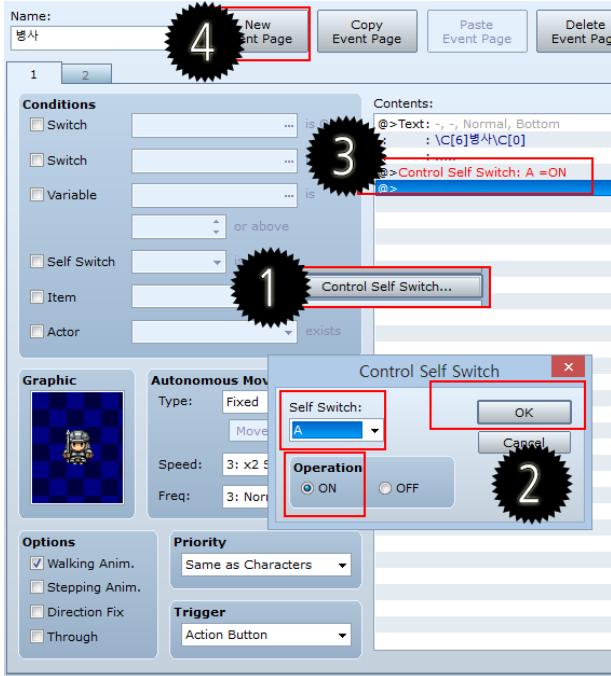
스위치에는 1-100 까지의 숫자가 부여되어있는데 이 숫자들이 스위치를 식별할 수 있는 스위치의 이름입니다.

이렇게 모든 스위치에는 이름이 붙어 있습니다.

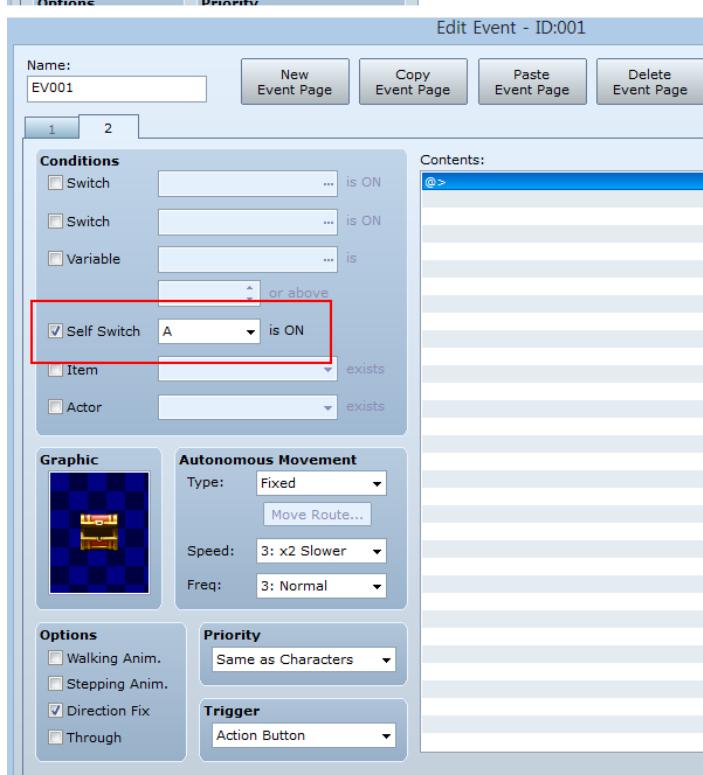
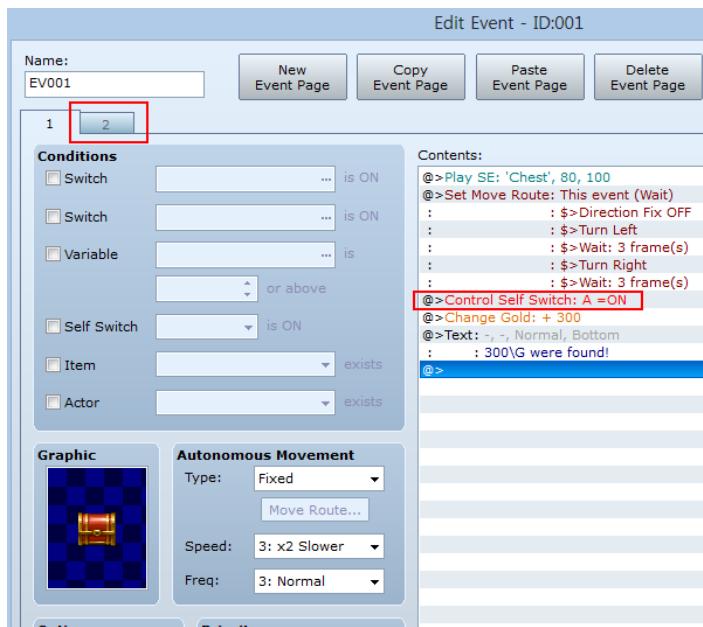
이 스위치의 이름만 알고 있으면 그 어느 맵에서도 스위치를 켜고 끄면서 흐름을 분기할 수 있다는 게 스위치의 가장 큰 특징중 하나입니다.

셀프 스위치에 대한 이론과 예제

셀프 스위치는 위의 스위치와 같은 개념이지만 사용할 수 있는 범위가 '현재 이벤트'에게만 한정되어있는 스위치입니다. 그래서 현재 이벤트 내에서 일어나는 일을 조건에 의해 두 가지 이상으로 바꿔주고자 할 때 유용하게 쓸 수 있는 기능입니다. 셀프 스위치는 매우 간단하기 때문에 예제를 통해 바로 실습해보도록 하겠습니다.



▲ 한 번 말을 걸면 대사가 바뀌는 엑스트라 캐릭터 이벤트



▲ 한 번 열리면 보물상자에서 더 이상 골드를 획득할 수 없는 보물상자 이벤트(Ctrl+3 으로 생성 가능)

이벤트는 이벤트 자신에게 쓸 수 있는 A, B, C, D 라는 이름을 가진 셀프스위치라는 것을 가지고 있습니다. 이렇게 셀프스위치는 현재 편집중인 이벤트에서만 사용할 수 있는 스위치입니다. 이벤트의 편집만으로 현재 이벤트의 셀프스위치의 값을 다른 이벤트에서 조작할 수 없습니다.

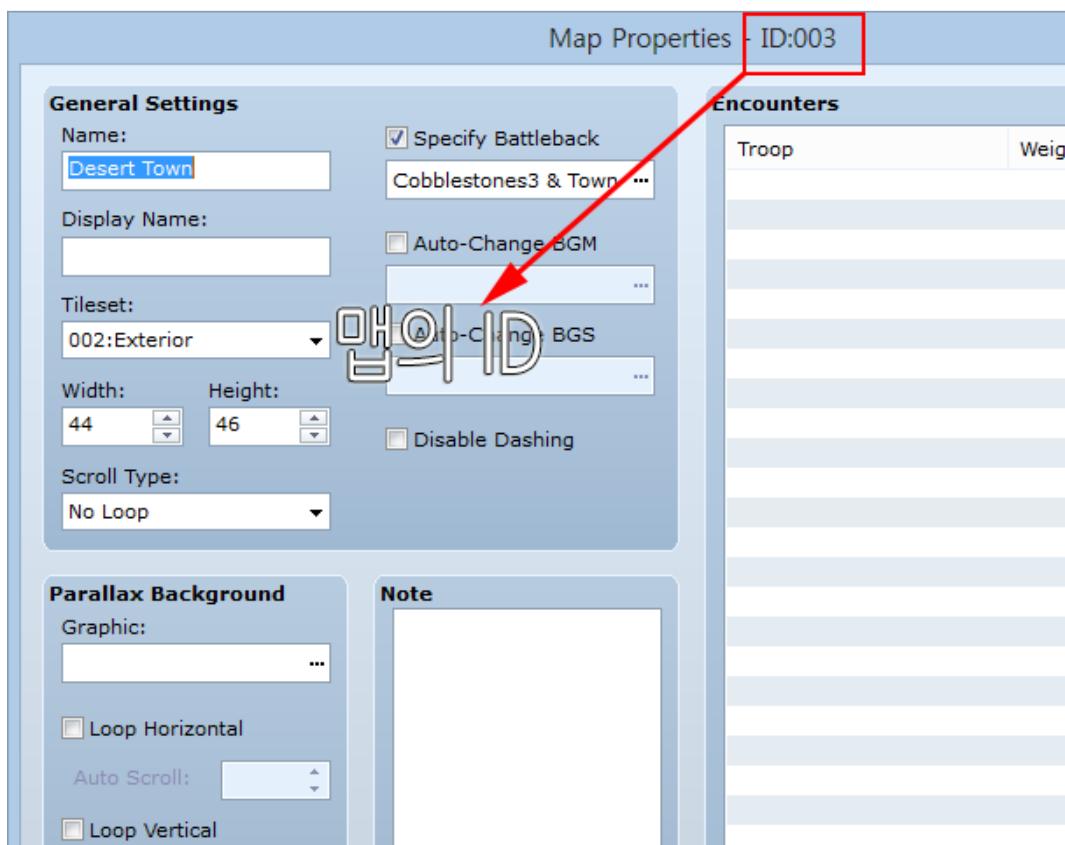
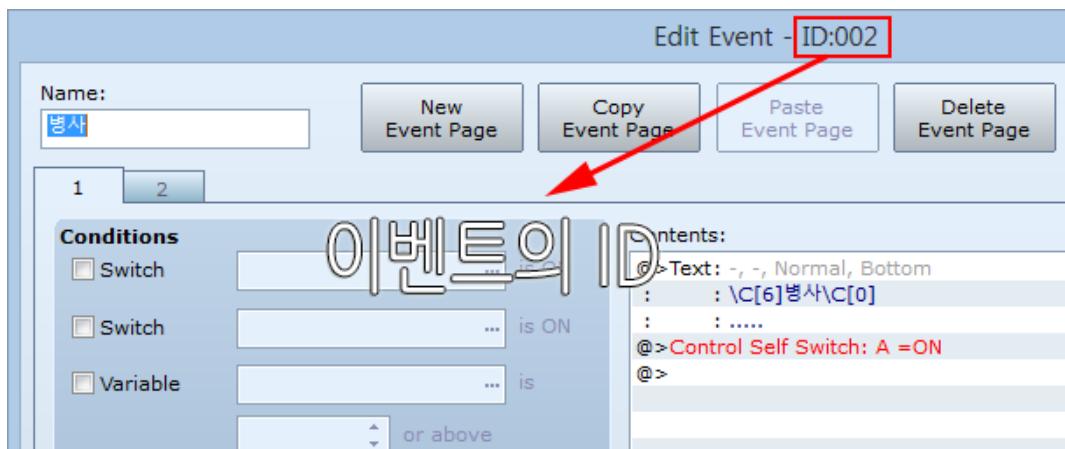
앞에 이론에서는 이름만 알고 있으면 어느 맵에서도 스위치를 켜고 끌 수 있다고 설명했습니다. 여러분들은 A,B,C,D 라는 이름이 셀프스위치의 이름이라고 여기고 있을 겁니다. 하지만 다른 이벤트들도 A,B,C,D 라는 이름을 가진 셀프스위치를 가지고 있습니다. 그런데 실험을 해보면 알겠지만 서로 전혀 다른 셀프 스위치들라는 것을 알 수 있습니다. 이름이 같지만

서로 다른 진짜 이유는 무엇일까요? 이론 설명에서는 이름만 가지고 있으면 조작이 가능하다고 했는데 조작이 불가능하다는 것이 확인되었습니다.

설명이 잘못된 걸까요? 아닙니다.

이렇게 각자만의 셀프 스위치를 가질 수 있는 이유는 셀프스위치의 진짜 이름이 단순 A, B, C, D 가 아니기 때문입니다.

셀프스위치의 진짜 이름에는 현재 맵의 ID, 현재 편집중인 이벤트의 ID까지도 셀프스위치의 이름으로 포함됩니다.



▲ 이벤트 ID와 맵의 ID를 확인하는 방법

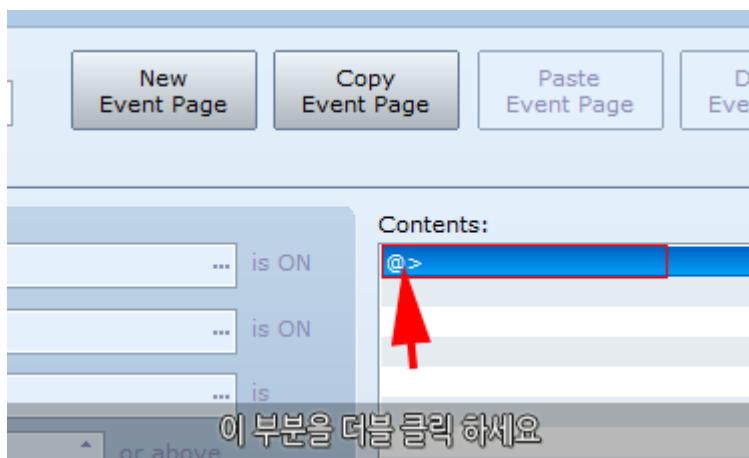
정확히 따지자면 셀프스위치의 이름은 셀프스위치[맵의 ID, 이벤트의 ID,'셀프스위치의 식별 문자']입니다. 따라서 병사이벤트의 경우 단순히 셀프스위치 A를 ON으로 바꿔준 것이 아니라 셀프스위치[3,2,'A']의 값을 ON으로 바꿔준 것입니다. 맵의 ID와 이벤트의 ID는 셀프스위치를 조작할 때 자동으로 이름으로 추가됩니다. 이렇게 되면 결국 같은 이름을 가진 셀프스위치는 하나도 없게 됩니다.

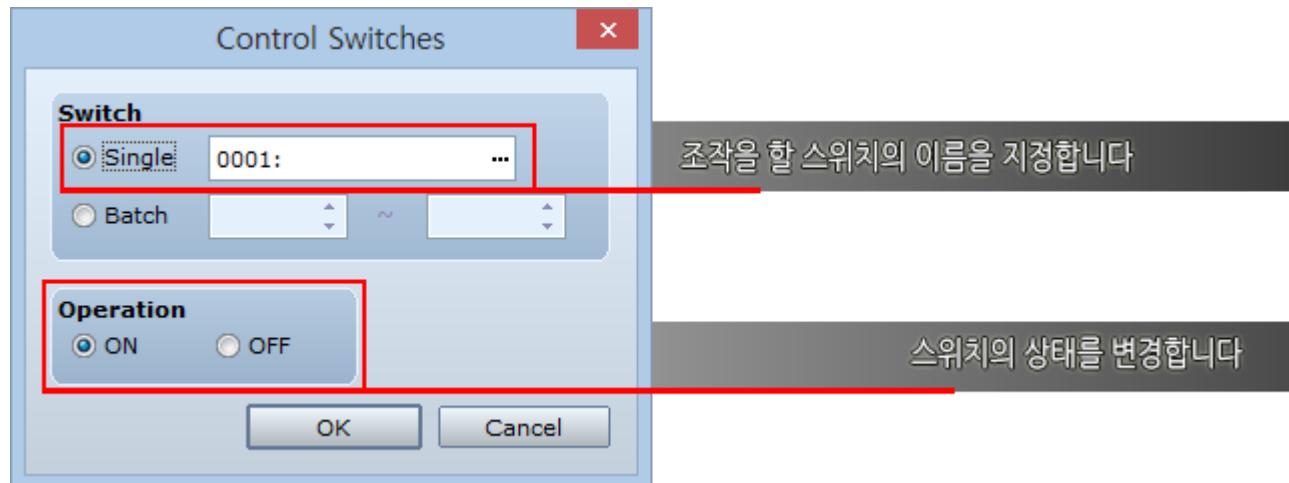
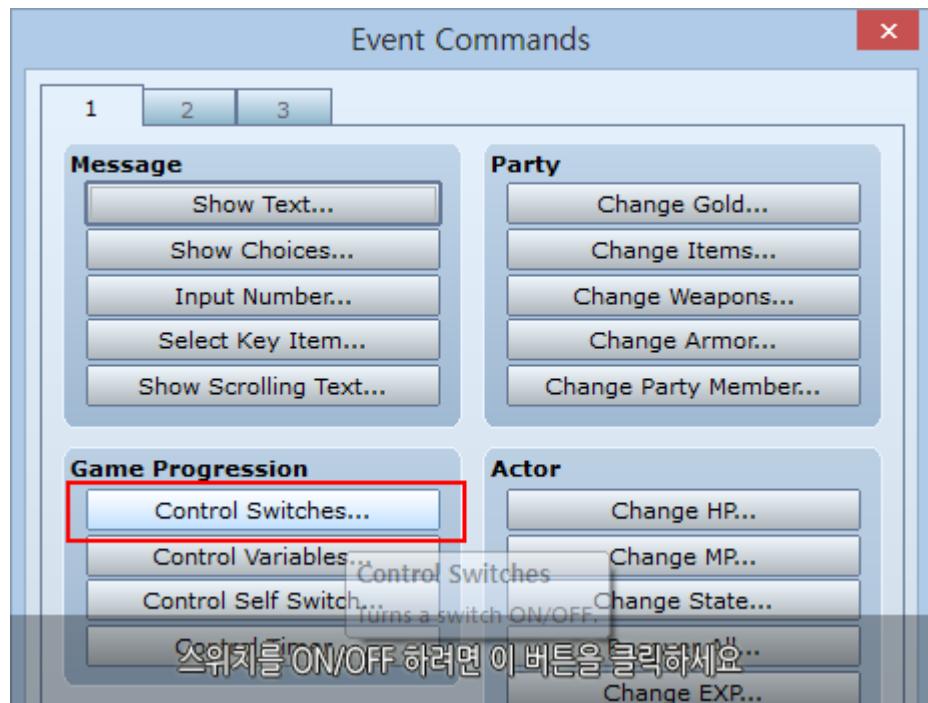
이렇게까지 설명하는 이유는 모든 이벤트엔 셀프스위치 A, B, C, D가 있지만 실상은 서로 다른 셀프 스위치이며 개별적인 값을 가지고 있다는 사실을 인지시켜드리기 위해서입니다. 셀프스위치는 스위치의 사용 범위가 현재 이벤트 내로 극히 제한되기 때문에 단순 대화 이벤트, 금전을 획득하고 다시 획득할 수 없는 보물 상자 이벤트 등에만 사용할 수 있는 스위치입니다.

스위치 사용 예제

일반 스위치는 셀프 스위치와는 달리 이름 하나만 알면 다른 이벤트에서도 사용할 수 있는 스위치입니다. 이 일반 스위치를 통해 퀘스트 같은 복잡한 이벤트나 시스템들도 만들 수 있습니다.

스위치는 RPG Maker VX Ace에 있는 개념으로 이 스위치는 ON(참) 또는 OFF(거짓) 중 하나의 값만 가지고 있습니다. 모든 스위치는 처음에 OFF(거짓) 상태에 있습니다. 예제를 보겠습니다.



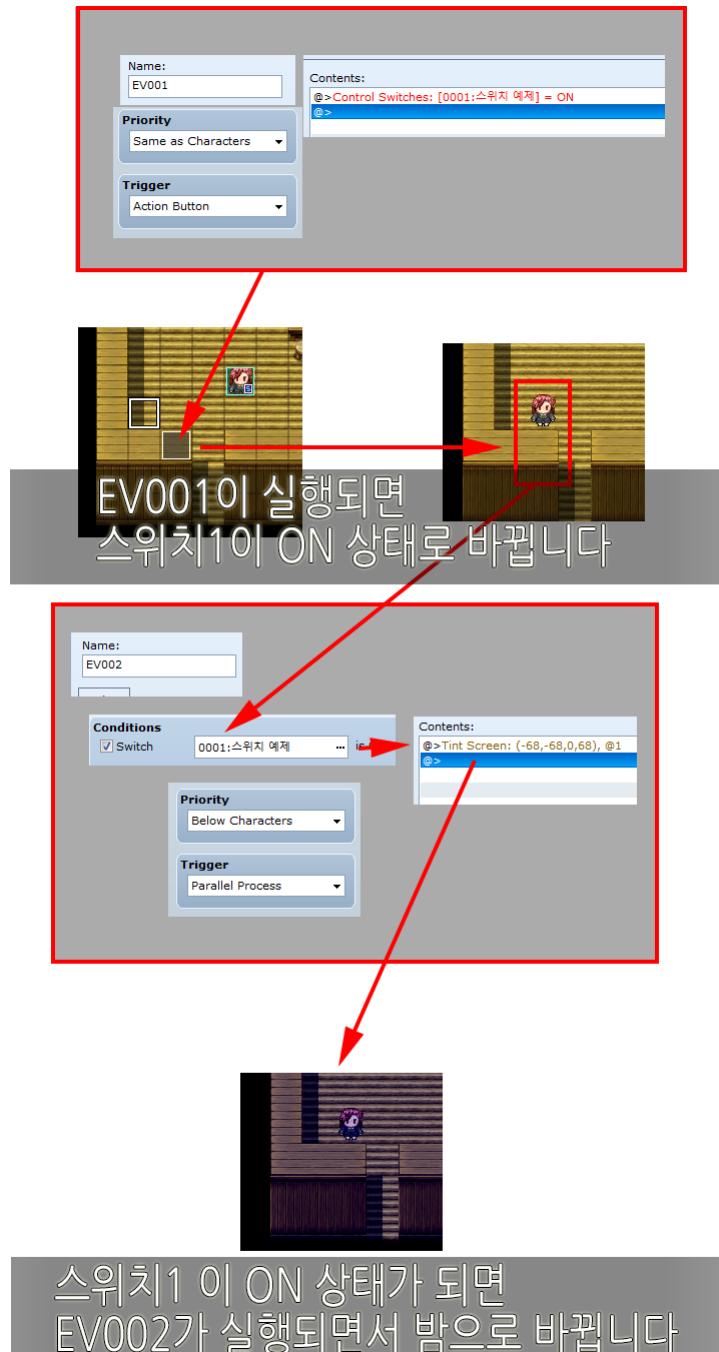


일반 스위치는 숫자 이외에 이름도 따로 지정할 수 있습니다. 고유의 이름은 왼쪽에 붙어있는 숫자가 진짜 이름입니다만 숫자는 기억하기 힘들기 때문에 문자열도 이름으로 지정할 수 있습니다. 숫자만으로는 나중에 제어할 스위치가 너무 많아 지면 햇갈려서 나중에 스위치를 찾기 힘들어질지도 모르기 때문입니다. 따라서 RPG Maker VX Ace에서는 숫자 이외에 도 문자열도 같이 지정할 수 있는 기능을 만들어놓았습니다.

그래서 나중에 해당 스위치가 어떤 기능이 있는지 저장했던 당시 문자열로 유추해볼 수 있게 되는 것입니다. 나중에 어떤 스위치인지 찾기 편하도록 말이죠. 다른 이벤트에서도 스위치의 고유한 이름을 알고 있다면 어느 맵이든 언제든지 스위치의 상태를 조작할 수 있기 때문에 이를 이용하여 스토리라인을 연결하거나 퀘스트를 만들거나 게임의 흐름을 바꿔놓을 수도 있습니다.

위와 같은 스위치의 조작을 통해 ON(참) 상태로 변경해주면 OFF에서 ON으로 상태가 바뀝니다. 위 그림에서는 1번 스위치의 값을 ON으로 바꿔줬습니다. 모든 일반 스위치들은 고유의 이름을 가지고 있기 때문에 어떤 값을 가지고 있는지 외부에서 체크할 수 있습니다. 그래서 1번 스위치의 값을 ON으로 바꿔준다면 다른 이벤트에서도 이를 활용할 수 있습니다.

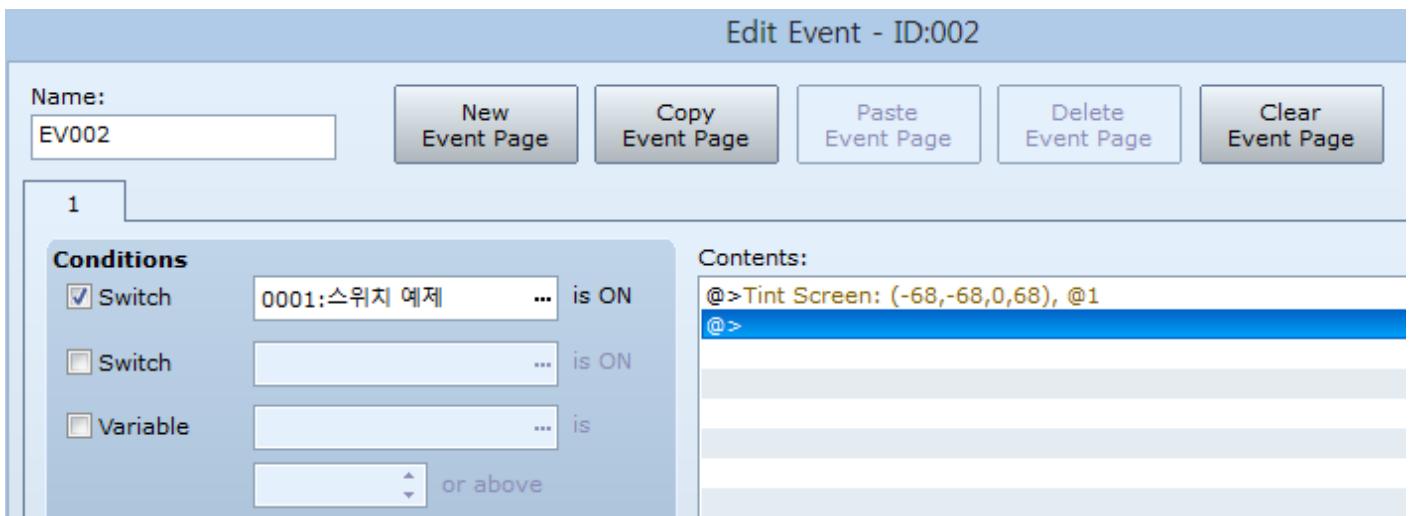
이 스위치의 이름을 이용하여 값을 체크하고 이를 응용해보겠습니다.



▲ 이벤트 1에서 스위치1를 작동시켰더니 이벤트2 가 실행되었습니다.

위 예제는 이벤트 1에서 스위치를 작동시킨 후에 다른 이벤트에서 방금 작동시킨 스위치의 값을 감지하여 화면의 색조를 변경하고 있는 모습이었습니다.

이렇게 스위치는 이벤트의 위치와 상관없이 다른 이벤트에서도 사용할 있습니다. 다시 한 번 EV002 를 살펴보겠습니다.



▲ 화면의 톤을 어두운 톤으로 변경합니다

위 이벤트는 병렬 처리를 통해 화면의 색조를 어둡게 변경하는 이벤트입니다. 그러나 시작 조건이 스위치 1번이 ON 상태이어야 하기 때문에 스위치 1번을 ON 상태로 바꿔줘야 합니다. 이 스위치가 ON 상태이면 시작되는 이벤트를 만든 것입니다. 다른 이벤트에서 이 1번 스위치의 값을 OFF에서 ON 상태로 바꿔줬으므로 위 예제가 성공적으로 동작하게 됩니다. 따라서 예제 이벤트의 실행에 의해 화면이 아래처럼 밤으로 변하게 됩니다.



이렇게 일반 스위치는 현재 이벤트에서 다른 이벤트로 그 흐름을 이어가기 위해 쓰인다는 것을 알 수 있습니다. 그러나 셀프스위치는 현재 이벤트에서 다른 이벤트로 흐름을 이어가진 못하며 자신에게만 사용할 수 있다는 사실도 알 수 있습니다.

그러나 이 스위치만으로 게임을 만들기엔 턱없이 부족합니다. 게임을 만드려면 더 많은 것이 필요하고 더 많은 것을 배워야 합니다. 조건에 따라 이벤트를 분기할 수도 있어야 하고 값을 어디가에 저장하기도 해야 퀘스트 같은 복잡한 이벤트를 만들 수 있습니다. 이런 내용들도 추후 강좌에서 배워보도록 하겠습니다.

[05강] 변수와 조건분기 - RPG 만들기 VX Ace 강좌

시작하기

2014/11/01 00:46

<http://blog.naver.com/biud436/220167979612>

RPG Maker VX Ace 왕초보탈출 5

"변수와 조건분기"

변수의 정의

변수라는 것은 숫자를 보관하기 위한 공간입니다. 이 공간에는 숫자로 되어있는 데이터들이 저장되어있습니다. 변수도 스위치와 마찬가지로 고유의 이름이 있기 때문에 나중에 필요할 때 이 이름을 이용하여 어떤 값이 저장되어있는지 확인할 수도 있고 새로운 값을 저장할 수도 있습니다. 변수에 저장된 값은 맵 어디에서든 참조할 수 있습니다.

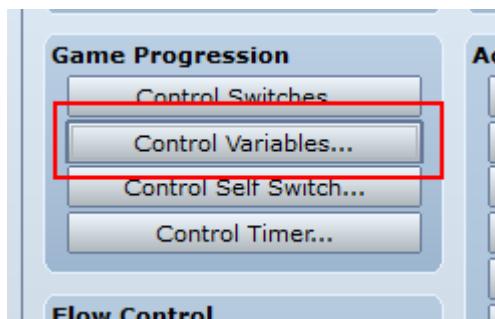
변수의 범위

이 시간에 배우게 될 기본 변수들 범위는 RPG Maker VX Ace 의 모든 공간에서 사용할 수 있는 변수입니다. 이러한 변수는 프로그래밍 관점에서는 전역 변수라고 부릅니다. 변수를 사용할 수 있는 범위에 대해서 설명하고 있는 이유는 RPG Maker VX Ace 에서 제공하는 기본 변수는 맵 에디터, 제작자들이 만든 모든 맵, 존재하는 모든 이벤트, 모든 데이터베이스에서 사용이 가능하다는 것을 설명하기 위해서입니다.

강의 초기부터 이렇게 프로그래밍과 스크립트가 자주 언급되는 이유는 RPG Maker VX Ace 는 모든 것이 루비라는 스크립트 언어로 되어있기 때문에 스크립트와 전혀 무관하지 않기 때문입니다. 하지만 이 단계에서 아직은 적절하지 않기 때문에 조금씩만 언급해나가도록 하겠습니다

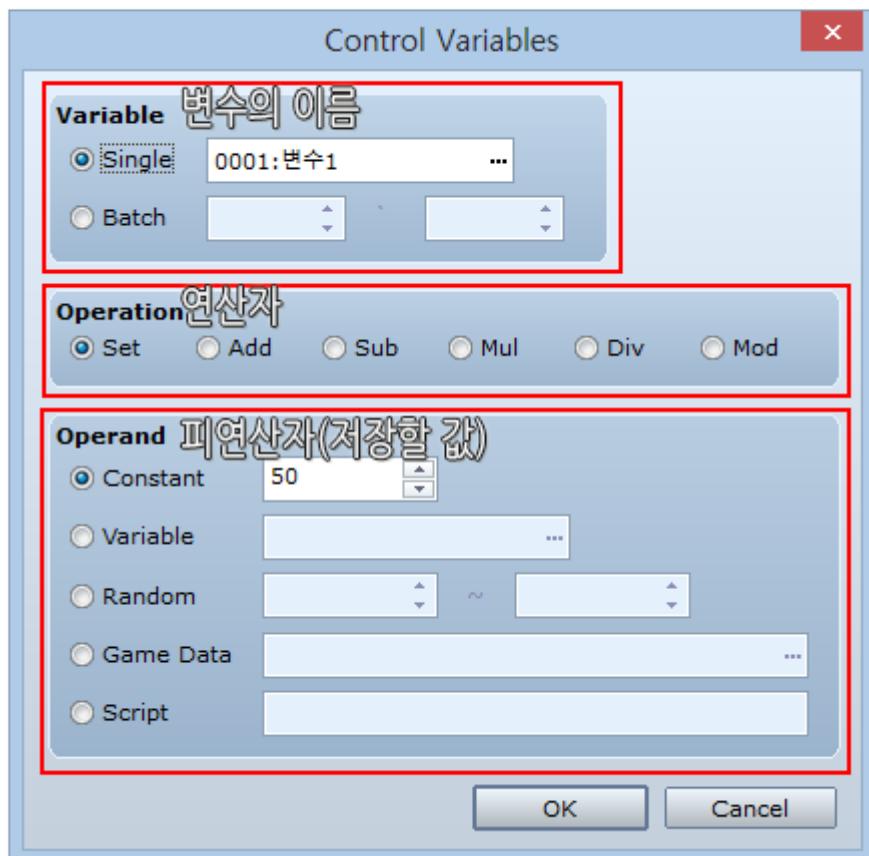
살펴보기

변수에 숫자를 저장하고 사칙연산까지 실습에 들어가며 바로 해보겠습니다. 더하기, 빼기, 곱하기, 나누기를 하는 방법은 자체는 여러분들이 이미 알고 있는 부분이기 때문에 변수의 조작창이 어떻게 구성되어있는지 단순하게 알아보기만 하겠습니다.



▲ 변수의 조작

변수는 게임의 흐름을 결정할 수 있는 전역 변수이며 중요하기 때문에 이벤트 커맨드 첫번째 페이지에 위치하고 있습니다. 위와 같이 되어있는 부분입니다.



변수를 조작할 수 있는 창으로 사칙연산을 할 수 있습니다. 연산자와 피연산자를 선택하여 변수에 값을 저장하는 역할을 합니다.

Variable 부분은 조작할 변수를 선택하고 이름을 바꾸는 부분입니다. 변수 하나를 선택하여 이름을 변경하거나 Batch 를 통해 일괄 선택을 할 수도 있습니다.

Operation 부분은 연산자를 선택하는 부분입니다. 왼쪽부터 대입(Set), 덧셈(Add), 뺄셈(Sub), 곱셈(Mul), 나눗셈(Div), 나눗셈을 할 때 몫을 구한 후에 남는 수, 즉 나머지값(Mod) 입니다.

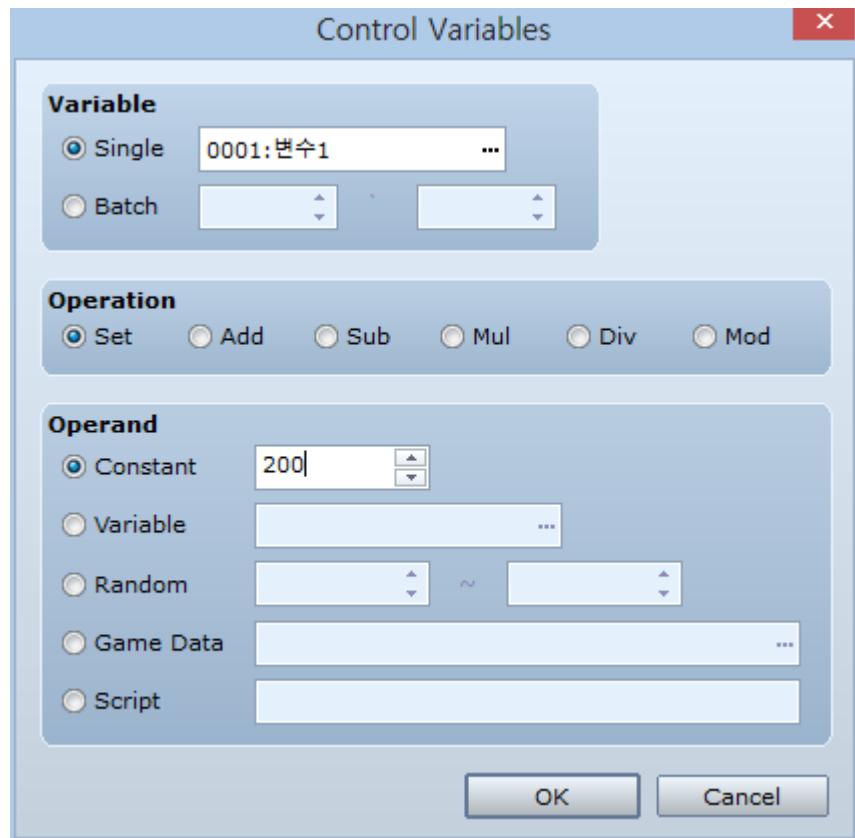
Operand 는 피연산자 부분입니다. 피연산자는 연산의 대상이 되는 값들을 골라 넣을 수 있습니다. 게임데이터를 불러올 수도 있습니다. 물론 모든 것은 숫자로 되어있습니다.

본론으로 들어가기 전에 간단하게 인터페이스를 살펴봤습니다. 물론 인터페이스가 간단하기 때문에 별다른 설명을 읽지 않아도 이해가 되는 부분이 있을 것입니다. 하지만 설명을 하지 않으면 실감하지 못하는 부분도 있습니다. 그런 부분에 대해서 지금부터 자세하게 설명을 하도록 하겠습니다.

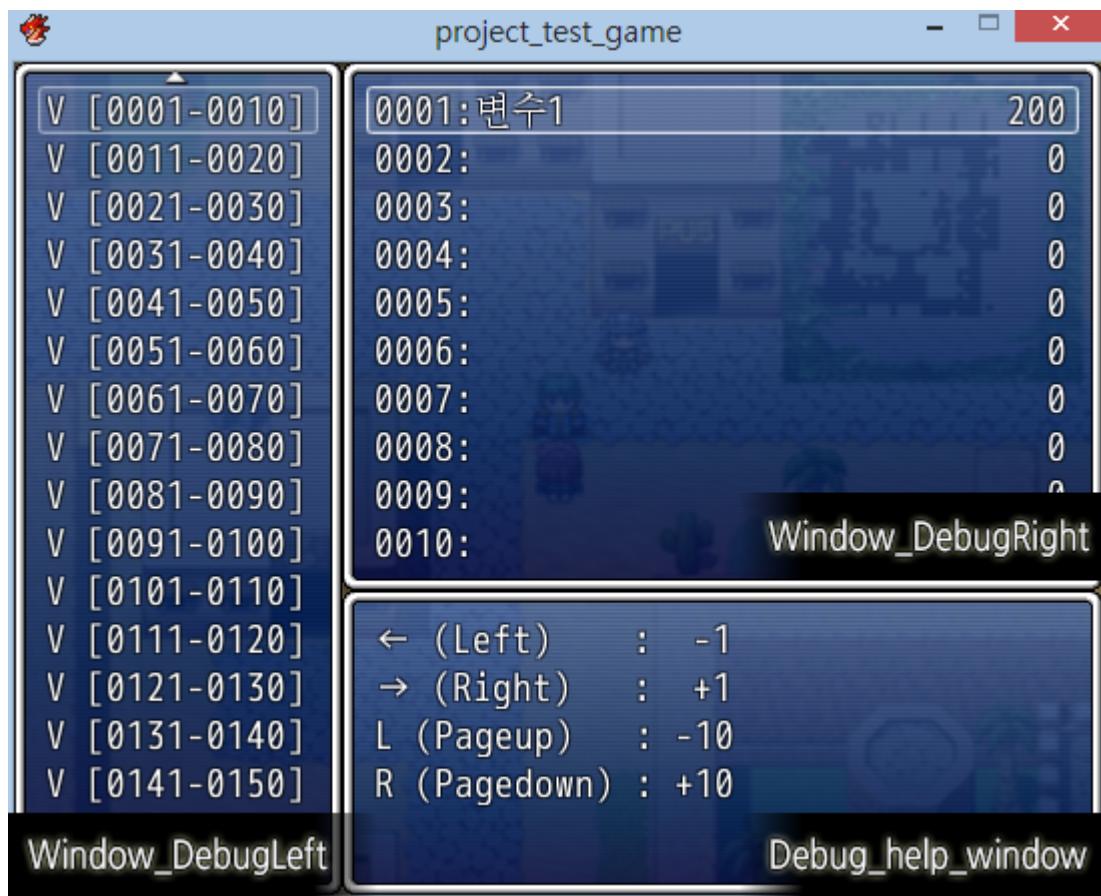
물론 간단한 튜토리얼부터 시작해보도록 하겠습니다.

변수에 숫자를 대입하면 값을 저장할 수 있다

변수는 이론보다 실습이 중요합니다. 그래서 우선 캐릭터 하나를 만들었습니다. 이 캐릭터는 아래 변수에 200 이라는 값을 대입합니다. 그 외에 하는 일은 없습니다. 대화창을 띄우지도 않고 그저 변수만 저장합니다.



이 캐릭터가 정말 200이라는 값을 변수1에 저장을 했는지 확인해보겠습니다. 변수값은 디버그 창으로 확인해보도록 하겠습니다. 이 디버그 창은 게임 내 스위치와 변수의 값들이 나열된 곳으로 F9를 누르면 열 수 있으며 Q, W로 행을 건너뛸 수 있습니다.

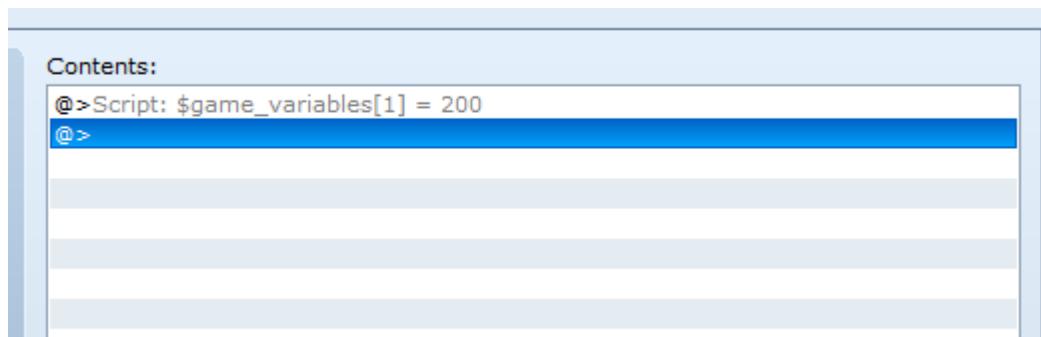


V는 Variables의 약자입니다. Window_DebugRight(오른쪽 윗 부분)을 보면 V0001에 200이라는 숫자가 성공적으로 저장되었다는 것을 알 수 있습니다. 대입을 하면 변수에 숫자가 그대로 저장이 된다는 것을 알 수 있습니다.

```
# $game_variables[변수의 번호] = 숫자
```

```
$game_variables[1] = 200
```

변수는 스크립트에서 표현하는게 더 간단할 수 있습니다. 변수는 전역 변수이기 때문에 전역 변수임을 뜻하는 \$(달러) 표시가 붙습니다. 다시 한 번 말하지만 변수는 전역 변수이기 때문에 어디에서든 값을 저장하고 읽고 쓰는게 가능합니다.



▲ 스크립트에서도 변수를 사용할 수 있습니다.

스크립트에서의 변수의 숫자 대입은 위와 같이 표현됩니다. 이런 스크립트 언어는 이벤트보다 더 직관적이라는게 장점입니다. 변수에 200이 대입되었다는 것은 누구나 직관적으로 알 수 있습니다.

The screenshot shows the RPG Maker VX Ace script editor. On the left, there's a list of variables: S [0001-0010], S [0011-0020], S [0021-0030], S [0031-0040], S [0041-0050], S [0051-0060], S [0061-0070], S [0071-0080], S [0081-0090], S [0091-0100], S [0101-0110], V [0001-0010], and V [0011-0020]. The variable V [0001-0010] is selected and highlighted with a blue border. On the right, a list of variables and their values is shown:

0001:변수1	200
0002:	0
0003:	0
0004:	0
0005:	0
0006:	0
0007:	0
0008:	0
0009:	0
0010:	0

▲ 스크립트를 통해서 변수를 조작해도 값은 동일하게 출력됩니다

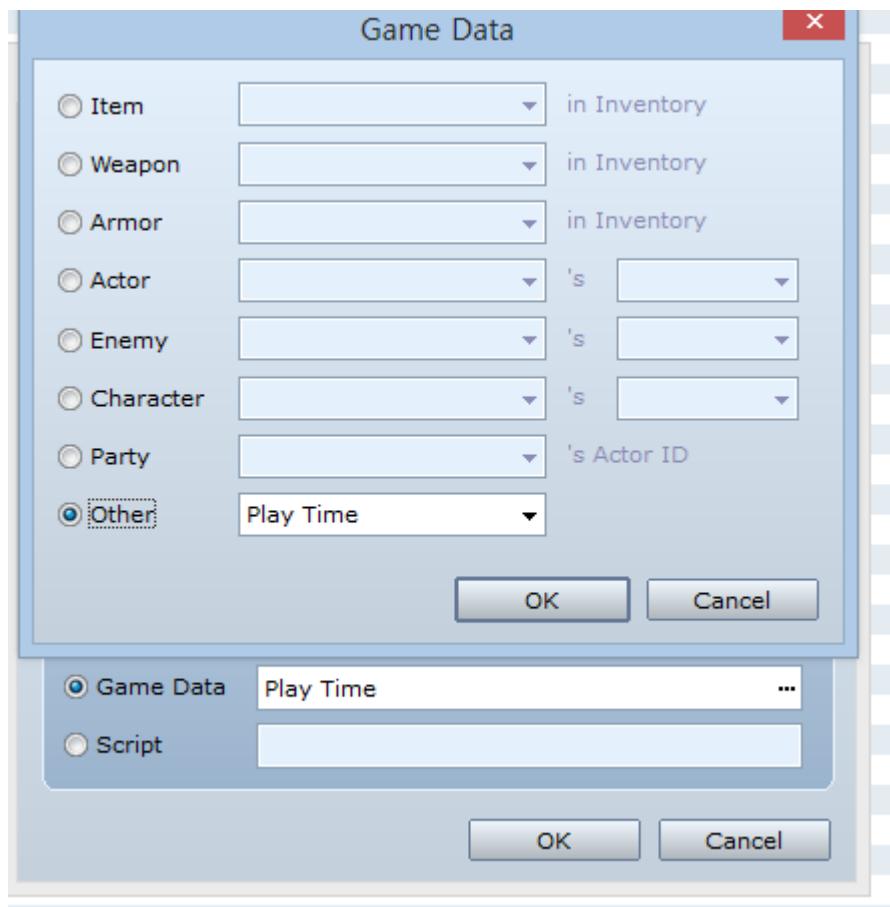
이렇게 스크립트를 설명하는 이유는 RPG Maker VX Ace에서 이벤트를 통해서 스크립트 언어를 대신 조작해주고 있다는 점을 인지시켜드리기 위해서입니다.

Mod (나머지 연산)에 대한 이론적인 설명

Mod 는 나눗셈을 할 때 몫을 구한 후에 남는 나머지값입니다. 나머지 값은 항상 나누는 수보다 값이 작습니다. 어떤 수를 나눠도 나머지값은 나누는 수보다 값이 적게 나오게 됩니다. 이 특징을 이용하면 게임에서 특정값을 무조건 최대값 미만으로 고정시킬 수 있습니다.

Play Time 예제를 통한 Mod의 이해

Mod 의 여러가지 사용 예제를 보겠습니다. 먼저 현재 플레이 타임을 변수에 대입해보도록 하겠습니다. 이렇게 하면 현재 플레이 시간이 변수에 대입될 것입니다.



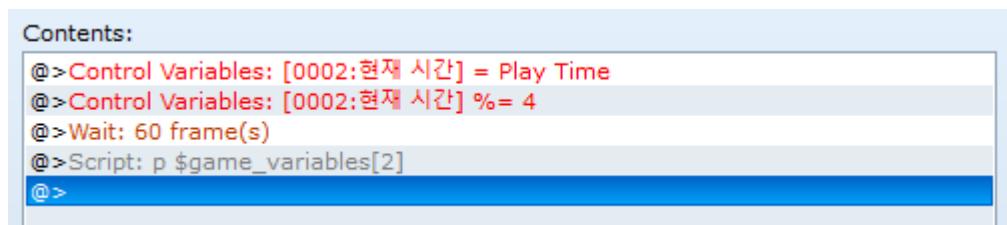
▲ Play Time 을 대입하고 있는 화면

게임 데이터란에서 Other - Play Time 을 통해 현재 플레이 시간을 변수에 대입할 수 있습니다.



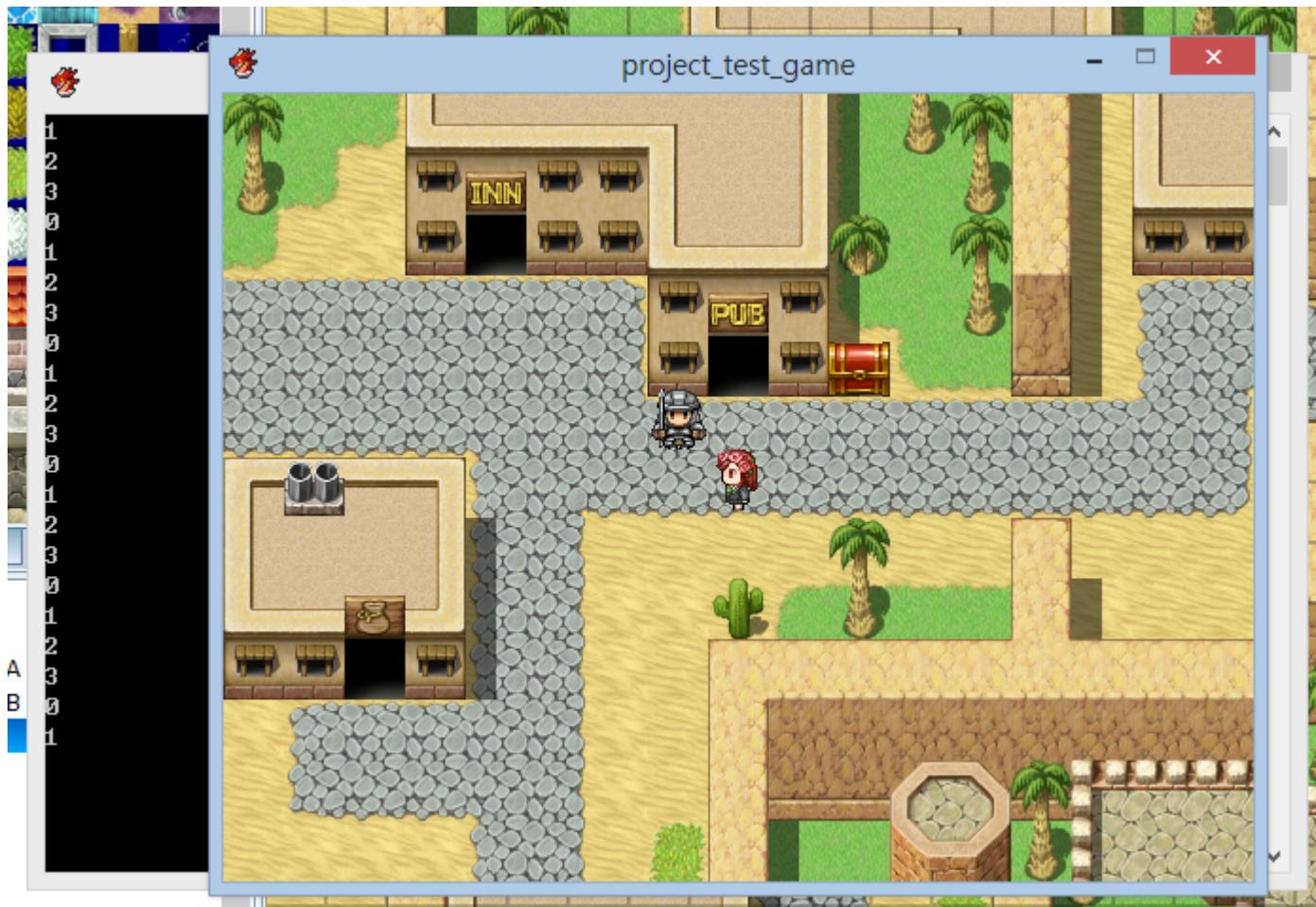
▲ 이벤트 커맨트 리스트에 변수의 조작 부분이 추가된 모습. (스크립트 언어와 동일한 모습)

병렬 이벤트를 통해 현재 시간을 4로 나누고 그 나머지를 나머지값을 얻어보도록 하겠습니다.



▲ 나머지 값의 계산 과정 (실제로 스크립트에서도 %= 로 나머지를 구합니다)

p는 콘솔에 메시지를 출력하는 스크립트 명령어이며 변수값을 확인할 수 있는 RPG Maker VX Ace에서 제공하는 또 하나의 디버그 기능이기도 합니다. 우리는 이전에 배웠던 스크립트로 변수에 접근하는 방법을 통해서 콘솔에 변수의 값을 출력해보도록 하겠습니다. 기억하셔야 할 점은 나머지 값은 항상 나누는 수보다 작다는 것입니다. 실제로도 그렇게 나오는지 한 번 확인해보겠습니다. 플레이타임은 매 초마다 그 값이 현재 플레이 시간에 맞게 증가하는 변수입니다. 그렇다면 결과는 어떻게 될까요?

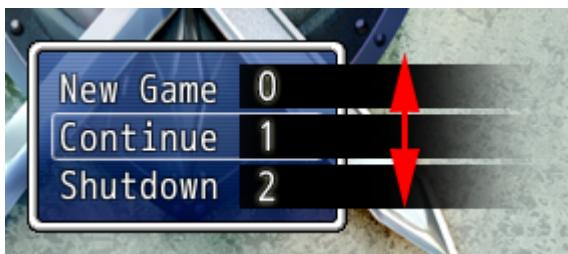


▲ 왼쪽에 출력 결과가 나옵니다

왼쪽에 출력 결과가 나오고 있습니다. 나누는 수였던 4 보다는 미만의 값이 출력되고 있습니다. 또한 플레이 시간이 증가되고 있기 때문에 나머지값도 함께 증가되고 있다는 것을 알 수 있습니다. 또한 4를 절대 넘진 않으며 0부터 다시 시작한다는 것을 알 수 있습니다. 이러한 특징은 다양한 곳에 응용할 수 있습니다. 보통 나머지값은 나누는 수 보다 작게 만들어 주기 위해 사용합니다만, 실제로 사용된 예제를 한 번 보겠습니다.

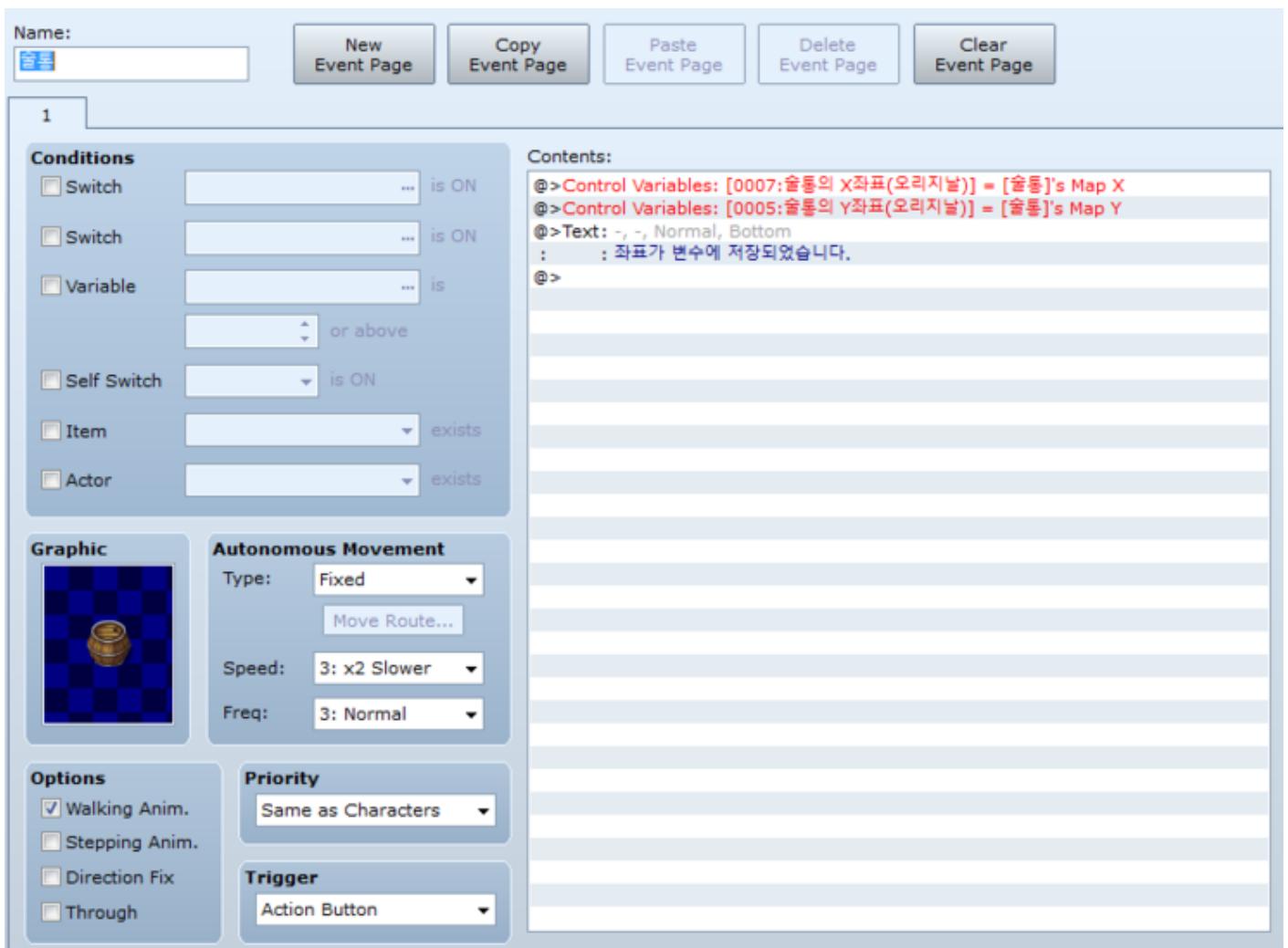
Mod를 이용하여 술통의 위치 바꾸기

Mod 가 사용된 적절한 예는 위와 같은 타이틀 메뉴 버튼들입니다. 타이틀 메뉴 버튼에는 번호가 붙어있습니다. 새로운 시작은 0번, 계속 하기는 1번이고, 종료 하기는 2번이 붙어있습니다.



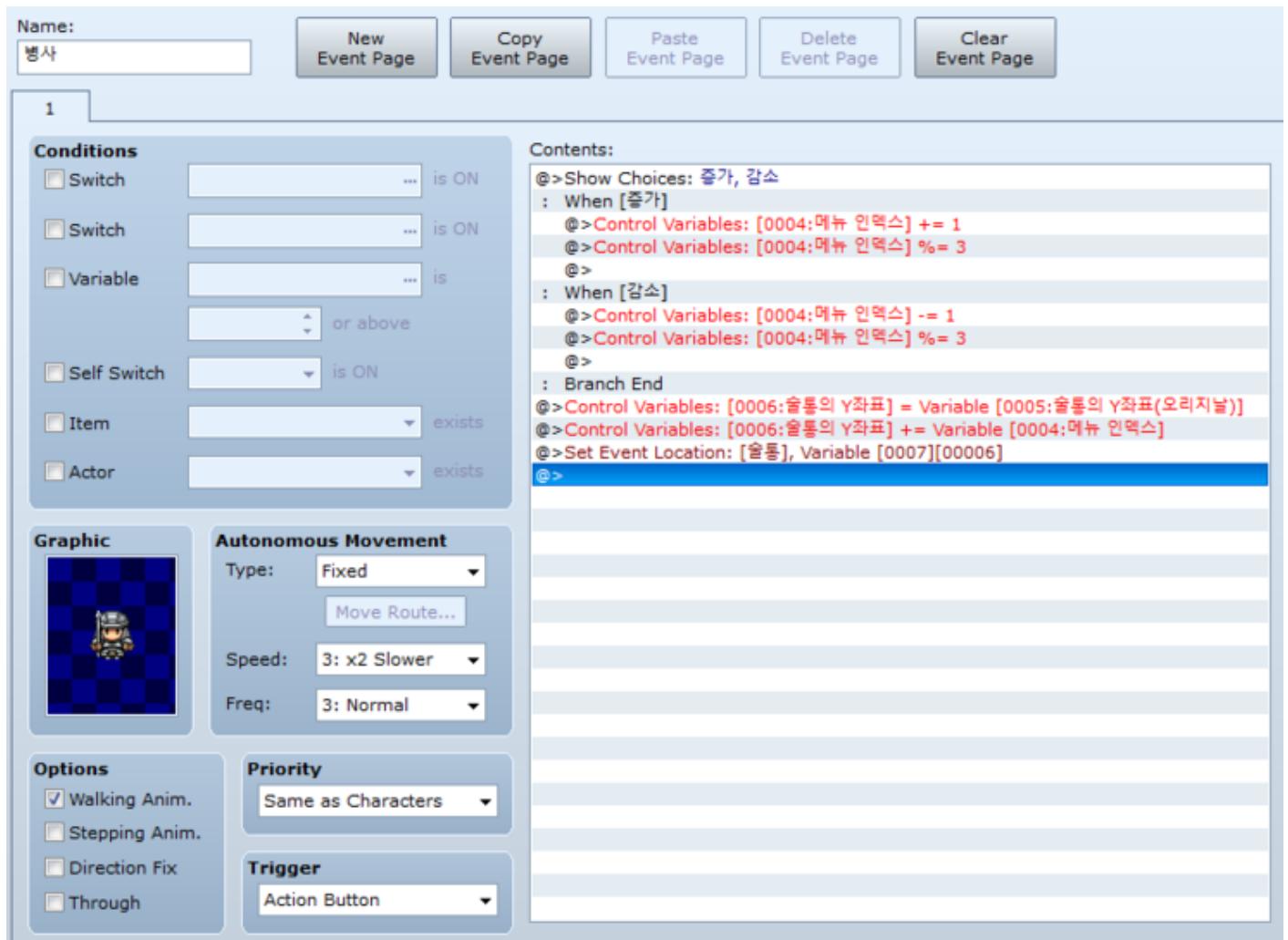
▲ 타이틀 메뉴의 모습

위쪽 방향키를 누르면 숫자가 감소합니다. 아래쪽 방향키를 누르면 숫자가 증가합니다. 하지만 범위는 0이상 2이하로만 나와야 한다는 것을 알 수 있습니다. 그런데 여기서 Mod 를 통해 최대값을 제한하면서 나머지값을 계산하면 어떻게 증가되고 감소되는 간에 무조건 0,1,2 라는 값만 나오게 됩니다. 이벤트 편집 에디터를 통해 이와 비슷하게 위치를 바꾸는 이벤트를 만들어서 확인해보도록 하겠습니다.



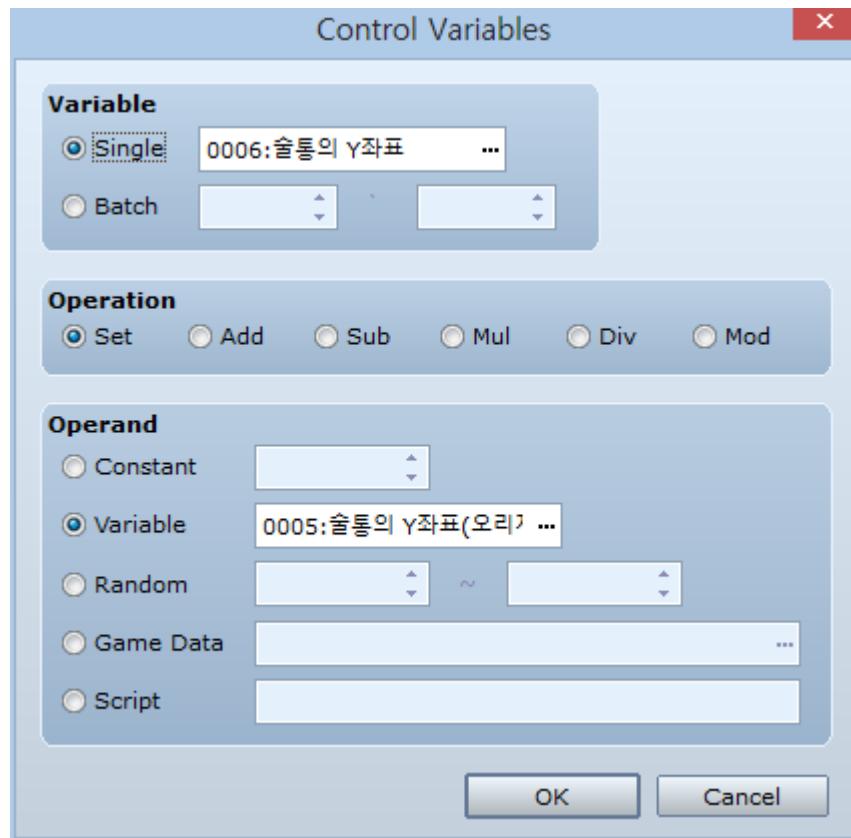
▲ 7번 변수와 5번 변수에 술통의 X좌표, Y좌표를 저장하고 있는 모습

5, 7번 변수는 술통이 맵의 어느 지점에 있는지를 나타내는 각각의 숫자들이 저장되어있습니다. 이 숫자들은 이벤트의 좌표라고 부릅니다. 술통 이벤트를 실행하면 술통의 초기 위치값이 변수에 저장됩니다. 이렇게 저장된 변수값은 아래 병사 이벤트에서 유용하게 사용됩니다.



▲ 술통의 위치를 옮기는 경비병 캐릭터. 선택지 기능으로 증가, 감소를 선택하게 하고 있습니다.

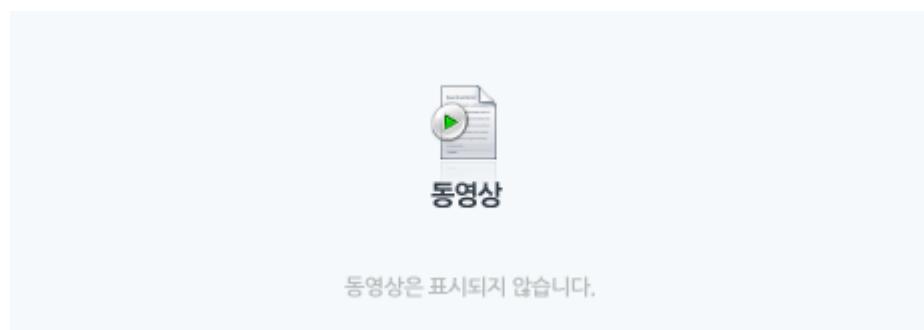
위는 선택지의 표시라는 기능으로 증가를 선택하면 값이 증가되고 감소를 선택하면 값이 감소되는 이벤트입니다. 모든 변수는 별도로 0을 대입하지 않아도 초기값이 0으로 자동으로 설정이 되어있습니다. 그렇기 때문에 메뉴 인덱스 변수는 처음부터 -1 을 감소시킬 수 있습니다. 메뉴의 갯수가 3개이기 때문에 3으로 나눠서 나머지값을 얻어줬습니다. 그러면 값은 0,1,2 가 나오게 됩니다. Mod 라는 것이 무엇인지 이해가 되었다면 0,1,2 라는 값이 어떻게 나오는지 이해가 되셨을 것입니다.



▲ 변수에 다른 변수의 값을 대입할 수도 있습니다.

이렇게 도출된 메뉴 인덱스 값을 이용하면 슬통의 Y좌표값을 계산해줄 수 있습니다. 초기 지점의 Y좌표에 메뉴 인덱스값을 더하면 될 것입니다. 그 후에 위치 지정 이벤트 명령을 통해 X값과 Y값을 지정해준다면 슬통도 메뉴처럼 컨트롤 할 수 있게 됩니다. 스크린샷으로는 과정이 잘 느껴지지 않기 때문에 이 결과는 동영상으로 확인해보겠습니다.

이벤트 실행 결과 확인



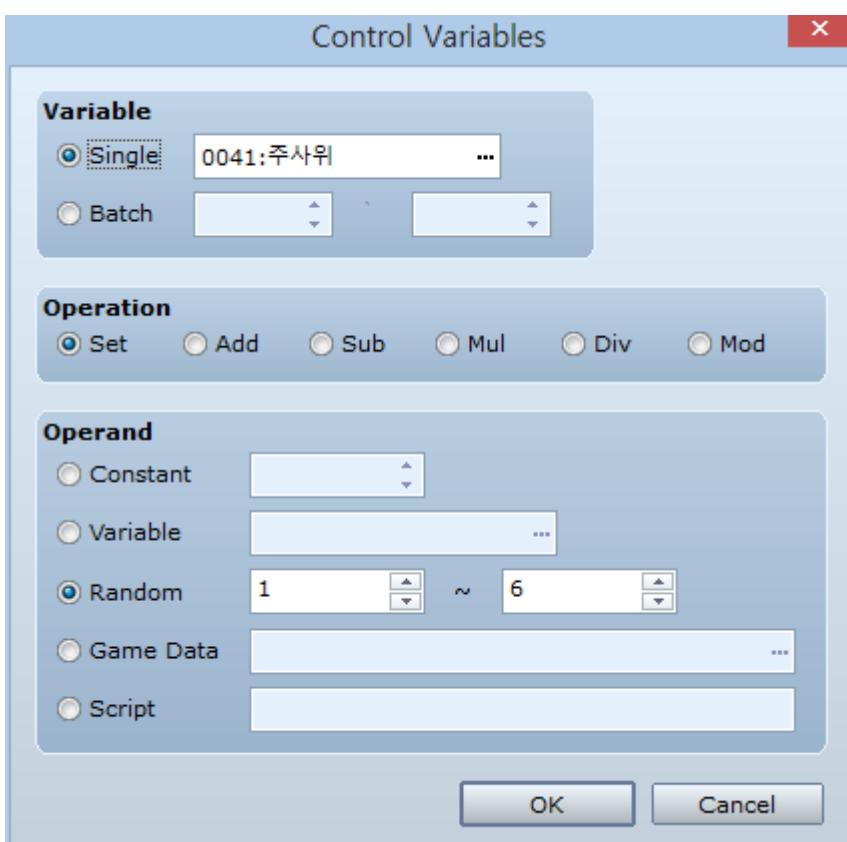
슬통이 메뉴처럼 컨트롤 되고 있다는 것을 볼 수 있습니다. 나머지값이 나누는 수보다 항상 작고 항상 일정한 범위 안에 있기 때문에 Y좌표값도 일정 범위 안으로 고정된 것입니다. 나머지 값은 이러한 특징 때문에 다양한 곳에서 활용될 수 있습니다.

조건 분기(Conditional Branch)

아직 변수값 대입에 Random이라는 것을 피연산자로 사용하지 않았습니다. 주사위를 돌리는 이벤트를 만들어서 램덤이라는 값을 확인해보도록 하겠습니다. 램덤이라는 것은 임의로 하나의 숫자를 정할 때 쓰는데요. 2개의 박스에 최소값과 최대값을 범위로 적으면 그 범위 내에 속하는 숫자 하나가 임의로 선택되면서 변수에 대입됩니다. 이러한 램덤값은 조건 분기(Conditional Branch)라는 기능과 함께 쓰이는게 일반적입니다. 변수는 조건 분기와 함께 쓰이는 경우가 많습니다. 그렇다면 조건 분기란 무엇일까요?

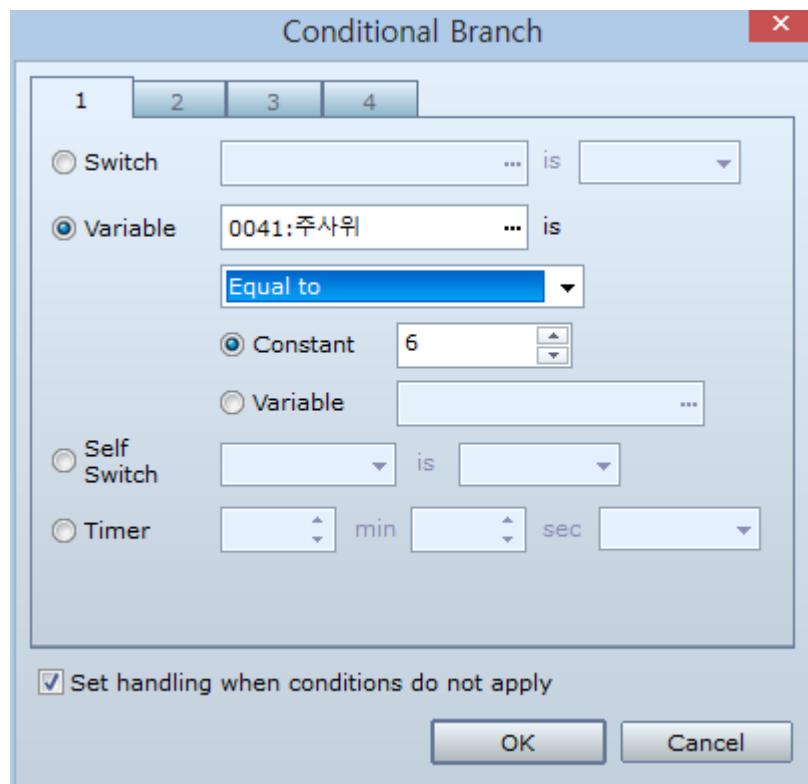
조건 분기는 참/거짓 유무에 따라 게임 흐름을 분기시켜주는 역할을 가지고 있습니다. 특정 조건이 참이면 어떤 일이 발생하기도하고 거짓이라면 발생을 하지 않을 수도 있습니다. 이렇게 참/거짓에 따라 2가지로 분기되면서 이벤트의 양상이 달라지는 것이 조건 분기라고 할 수 있습니다.

예를 들어보겠습니다. 간단하게 주사위를 만들어서 6이 나오면 선물을 주고 그게 아니라면 선물을 주지 않는 이벤트를 만들어보겠습니다.



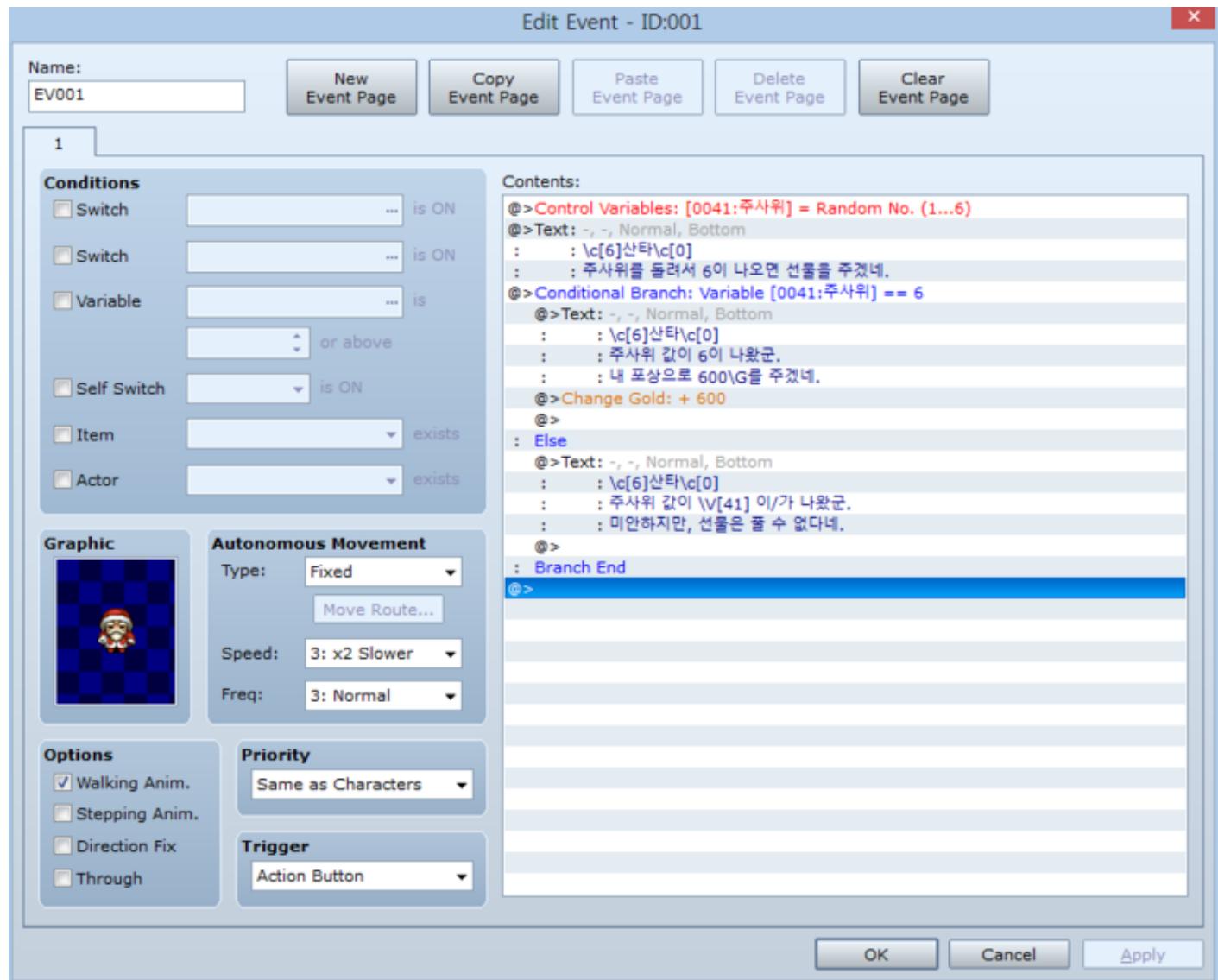
▲ 1부터 6까지의 숫자 중 하나가 임의로 대입됩니다

우리는 피연산자 중에 램덤이라는 부분을 배우지 않았습니다. 범위를 지정하면 임의로 하나의 숫자를 골라내는 것이 램덤입니다. 하나 숫자가 변수에 대입이 되면 그 변수의 값을 가지고 조건 분기로 조건을 걸고 그 조건이 참이라면 참에 해당하는 이벤트가 나오고 거짓이라면 거짓에 해당하는 대화가 나오게 됩니다.



▲ 변수의 값이 6과 같을 때

조건 분기는 첫 페이지에 Conditional Branch 라는 버튼을 누르면 만들 수 있습니다. 다양한 조건들을 체크박스를 체크하여 선택이 가능한데 우리는 변수(Variable) 부분을 체크하여 변수만 체크하도록 하겠습니다. 아래 Set handling when conditions do not apply 에 체크하면 거짓인 경우도 체크할 수 있게 됩니다.



▲ 조건 분기를 이용한 주사위 이벤트

41번 변수의 값이 6과 같은지 아니면 같지 않은지에 대해서 검사하고 있습니다. 6과 같다면 위쪽에 있는 대화가 뜨게 되고, 같지 않으면 아래쪽에 있는 대화가 뜨게 됩니다. 이렇게 이벤트의 양상을 두 갈래로 분기할 수 있습니다. 이 이벤트에서는 같은값만 조건으로 체크하고 있지만 조건 분기 변수 부분에서는 이외에도 변수의 값이 특정값 이상일 때, 이하일 때, 미만일 때, 초과, 같지 않을 때의 경우도 체크할 수 있습니다.



▲ 주사위를 돌려서 선물을 주는 산타

랜덤값이 1로 나왔습니다. 1이 나오게 되면 6과 같지 않기 때문에 조건이 맞지 않아서 집안에 머물러 있는 산타는 선물을 주지 않게 됩니다.



▲ 오랜 시도 끝에 겨우 600원을 받고 있는 모습

주사위 값이 6이 나오게 되면 조건에 맞는 값이기 때문에 산타에게서 선물을 받을 수 있게 됩니다. 이렇게 조건 분기는 흐름을 분기합니다.

조건 분기를 이벤트의 흐름을 두 갈래로 나눌 수 있었습니다. 조건 분기에서는 변수 뿐만 아니라 스위치도 체크할 수 있고 주인공이 이 아이템을 가지고 있는지, 키가 눌려있는지에 대해서도 참/거짓을 체크할 수 있습니다. 이외에도 조건들이 많기 때문에 어떤 조건들이 있는지 조건 분기를 열어서 한 번 살펴보시기 바랍니다.

조건 분기의 사용법을 알게 되었으니 변수의 값 뿐만 아니라 아직 사용하지 못한 조건 분기들도 언젠간 유용하게 사용할 수 있을 거라 생각합니다.

Q. 셀프 변수도 있나요?

셀프 변수는 따로 구현되어있지 않기 때문에 구현된 스크립트를 추가해서 사용해야 합니다.

아래 링크를 참고해보세요.

[셀프 변수](#)

[06강] 비밀 번호의 처리 - RPG 만들기 VX Ace 강좌

시작하기

2014/11/05 16:40

<http://blog.naver.com/biud436/220172415977>

RPG Maker VX Ace 왕초보탈출 6

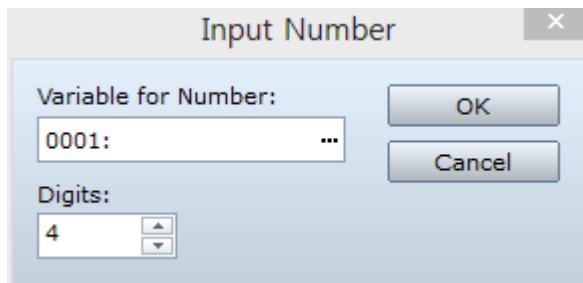
"비밀 번호의 처리"

안녕하세요. 러닝은빛입니다. 이번 강좌에서는 저번 시간에 이어서 조건 분기에 대한 내용을 다뤄보도록 하겠습니다. 여러분들이 쭈쭈르 게임을 즐겨해보셨다면 비밀번호를 입력받게 하는 이벤트를 많이 보셨을 겁니다. 이벤트 커맨드에는 숫자 값을 직접 입력받아서 변수에 저장시키는 이벤트 커맨드가 있습니다. 변수에 숫자값을 입력받은 후에 조건분기로 이 변수의 값이 특정값과 맞는지 비교해준 후 참/거짓 처리를 해주면 되는 매우 간단한 이벤트 중 하나입니다.

왕초보탈출이라는 강좌에 매우 적합한 강좌라고 볼 수 있습니다. 비밀번호의 처리는 아래 두 가지 단계로 나눌 수 있습니다. 첫번째 단계는 Input Number라는 이벤트 커맨드를 지원합니다. 이 기능을 활용하면 플레이어로부터 숫자를 입력받을 수 있습니다. 숫자 형식으로 1부터 8자까지 입력받을 수 있습니다.

1. 플레이어로부터 숫자를 입력받고 입력받은 값을 특정 변수에 저장하는 기능

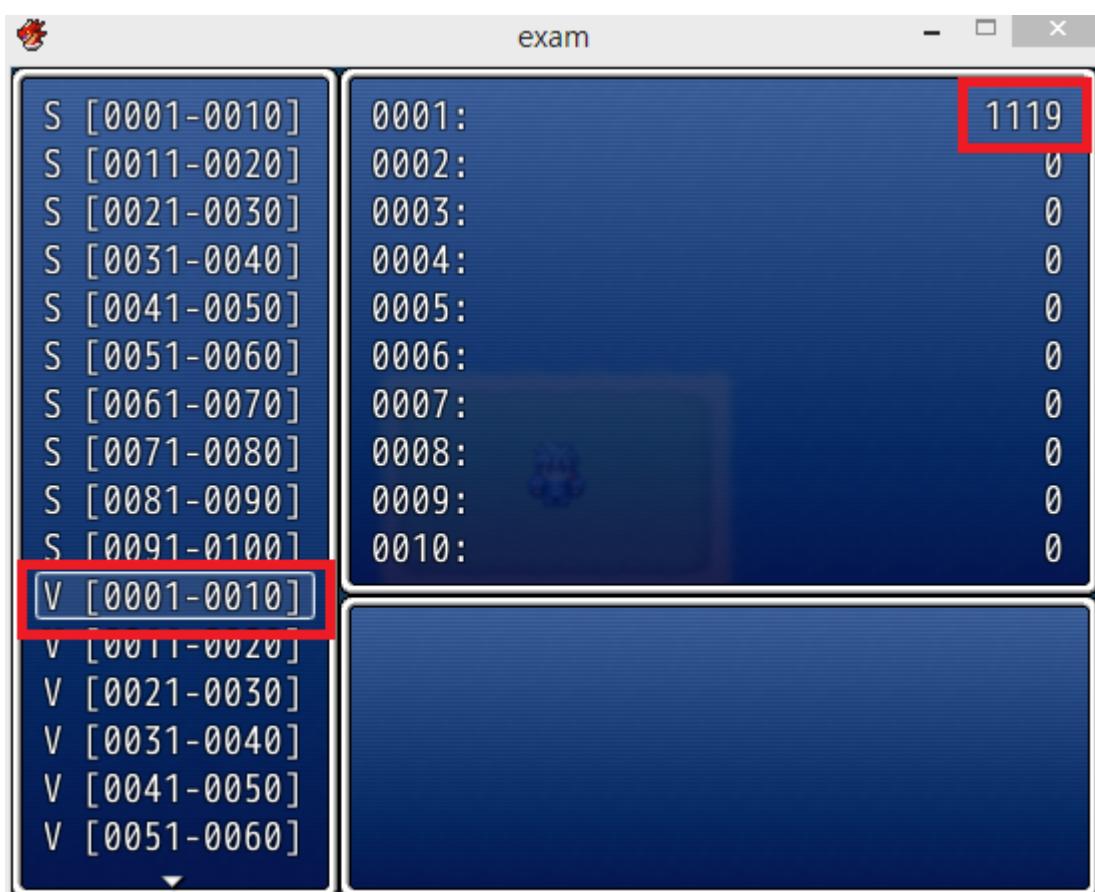
Input Number라는 이벤트 커맨드를 누르면 변수 선택 박스와 숫자의 자릿수를 고를 수 있는 박스가 보입니다. 변수와 자릿수를 고르는 것인데 이 기능을 통해 숫자를 입력받을 수 있습니다. 입력받은 숫자는 변수 선택 박스를 통해 골랐던 변수에 저장됩니다. 보통 게임에서는 4자리를 많이 사용합니다.



▲ 숫자의 자릿수와 값을 저장할 변수를 고릅니다



▲ 인게임에서 위와 같은 형태로 숫자를 입력받게 됩니다.



▲ 디버그 기능으로 변수값을 확인해본 결과 1119라는 값이 저장되어있는 것을 알 수 있습니다.

2. 입력받은 숫자값을 조건분기를 통해 체크해서 참과 거짓으로 흐름을 분기해줘야 합니다.

수자를 자유롭게 입력받긴 했지만 변수에 어떤 값이 저장되어있는지는 알 수 없기 때문에 조건 분기를 통해서 체크를 해줘야 합니다. 설정하고 싶은 비밀번호도 조건분기에서 설정할 수 있습니다. 설정하고자 하는 비밀번호의 값이 1119 이기 때문에 플레이어가 입력한 값이 1119 인지를 체크해주면 됩니다. 결과는 아래와 같습니다.

Contents:

```
@>Text: -, -, Normal, Bottom
: 할머니
: 이 섬에서 빠져나가고 싶으면 암호를 대게
@>Input Number: [0001], 4 digit(s)
@>Conditional Branch: Variable [0001] == 1119
@>Text: -, -, Normal, Bottom
: 할머니
: 맞췄구먼, 이 섬을 빠져나가게 해주겠네.
@>
: Else
@>Text: -, -, Normal, Bottom
: 할머니
: 틀렸네. 빠져나갈 생각이 전혀 없나보구먼
@>
: Branch End
@>
```

▲ 변수값을 체크하여 참(True)과 거짓(False)으로 흐름을 분기하고 있는 모습

매우 간단한 과정으로 비밀번호를 체크하고 있다는 것을 알 수 있습니다.

3. 비밀번호가 걸려있는 보물상자

단축메뉴(Ctrl + 3)로 만들 수 있는 기본 보물상자에 자물쇠를 걸어보았습니다. 비밀번호를 맞춰야만 풀리는 자물쇠이기 때문에 비밀번호를 맞추게 될 때에만 상자가 열리게 됩니다.

Edit Event - ID:002

Name: EV002

New Event Page Copy Event Page Paste Event Page Delete Event Page Clear Event Page

1 2

Conditions:

- Switch is ON
- Switch is ON
- Variable is ... or above
- Self Switch is ON
- Item exists
- Actor exists

Graphic:

Autonomous Movement:

- Type: Fixed
- Move Route...
- Speed: 3: x2 Slower
- Freq: 3: Normal

Options:

- Walking Anim.
- Stepping Anim.
- Direction Fix
- Through

Priority: Same as Characters

Trigger: Action Button

Contents:

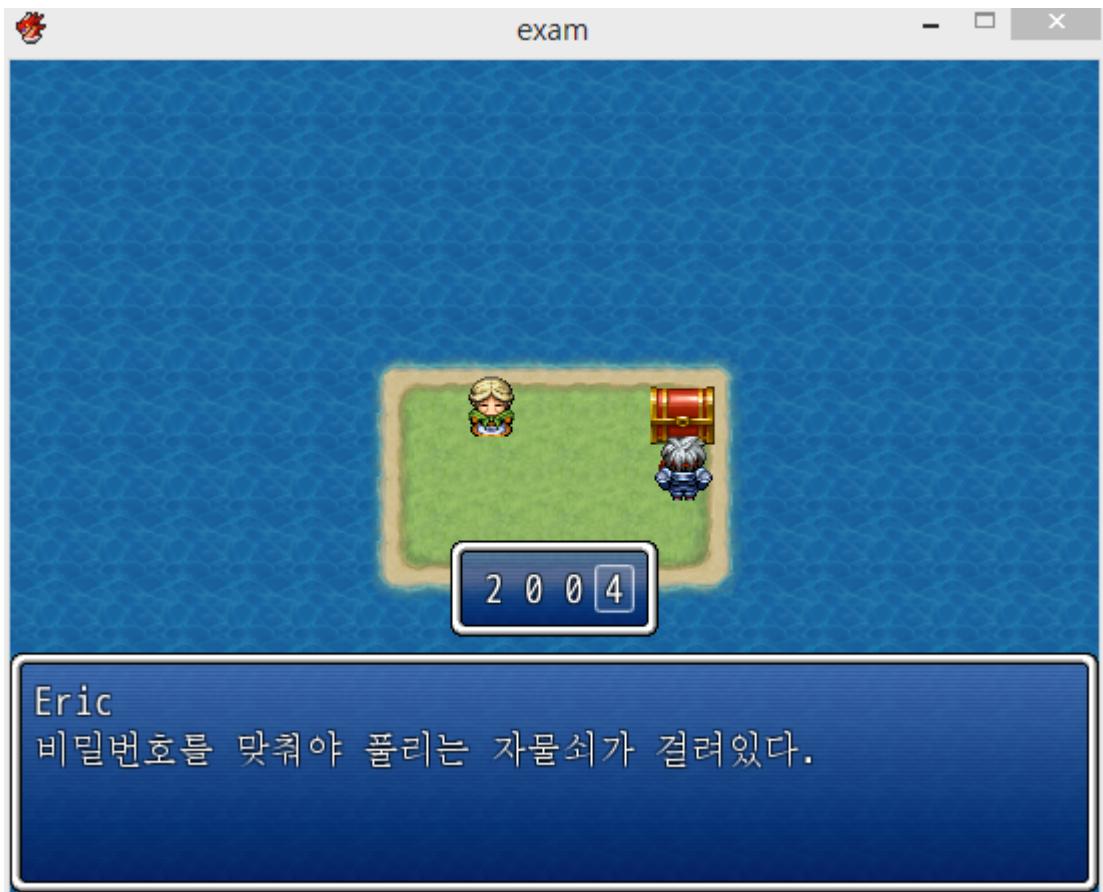
```

@>Text: -, -, Normal, Bottom
: : \n[1]
: : 비밀번호를 맞춰야 물리는 자물쇠가 걸려있다.
@>Input Number: [0002], 4 digit(s)
@>Conditional Branch: Variable [0002] == 2004
@>Play SE: 'Chest', 80, 100
@>Set Move Route: This event (Wait)
: : $>Direction Fix OFF
: : $>Turn Left
: : $>Wait: 3 frame(s)
: : $>Turn Right
: : $>Wait: 3 frame(s)
@>Control Self Switch: A =ON
@>Change Gold: + 300
@>Text: -, -, Normal, Bottom
: : 300\G were found!
@>
: Else
@>Text: -, -, Normal, Bottom
: : \n[1]
: : 비밀번호가 맞지 않아 자물쇠가 풀리지 않는다.
@>
: Branch End
@>

```

OK Cancel Apply

▲ 보물상자 이벤트의 응용



▲ 비밀번호를 풀고 있는 모습

[07강] 게이지바 만들기 - RPG 만들기 VX Ace 강좌

시작하기

2014/11/05 16:42

<http://blog.naver.com/biud436/220172418665>

RPG Maker VX Ace 왕초보탈출 7

"게이지바 만들기"

안녕하세요. 러닝은빛입니다.

이번 왕초보탈출 강좌에서는 체력, 마력, 경험치 게이지바를 각각 만들어보도록 하겠습니다. 변수에 플레이어의 능력치를 설정할 수 있게 되고 그림을 표시하거나 변수에 설정된 좌표에 따라 그림을 이동시키는 법에 대해서 알 수 있게 됩니다.

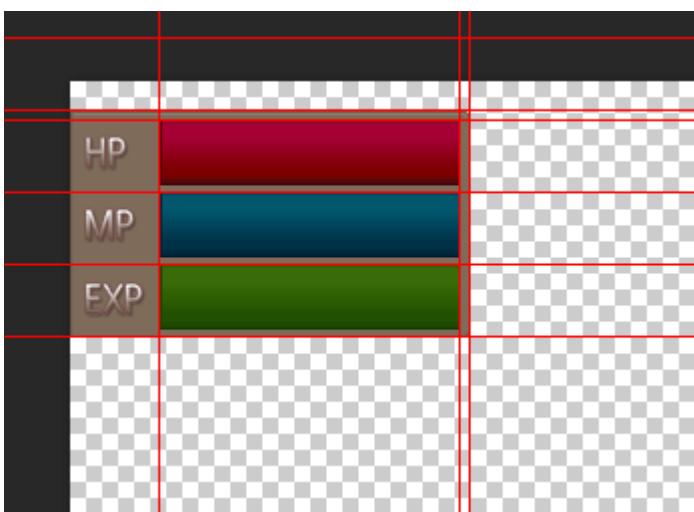
스프라이트를 만들고 각 스프라이트의 좌표 추출하기

게이지바의 틀, HP, MP, EXP 를 기준으로 총 4개의 스프라이트를 만들어야 합니다. 그럴려면 먼저 아래와 같이 비트맵 제작 프로그램이 준비되어있어야 하는데요. 사실 스프라이트를 만들 수 있는 어느 프로그램이든 괜찮습니다만 저는 포토샵으로 작업을 했습니다. 사실 스프라이트 제작 작업은 시간이 많이 걸리는 작업입니다. 또한 포토샵 강좌나 이런 스프라이트를 만들 수 있는 방법은 인터넷에 검색하면 많이 나오기 때문에 이 게시물에서 굳이 설명을 하진 않겠습니다. (포함하면 포함할 순 있지만 별도의 노력을 필요로 하는 분야라 강의 쪽에 포함하는데는 한계가 있습니다) 결과적으로 우선 아래와 같은 형식의 이미지를 만들어야 합니다.



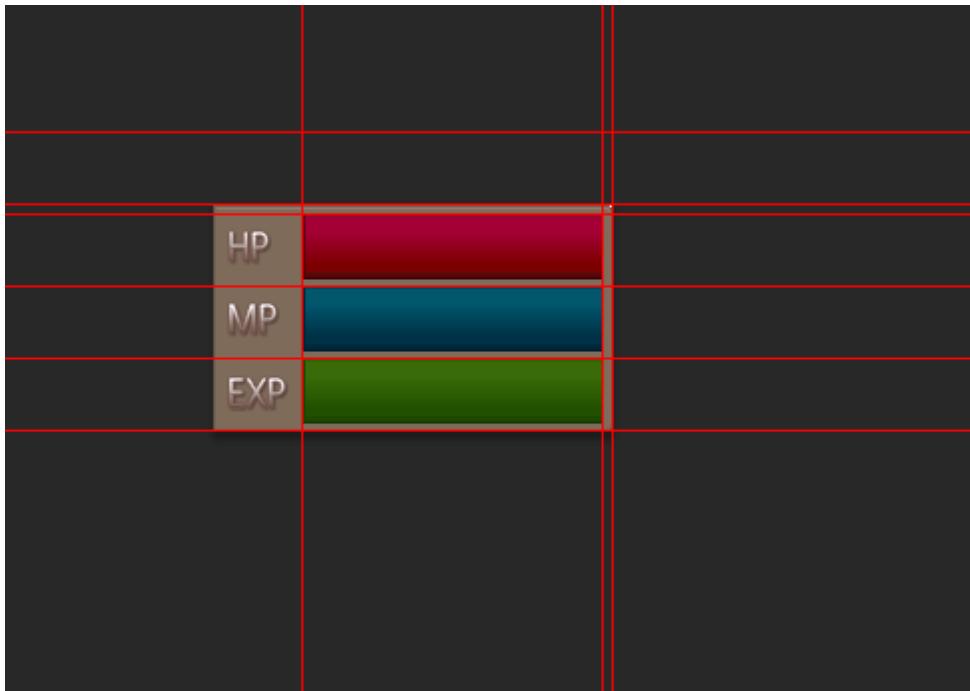
▲ 포토샵 화면

틀 이미지는 왼쪽에 텍스트가 있는 공백 부분이 있고 게이지바가 위치하는 공간은 비어있어야 합니다.



▲ 포토샵 화면

게이지바를 레이어 별로 만들었다면 게이지바의 주요 좌표를 먼저 추출할 수 있어야 합니다.



▲ 포토샵 화면

좌표는 화면 기준이 아니라 게이지바 틀을 기준으로 잡고 내부 좌표를 추출해보겠습니다. 먼저 체력과 마력 경험치 게이지바의 가로 길이를 추출합니다. 다음에는 게이지바의 틀에서 HP,MP,EXP 스프라이트가 시작되어야 하는 지점의 좌표를 추출해야 합니다. 시작되는 지점은 각 스프라이트(HP,MP,EXP)가 위치해야 하는 지점의 가장 왼쪽 위의 점입니다. 그 지점의 좌표를 추출하면 아래와 같이 정리할 수 있게 됩니다.

구분	값
게이지바의 가로 길이	150
HP 게이지바	45, 5
MP 게이지바	45, 41
EXP 게이지바	45, 77

좌표를 추출했다면 이제 모든 준비가 끝이 났습니다. 레이어 별로 이미지를 복사하여 추출하면 총 4개의 스프라이트가 나오게 됩니다. 스프라이트를 준비하지 못하신 분들은 이 스프라이트들을 내려받아주시길 바랍니다.



▲ gauge.png



▲ exp.png

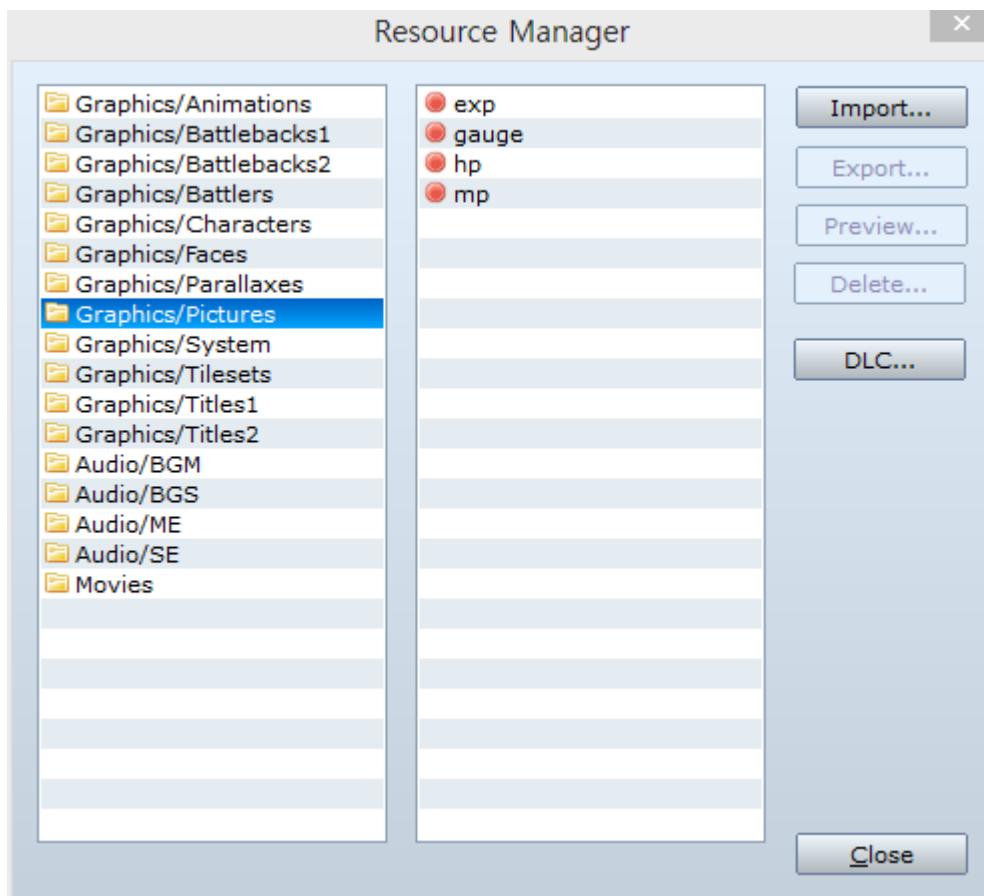


▲ mp.png



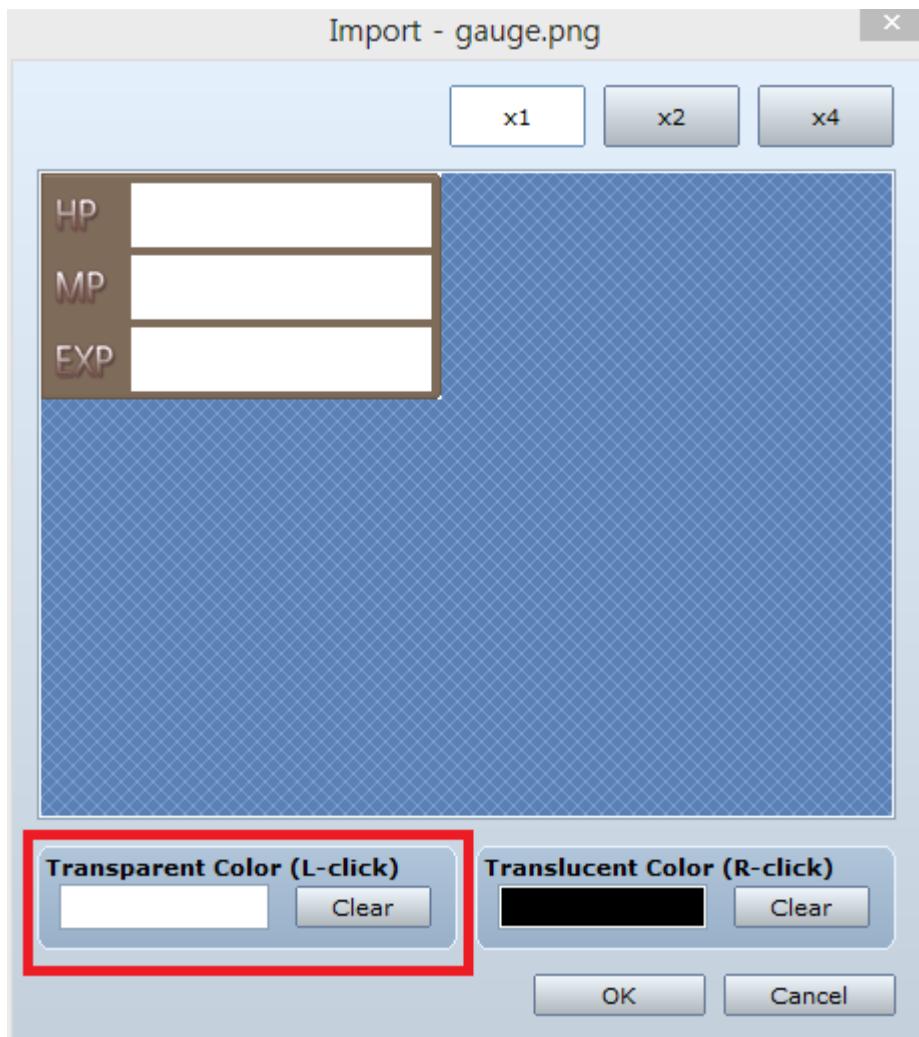
▲ hp.png

컴퓨터에 스프라이트들을 내려받았다면 RPG Maker VX Ace에서 F10 버튼을 눌러서 아래와 같은 리소스 매니저 창을 띄워주시기 바랍니다. 번역하면 리소스 관리자라는 기능으로 준비된 기본 RTP 이외의 리소스들을 불러올 수 있는 기능입니다.



▲ 리소스 매니저

리소스 매니저(F10)에서 Graphics/Pictures 폴더로 방금 만든 스프라이트 4개를 불러오시기 바랍니다. 투명색 설정이 필요한 부분이 있다면 불러오는 과정에서 투명색을 설정할 수 있게 됩니다. 하지만 포토샵 등에서 투명 레이어를 별도로 만들었다면 자동으로 투명한 부분을 인식합니다. 이 경우에는 투명색 설정창이 뜨지 않습니다.



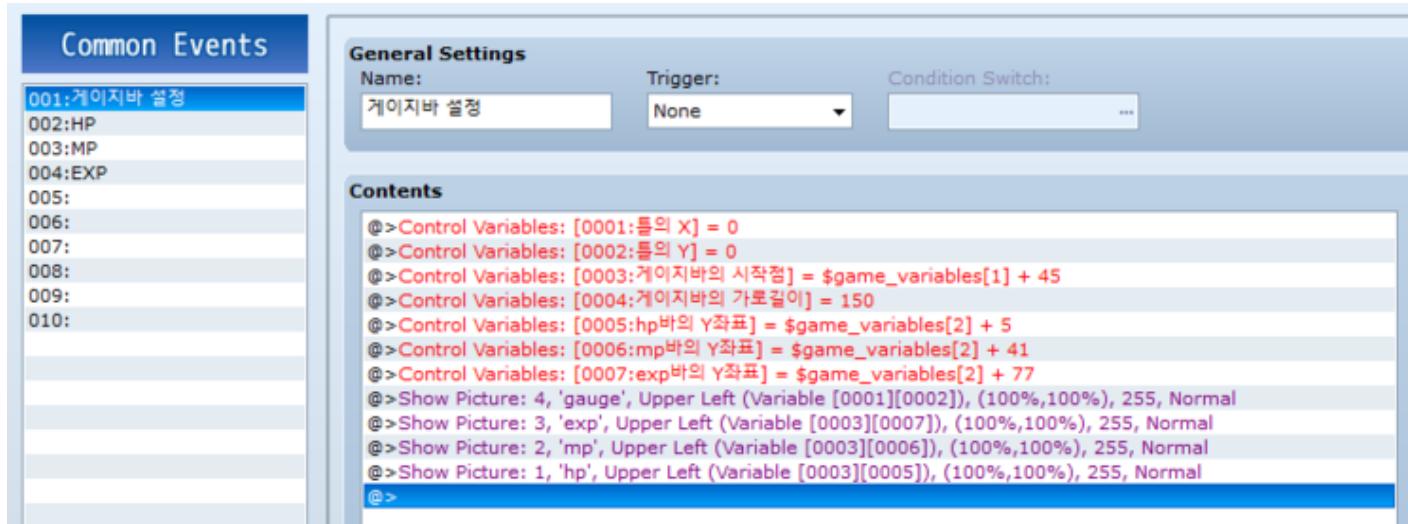
▲ 투명색을 설정하고 있는 모습

하지만 그 경우가 아니라면 위와 같이 투명색을 설정할 수 있는 별도의 창이 뜨게 됩니다. 투명색으로 설정되어야 할 지점을 클릭하면 그림이 표시될 때 해당 부분과 같은 색상은 모두 투명색으로 바뀐 상태로 출력됩니다.

RPG Maker VX Ace에서는 자주 쓰이는 이벤트를 미리 만들어서 맵 어디에서든지 다시 불러올 수 있는 기능이 있습니다. 이런 이벤트를 커먼이벤트라고 부르는데요. 이 커먼이벤트는 데이터베이스에서 설정할 수 있습니다.

게이지바의 기본 설정

기본 설정을 하기 위해 데이터베이스 - 커먼이벤트 1번에 "게이지바 설정" 이라는 이벤트를 만들어보겠습니다. 더불어 왼쪽 커먼 이벤트의 이름들도 미리 아래 스크린샷과 같이 각각 게이지바 설정, HP, MP, EXP로 설정해주시기 바랍니다.

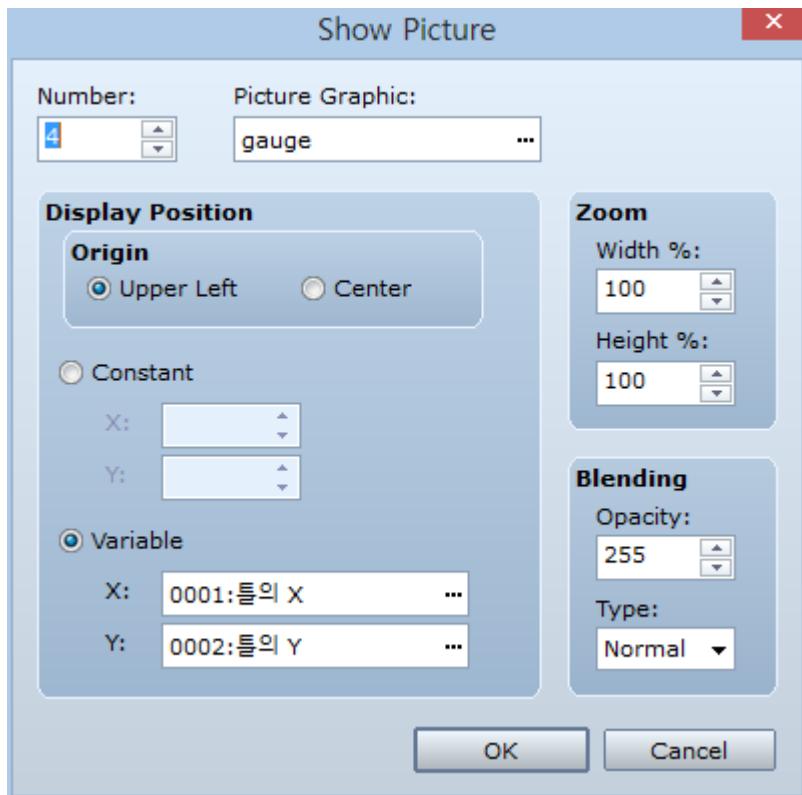


▲ 커먼 이벤트

먼저 저는 1번부터 7번 변수를 위와 같이 설정해줬습니다. `$game_variables[1]`은 이전강좌에서 스크립트에서 변수에 접근하는 또 방법이라고 설명드린 적이 있습니다. 이와 같이 스크립트 부분에서 변수에 접근하면 덧셈 연산의 결과를 변수에 저장 할 수 있게 됩니다. 이렇게 하면 이벤트의 길이를 더 줄일 수 있게 됩니다.

구분	X좌표	Y좌표
틀	0	0
HP바	45	틀의 Y좌표 + 5
MP바	45	틀의 Y좌표 + 41
EXP바	45	틀의 Y좌표 + 77
값		
게이지바의 시작점	틀의 X좌표 + 45	
게이지바의 가로길이	150	

이렇게 틀에 대해 상대적으로 좌표를 만들면 틀의 위치에 따라 상대적으로 각 스프라이트의 위치가 바뀌게 됩니다. 하단에 보이는 Show Picture는 2번째 페이지에 있습니다.



▲ 그림의 표시

이번 시간에는 Show Picture 커맨드와 Move Picture 커맨드를 이용하여 게이지를 출력하게 되는데 이 커맨드들은 모두 2페이지에 있습니다.

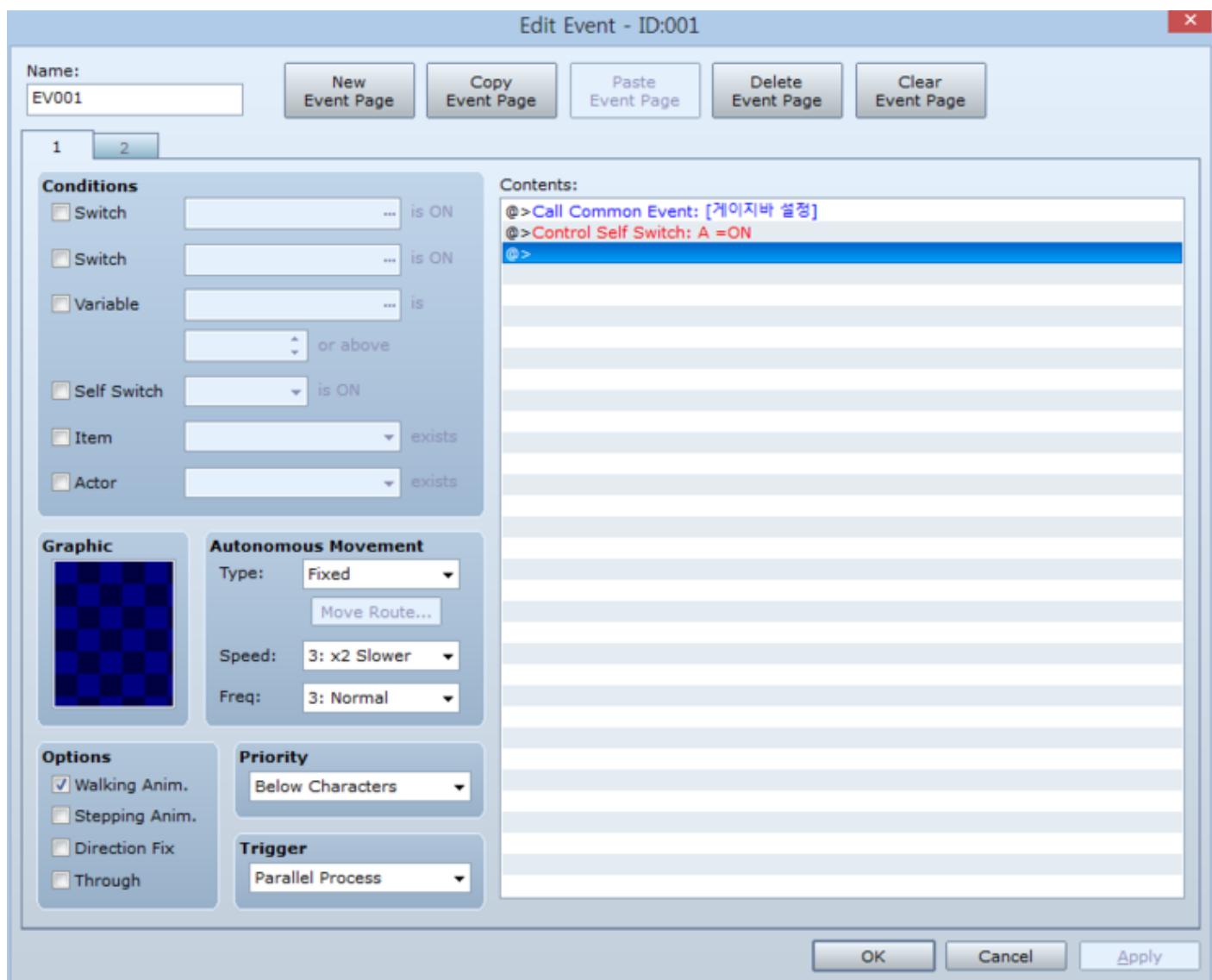
Picture Graphic 박스는 표시 할 그림을 선택하는 박스로 리소스 매니저를 통해 Graphics/Pictures 폴더로 불러왔던 그림들 중 하나를 설정할 수 있습니다.

Number는 그림의 번호를 지정할 수 있는 에디트 박스입니다. 숫자를 지정해야 하는데 이 숫자는 그림들의 Z좌표를 나타냅니다. Z좌표라는 것은 2D 게임이지만 3D 게임처럼 바닥이 있다고 가정하면 바닥에서부터 얼만큼 떨어져있는지를 말해주는 좌표입니다. 이 박스에 설정된 숫자는 그림의 기본 Z좌표값에 더해지게 되므로 같은 위치에 그림을 표시할 때 숫자값이 낮으면 숫자값이 높게 설정된 그림에 가려지게 됩니다.

확대(Zoom)와 혼합(Blending) 옵션이 있습니다. Opacity는 투명도입니다. 0은 투명이고 255가 불투명 상태입니다. 이 부분은 아직까진 굳이 건드리지 않아도 되는 부분입니다.

먼저 원점을 왼쪽 위(Upper Left)로 설정하고 그림 번호와 변수(Variable) 부분을 스크린샷을 잘 참고하여 설정해보세요. 참고로 게이지바의 틀은 제일 높은 번호로 설정해줘야 합니다. 그 이유는 게이지바의 틀이 가장 앞쪽에 표시되어야 하기 때문입니다.

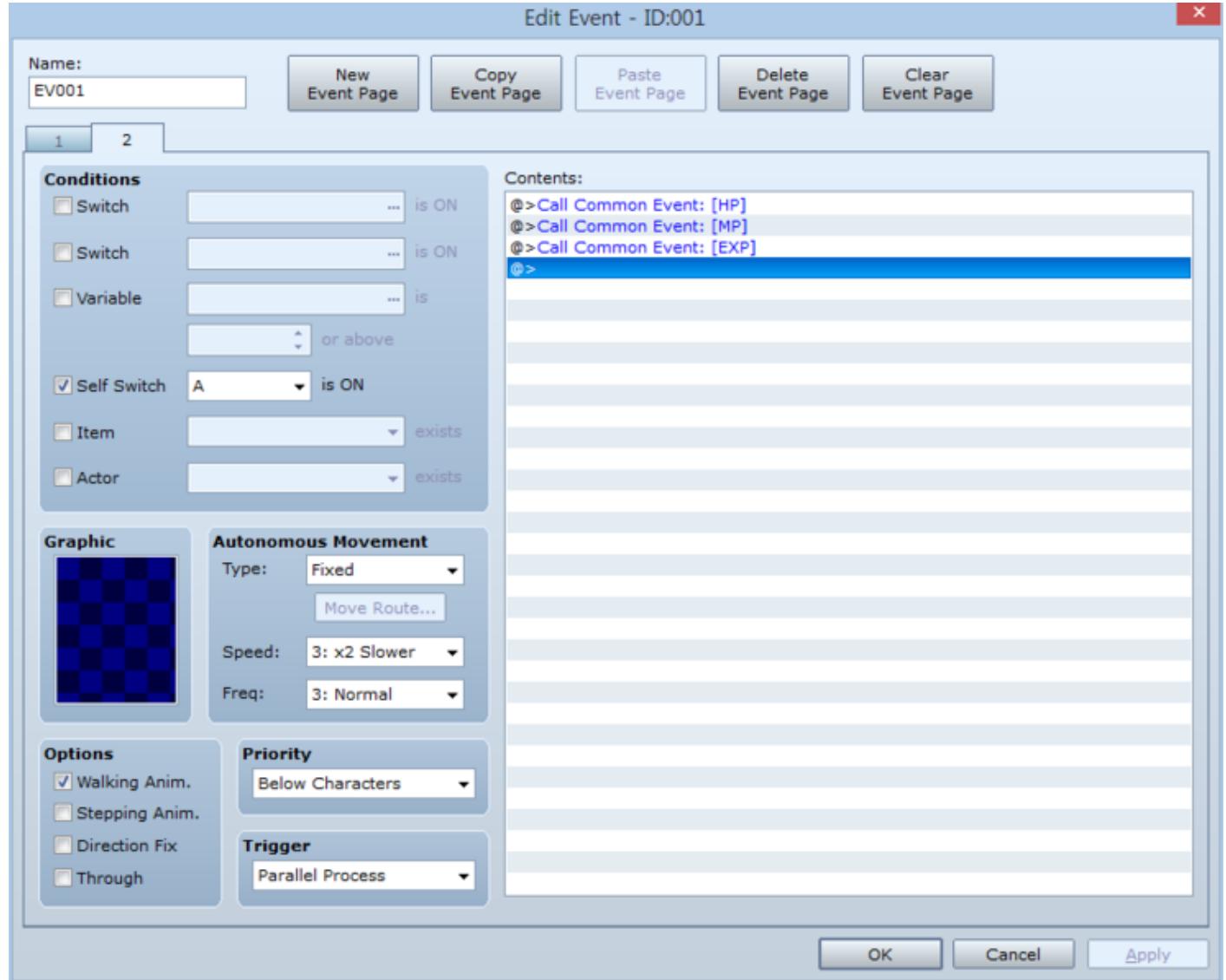
게이지바의 모든 설정을 마쳤다면 게이지바가 화면에 잘 표시되는지 확인해봐야하는데 가장 처음에 시작되는 맵에 병렬 이벤트를 만들어서 확인해볼 수 있습니다.



▲ 커먼 이벤트 호출

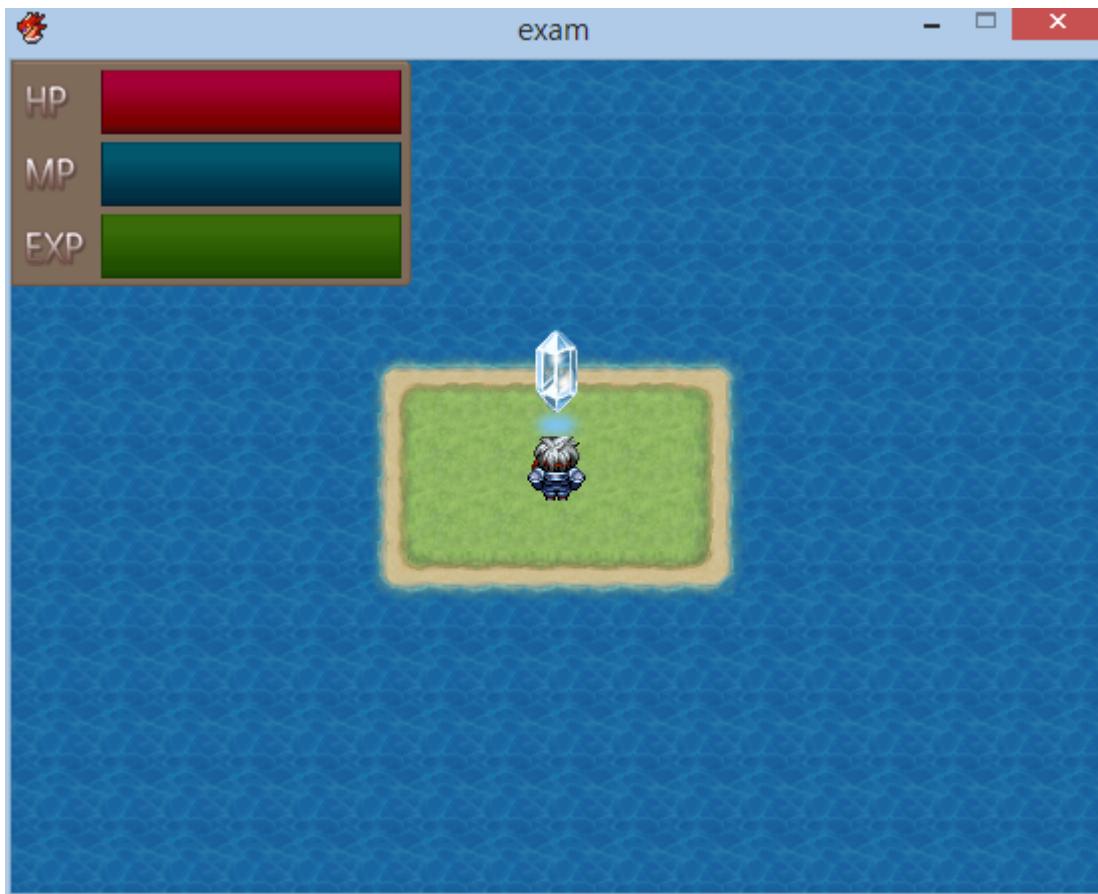
위와 같이 Call Command Event(커먼 이벤트)라는 부분을 이벤트 커맨드 목록에서 찾아서 방금 만든 커먼 이벤트를 호출해줍니다. 셀프 스위치를 하나 만들어서 다음 이벤트 페이지로 바로 연결해줍니다.

Edit Event - ID:001



▲ 커먼 이벤트 호출

아직은 실행 할 내용이 없지만 다음 페이지에 있는 커먼 이벤트들도 미리 만들어주도록 합시다. 이렇게 설정을 해줬다면 결과를 테스트 플레이를 통해 확인해보면 아래와 같이 표시됩니다.



▲ 테스트 플레이

위와 같이 표시된다면 성공한 것입니다. 그러나 이렇게 표시가 되지 않는다면 무언가가 잘못된 것이니 다시 확인해보시기 바랍니다.

체력 게이지바와 마력 게이지바의 이동

체력 게이지바와 마력 게이지바를 이동시키기 위해서는 HP의 비율을 계산해야 합니다. 기본 이벤트 커맨드 변수와 그림의 표시>Show Picture 그리고 그림의 이동>Move Picture 기능을 이용하여 게이지바를 이동시켜보겠습니다. 발전된 형태의 스크립트도 있지만 굳이 좌표를 이동시키는 방식으로 구현을 하는 이유는 2K시절을 계승하고자하는 것이 아닙니다. 스크립트로 만들기 위해서는 알아야 할 문법 자체가 워낙 많고 어렵기 때문입니다. 또한 애초에 프로그래밍을 배우지 않았다면 스크립트를 잘 다루기가 워낙 어렵고 배우는 길도 만만치 않습니다. 알만툴을 하는데 루비까지 제대로 배우고 루비의 매력에 빠져서 스크립터의 길을 걷는 분들은 알만툴 전체를 통들어서 그리 많지 않을 거라 봅니다. 그래서 이벤트로도 직접 만들어보면서 게이지바를 만드는 느낌이라는게 무엇인지 파악해보자 하는 것입니다.

The screenshot shows the 'General Settings' window for a script named 'HP'. The 'Name:' field contains 'HP', the 'Trigger:' dropdown is set to 'None', and the 'Condition Switch:' dropdown has '...' selected. Below the settings, the 'Contents' section displays the following script code:

```
@> Control Variables: [0012:HP비율] = [Eric]'s HP  
@> Control Variables: [0012:HP비율] *= Variable [0004:게이지바의 가로길이]  
@> Control Variables: [0012:HP비율] /= [Eric]'s MHP  
@> Control Variables: [0009:HP바의 X좌표] = Variable [0003:게이지바의 시작점]  
@> Control Variables: [0009:HP바의 X좌표] += Variable [0012:HP비율]  
@> Control Variables: [0009:HP바의 X좌표] -= Variable [0004:게이지바의 가로길이]  
@> Move Picture: 1, Upper Left (Variable [0009][00005]), (100%,100%), 255, Normal, @10  
@>
```

▲ HP

2K 시절을 기억하고 계시는 분들이 있을지는 모르겠지만 (있다면 최소 20대 중반) 게이지바를 누구의 도움도 없이 직접 만드는 것은 액알의 시초가 되는 일이기도 했습니다. 중고수로 가는 길목 중 하나가 이벤트로 게이지바 만들기였지 않을까 생각합니다. 하지만 스크립트라는게 생기면서부터 이러한 개념은 조금 달라졌습니다.

많은 제작자분들이 스크립트가 주는 편리함에 빠지게 된 것입니다.

초보자도 마찬가지이고 중고수분들도 마찬가지입니다. 누군가 구현해놓은 것을 스크립트라는 형태로 공유하기가 쉬워졌기 때문에 제작 방식 자체가 달라지게 되었습니다. 직접 만들 생각보다 스크립트를 찾아 헤매는 기현상이 발생하기도 했습니다.

이렇게 제작 방식이 달라진 이유는 알만한 자체에 있습니다. 연구자들이 많이 사라져서 그런 것이기도 하지만 툴의 실행 구조 방식이 이벤트 에디터를 통해 작성된 스크립트를 대신 실행시켜주는 구조로 바뀌었기 때문에 스크립트를 수정해야만 고칠 수 있는 것들이 생겨났기 때문입니다.

하지만 이렇게 이벤트 에디터가 스크립트를 대신 작성해주는 이런식의 방식은 세밀하지 못합니다. 언젠가 게임제작에 한 계점이 찾아올 확률이 높아집니다. 이유는 에디터가 제공하는 툴을 벗어나지 못하기 때문입니다. 하지만 스크립트에선 위와 같이 조금 더 효율적으로 일을 처리할 수 있습니다. 배우기는 까다롭지만 간단한 스크립트 명령은 누구나 익힐 수 있기 때문에 알고 있으면 활용하기 좋습니다. 간단한 스크립트 명령에 대해서는 알고 있어야 하는 것이 RPG Maker VX Ace에서의 새로운 제작 방식이기 때문입니다.

본론으로 들어가자면 게이지바는 Move Picture 커맨드에 의해 변수에서 설정한 좌표에 따라 서서히 움직이게 되어있습니다. Y좌표 변수는 이전 이벤트에서 이미 설정해준 5번 변수를 설정해놓으면 되지만 X좌표는 체력값에 따라서 바꿔어야 하기 때문에 X좌표값을 실시간으로 계산하여 바꿔줘야 합니다. 계산식은 아래와 같습니다.

$$HP \text{ 비율} = (\text{현재 HP} * \text{게이지바의 가로길이}) / \text{최대 HP}$$

HP 비율은 위와 같은 공식에 의해 산출할 수 있습니다. 게이지바는 체력이 둘째 되면 화면의 좌측으로 이동해야 합니다. 이동을 해야하는 총 거리는 게이지바의 가로길이만큼입니다.

게이지바의 가로길이는 150픽셀입니다. 따라서 이 공식에 의해 최종값은 무조건 0 이상 150 이하가 나오게 됩니다. 이유는 HP가 150등분되었기 때문입니다. 따라서 체력이 아무리 많아도 체력비율은 0 이상 150 이하라는 범위에 있게 됩니다.

게이지바는 체력이 둘째 되면 화면의 좌측으로 이동해야 합니다. 이동을 해야하는 총 거리는 게이지바의 가로길이만큼입니다. 게이지바가 시작점부터 체력이 둘째 되면 서서히 좌측으로 이동을 해야 하기 때문에 HP비율 - 게이지바의 가로길이를 해줍니다. HP비율은 체력이 가득 찼을 땐 게이지바의 가로 길이와 같기 때문에 결과는 게이지바의 가로길이 - 게이지바의 가로길이가 되어 결과는 0이 됩니다. 따라서 체력이 가득차있으면 게이지바는 시작점 그대로에 있게 되는 것입니다.

$$\text{HP바의 X좌표} = \text{게이지바의 시작점} + \text{HP비율} - \text{게이지바의 가로길이}$$

공식으로 도출하면 위와 같습니다. 게이지바의 시작점부터 ($\text{HP비율} - \text{게이지바의 가로길이}$) 가 더해집니다. 이 계산 부분은 체력이 가득찼을 땐 0이고, 체력이 둘째 되면 마이너스값으로 변하게 됩니다. 결국 시작점이 바뀌면서 체력이 둘째 되는 것처럼 보이게 되는 것입니다.

이렇게 계산된 값을 Move Picture 커맨드를 이용하여 변수를 이 그림의 좌표로 등록해주면 게이지바가 성공적으로 이동을 마치게 됩니다.

마력 게이지바의 이동

원리를 이해했다면 마력 게이지바도 수월하게 만들 수 있습니다. 달라진 공식이 없기 때문에 체력에서 만든 것을 복사하여 그저 붙여넣기 해준 후에 변수만 바꿔주면 되기 때문입니다.

General Settings

Name:	Trigger:	Condition Switch:
MP	None	...

Contents

```

@>Control Variables: [0013:MP비율] = [Eric]'s MP
@>Control Variables: [0013:MP비율] *= Variable [0004:게이지바의 가로길이]
@>Control Variables: [0013:MP비율] /= [Eric]'s MMP
@>Control Variables: [0010:MP바의 X좌표] = Variable [0003:게이지바의 시작점]
@>Control Variables: [0010:MP바의 X좌표] += Variable [0013:MP비율]
@>Control Variables: [0010:MP바의 X좌표] -= Variable [0004:게이지바의 가로길이]
@>Move Picture: 2, Upper Left (Variable [0010][00006]), (100%,100%), 255, Normal, @10
@>

```

▲ MP

공식도 체력 게이지바와 같습니다.

MP 비율 = (현재 MP * 게이지바의 가로길이) / 최대 MP

MP바의 X좌표 = 게이지바의 시작점 + MP비율 - 게이지바의 가로길이

이 공식을 변수의 조작을 통해 설정해준 후 Move Picture 로 정지된 그림을 이동시켜주면 됩니다. 저는 이동에 10프레임 이 소요되게 해놓았습니다. 그래서 조금 더 부드럽게 이동을 하게 됩니다.

경험치 게이지바의 이동

경험바도 동일하게 만들 수 있습니다. 하지만 경험치의 경우에는 변수의 조작 에디터에 현재 레벨에서의 최대 경험치라는 변수가 없어서 기본 이벤트 에디터로는 변수를 조작할 수 없다는게 문제입니다. 그래서 2K시절에는 레벨이 한 단계 더 높은 가상의 캐릭터를 하나 만들어서 이를 활용하는 방식을 사용했지만 이젠 그럴 필요가 없어졌습니다. RPG Maker VX Ace에서는 스크립트를 사용할 수 있기 때문에 필요한 변수를 직접 스크립트로 만들어줄 수 있기 때문입니다.

General Settings

Name:	Trigger:	Condition Switch:
EXP	None	...

Contents

```

@>Script: actor = $game_party.members[0]
:   : $game_variables[15] = actor.max_level? ? actor.current_level_exp : actor.exp -
:   : actor.current_level_exp
:   : $game_variables[16] = actor.max_level? ? actor.current_level_exp : actor.next_level_exp
:   : - actor.current_level_exp
@>Control Variables: [0014:EXP비율] = Variable [0015:현재 EXP]
@>Control Variables: [0014:EXP비율] *= Variable [0004:게이지바의 가로길이]
@>Control Variables: [0014:EXP비율] /= Variable [0016:최대 EXP]
@>Control Variables: [0011:EXP바의 X좌표] = Variable [0003:게이지바의 시작점]
@>Control Variables: [0011:EXP바의 X좌표] += Variable [0014:EXP비율]
@>Control Variables: [0011:EXP바의 X좌표] -= Variable [0004:게이지바의 가로길이]
@>Move Picture: 3, Upper Left (Variable [0011][00007]), (100%,100%), 255, Normal, @10
@>

```

▲ EXP

이 스크립트는 현재 EXP 와 최대 EXP 를 계산하여,
각각 \$game_variables[15], \$game_variables[16] 변수에 저장을 하고 있습니다.
삼항 연산자라는 것을 통해 플레이어의 레벨이 최고 레벨에 있는지 확인하여
값을 분기시켜주고 있습니다.

```

1 actor = $game_party.members[0]
2 # 현재 EXP
3 $game_variables[15] = actor.max_level? ? actor.current_level_exp : actor.exp - actor.current_level_exp
4 # 최대 EXP
5 $game_variables[16] = actor.max_level? ? actor.current_level_exp : actor.next_level_exp - actor.current_level_exp

```

\$game_variables[변수의 번호] 는 변수의 조작을 통해 생성되는 변수들과 같습니다. 따라서 15번과 16번 변수에 현재 경험치와 최대 경험치값이 저장되어있으므로 이제 아래 공식을 제대로 활용할 수 있게 되었습니다.

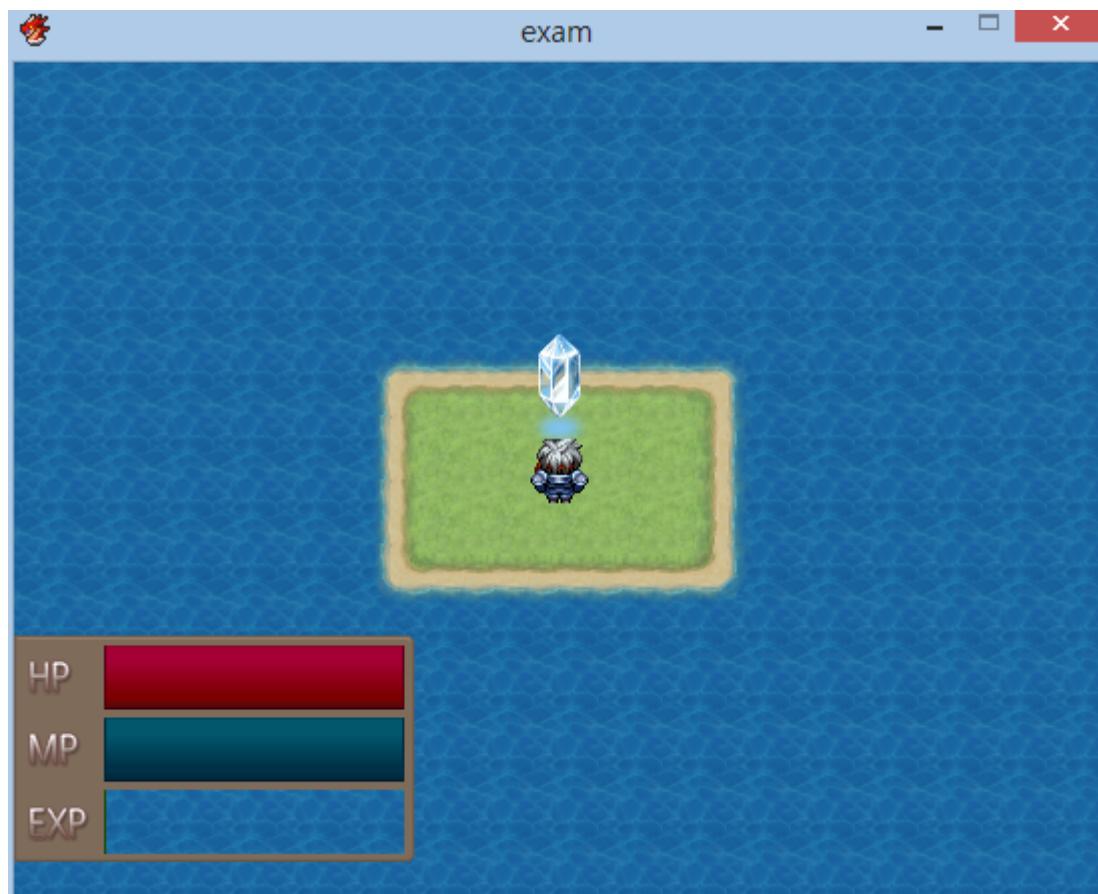
EXP 비율 = (현재 레벨의 EXP * 게이지바의 가로길이) / 최대 EXP
EXP바의 X좌표 = 게이지바의 시작점 + EXP비율 - 게이지바의 가로길이

이 공식을 그대로 적용하면 경험치 게이지바가 성공적으로 이동을 하게 됩니다. 전에 만들어둔 병렬 이벤트에 자동으로 반영되기 때문에 그대로 테스트 플레이 게임을 돌려볼 수 있습니다. 테스트 플레이를 하게 되면 아래 동영상과 같이 표시됩니다.

결과



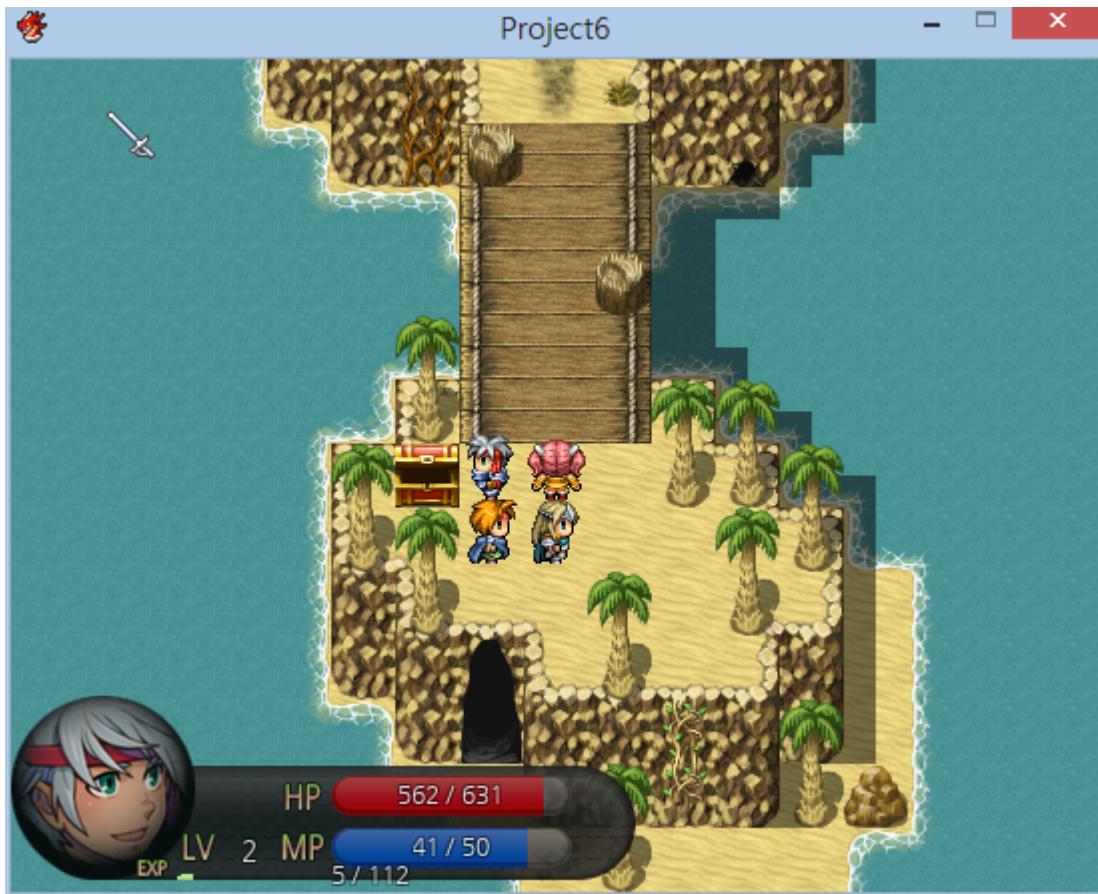
▲ 테스트 플레이 영상입니다



▲ 틀의 Y좌표를 바꿔도 정상 동작합니다.

스크립트로 게이지바 만들기

RPG Maker VX Ace에서는 루비 스크립트라는 강력한 스크립트 언어를 제공하고 있는데 화면에 스프라이트를 그릴 수 있는 API도 제공합니다. 이 API를 게임 라이브러리라고 부르는데 이 API를 활용하면 더 화려한 게이지바를 그릴 수도 있습니다. 하지만 스크립트 유저가 되려면 루비라는 객체 지향 프로그래밍 언어를 알아야 하는데 이를 익히려면 별도의 노력이 필요합니다. 그래서 당장은 스크립트로 무언가를 만드는 것이 불가능 하겠지만 스크립트의 활용은 스크립트를 몰라도 누구나 할 수 있기 때문에 스크립트로 게이지바를 만드는 방법 또한 찾아볼만 합니다.



▲ 필자가 직접 만든 HUD 스크립트

RPG Maker VX Ace로 게임을 만드는 게임 제작자들은 자신이 직접 만든 스크립트를 커뮤니티 사이트나 블로그에 공유하는 일이 많습니다. 그래서 조금만 찾아보면 이런 스크립트를 어렵지 않게 구할 수 있습니다. 또한 RPG Maker VX Ace의 스크립트들은 기능 별로 메소드가 잘 나뉘어있고 관련 문법도 제공하기 때문에 호환성 문제도 크게 없었습니다. 그래서 스크립트도 하나의 부품처럼 추가해서 사용할 수 있습니다.

[08강] 기본 전투의 처리 1편 (기초편) - RPG 만들기 VX Ace 강좌

시작하기

2014/12/28 22:34

<http://blog.naver.com/biud436/220223062236>

RPG Maker VX Ace 왕초보탈출 8

"RPG Maker VX Ace - 기본 전투의 처리"

왕초보탈출 강좌의 8편은 RPG 만들기 VX Ace에서 기본적으로 제공하고 있는 기본 전투 처리 시스템에 대해서 알아보는 시간입니다. 이 강좌에서 설명하는 모든 개념을 다 알아야 하는 것은 아닙니다. 요즘 게임의 추세를 살펴보면 툴에서 기본으로 제공하는 턴제 전투를 활용한 게임들이 거의 없다는 것을 알 수 있습니다. 게다가 턴제 전투는 기본적으로 제공되는 것이기 때문에 옛부터 액션 롤플레잉 게임의 전투 스타일보단 여러가지 면에서 괄시 당하는 분위기가 있었습니다. 모듈화되어 있어서 전투를 쉽게 만들 수 있지만 이를 활용하여 밸런스를 맞추고 본인이 기획한 대로 지루하지 않게 전투와 레벨을 구성하여 진행하는 것은 매우 난이도 있는 일이기 때문입니다.

요즘에 나오는 RPG들은 거의 모두 액션 전투 스타일을 실시간으로 지향하는 게임들입니다. 필드에서 실시간으로 전투를 진행하고 적을 처치하기 위해서 알아야 할 마법 스킬과 아이템들 그리고 그 응용 방법도 너무나 많습니다. 우리는 그런 게임들을 플레이하면서 자라왔으며 또 즐겨왔습니다. 지금 이 시간에도 누군가가 그 방식의 게임을 플레이 하고 있을 것입니다. 그런 환경에서 접하는 턴제 전투는 낯선 전투 방식으로 느껴질 수 밖에 없습니다.

턴제 전투는 옛날 RPG의 전유물이 아닙니다. 현재 서비스되고 있는 게임들 중에서도 턴제 전투 방식의 게임들이 있습니다. 어떤 장점이 있길래 턴제 전투를 사용하는 것일까요? 턴제 전투 방식은 왜 사용하는 것이며 그 게임들의 특징은 무엇인지 그리고 턴제 이러한 전투 방식이 어디서 어떻게 왜 등장하게 되었는지 그 배경부터 알아보도록 하겠습니다.

그럴려면 RPG의 유래부터 알아봐야 합니다.

1. RPG의 유래

RPG는 서양에서 1970년대 최초의 TRPG인 던전 앤 드래곤 시리즈(이하 D&D)를 시작으로 시작된 게임 방식입니다. 영어로는 Role-Playing Game 이라고 부르고 한국어로 해석하면 역할 분담 놀이라는 뜻입니다.

RPG는 우리가 아는 컴퓨터가 등장하기 이전에는 실제 사람들이 테이블에 둘러앉아 진행하는 일종의 보드게임과 같은 형태로 진행되는 게임이었습니다.

게임의 규칙이 담겨있는 룰북을 보면서 진행하는 형태의 게임이며 플레이어들은 지속적으로 진행자와 대화를 나누게 됩니다. 플레이어는 그 책을 보고 캐릭터를 만들기도 하고, 진행자는 준비된 시나리오에 따라 게임 세계에서 벌어지고 있는 일들을 나레이션으로 설명합니다. 때로는 주사위를 굴려서 전투의 결과를 알려주기도 해야 했고, 아이템을 주거나 퀘스트를 주기도 해야 했습니다. 플레이어들은 게임 진행자의 설명에 따라 실제 영웅이 된 것처럼 대화하거나 전투를 하기도 했고 퀘스트 진행하거나 알맞는 행동을 해야 했습니다.

진행자의 역량이 중요할 수 밖에 없었고 오프라인에서 매번 모여야 하기 때문에 단점이 있었습니다. 1970년대 북미에서 이런 던전 앤 드래곤 시리즈(TRPG)가 컴퓨터로 구현되기 시작하면서 울티마, 마이트 앤 매직, 위저드리 시리즈와 같은 RPG의 시초가 되는 게임들이 등장했습니다. 그러다가 위저드리 시리즈가 콘솔 게임이 주를 이루는 일본시장에 큰 히트를 치게 되면서 그 후에 탄생하는 많은 일본 RPG에 영향을 주게 되면서 드래곤 퀘스트 같은 일본식 RPG가 탄생합니다.

2. 일본식 RPG란 무엇인가?

일본은 콘솔 게임 시장이 발달했고 우리나라와는 시장 환경이 다른 곳입니다. 이 때문에 게임도 서양과는 다른 방향으로 발전을 해왔습니다. 일본에서 만든 RPG들은 서양에서 만든 RPG들과는 큰 차이가 있습니다. 서양식 RPG는 자유도가 높고 NPC에게 말을 걸어서 퀘스트를 받거나 필드에서 바로 전투를 벌여 몬스터를 사냥하거나 진행 방향이 다양한데 반해 일본식 RPG는 스토리 중심의 어드벤쳐 식으로 진행이 되는 것이 많아 스토리가 일방향적이고 자유도가 상대적으로 떨어집니다. 또한 맵 상에서 적과 만나면 턴마다 명령을 지정하여 싸우는 방식이기 때문에 복잡한 전투 시스템이 필요하지 않아 전투 시스템 제작이 더 간편하며 이러한 전투 방식은 콘솔 기반의 게임에도 적합합니다. 대표적으로 드래곤 퀘스트, 파이널 판타지, 포켓몬스터 시리즈들과 같은 유명한 게임들이 있습니다. 이런 일본식 RPG들은 세계관도 매우 독특하며 인상적입니다.

RPG Maker 시리즈도 일본에서 제작된 게임제작툴입니다. 그래서 기본적으로 일본식 RPG의 기본 틀 위에서 시작합니다. 그래서 기본적으로 턴을 주고 받으며 전투를 진행하는 턴제 전투 시스템을 제공합니다. 미리 만들어진 시스템이 전투를 진행하고 조율합니다.

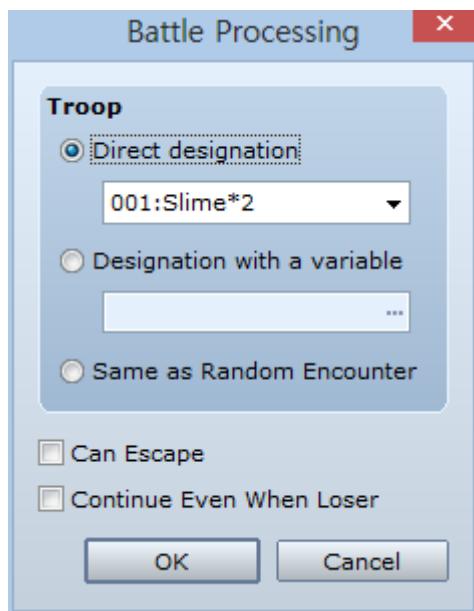
이 전투 시스템은 미리 만들어져 있습니다. 그렇기 때문에 시스템 상으로 배경 그래픽을 그리는 명령을 내리거나 매 턴마다 플레이어에게 명령을 받거나 입력 받은 데이터를 계산하여 데미지를 줘야 하는 일을 할 필요가 없습니다.

또한 그 데이터에 따라 마법 애니메이션이 발생하기도 하고 소리도 재생되기도 해야하고, 전투가 끝나면 보상도 줘야하며 경험치도 상승시켜야 하고 마법도 새로 배우게 해야 합니다. 그러나 이 모든 과정을 제작하기 위해 노력할 필요는 없습니다. 이미 모두 제작이 완료되어있기 때문입니다.

하지만 여러분들이 원하는 이야기를 펼쳐나갈려면 이 전투 시스템을 어떻게 조작하는지 알아야 합니다. RPG Maker 툴에서는 전투 시스템 조작을 위해 '데이터베이스'라는 에디터를 제공하고 있습니다. 데이터베이스를 통해 전투 구동을 위해 사용되는 데이터값을 조작할 수 있습니다.

3. 텐제 전투 시스템의 기본 구성

RPG Maker 툴에선 기본 전투를 시작하는 방법도 매우 간단합니다. 전투 시스템이 스크립트로 이미 만들어져있기 때문에 복잡한 프로그래밍이나 이벤트가 필요하지 않습니다. 과정을 생각할 필요 없이 미리 준비된 데이터를 불러와서 전투를 바로 시작할 수 있습니다. 전투의 처리라는 이벤트를 통해서 전투를 시작할 수 있습니다. 이 인터페이스 창은 매우 단순하게 되어있습니다. 단지 적 그룹(Troop)을 골라주기만 하면 되기 때문입니다. 아래 스크린샷을 참조하면 슬라임이 등장하게끔 설정되어있다는 것을 알 수 있습니다. 이 인터페이스를 통해 전투가 시작되게끔 할 수 있습니다.



▲ 전투의 처리 이벤트 커맨드

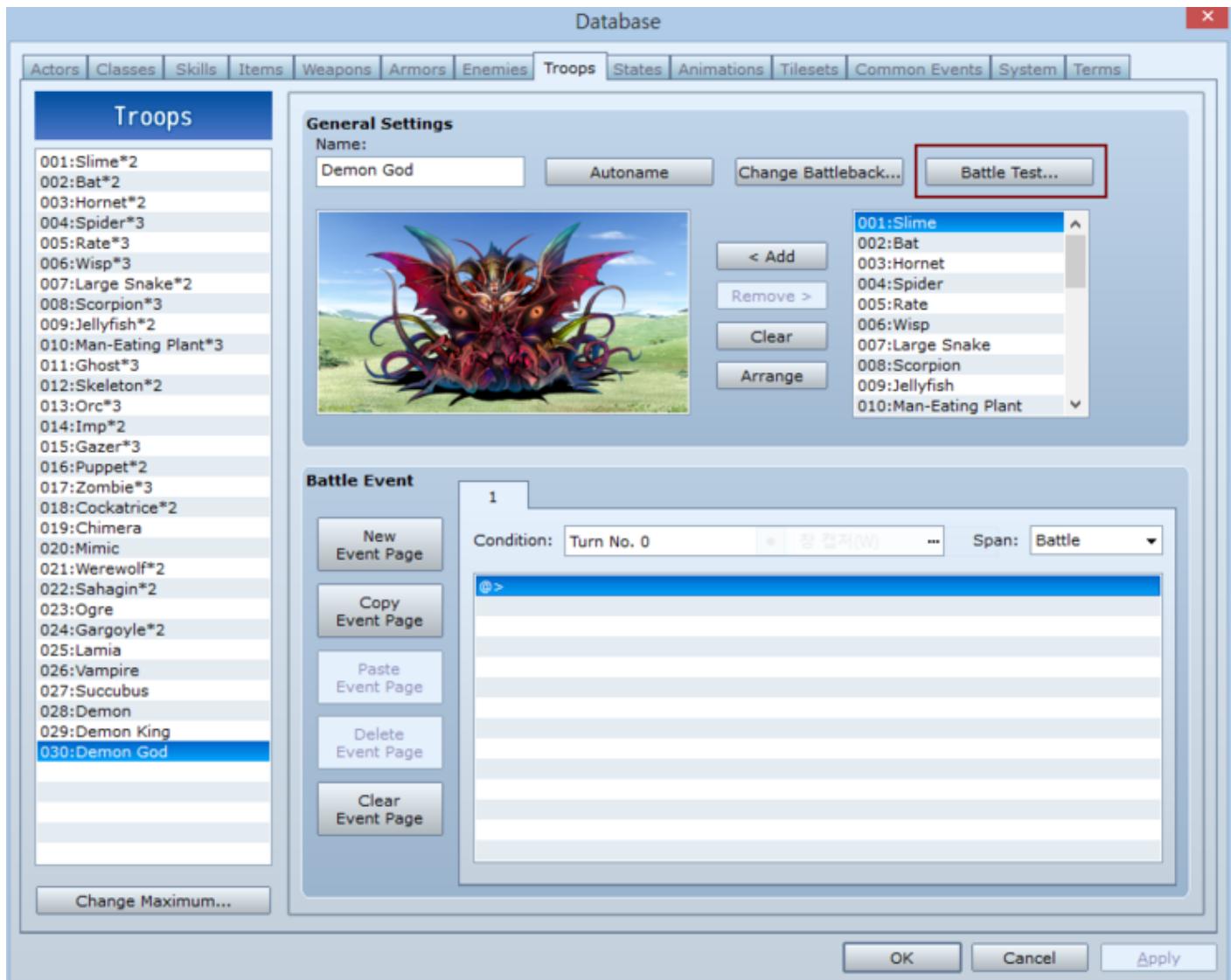
Battle Processing은 이벤트 편집창에서 이벤트 커맨드 리스트들 둘러보시다보면 발견할 수 있습니다. 출현 대상 적그룹은 리스트박스나 변수를 통해 직접 설정할 수도 있고 맵의 엔카운터 설정을 따를 수도 있습니다.

위 이벤트 커맨드를 만들게 되면 슬라임과의 전투가 시작될 것입니다. 이외에도 테스트 목적 차원으로 제공되는 기능도 있습니다. 데이터베이스의 적그룹 탭에 있는 배틀 테스트 버튼을 누르면 전투 테스트도 가능합니다.

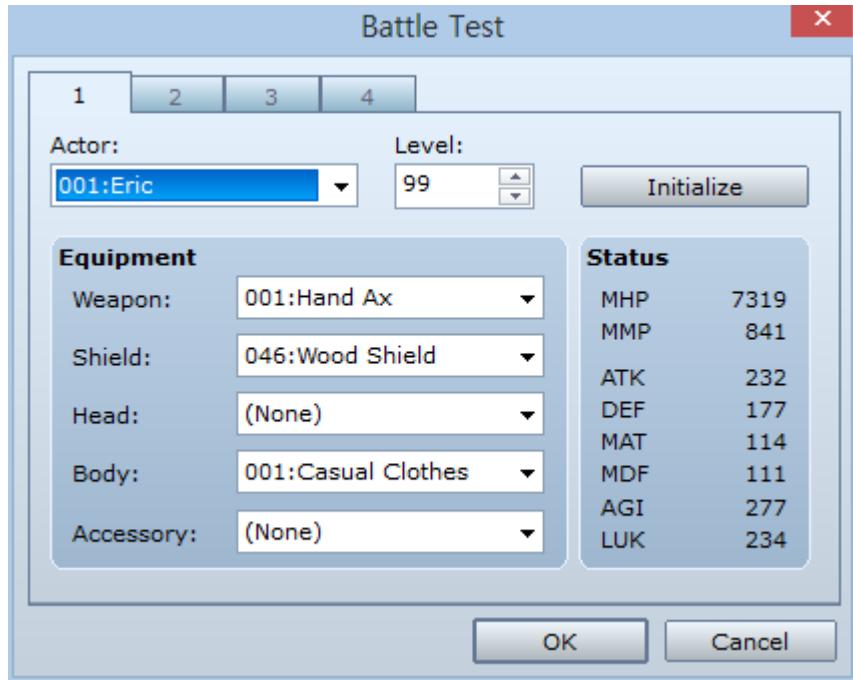
전투는 지난 맵 강좌 시간에서 배웠던 엔카운터 조건에 따라서 발생할 수도 있고 이벤트 커맨드를 통해 발생시킬 수도 있으며 데이터베이스에서 배틀 테스트를 하기 위해 의도적으로 발생시킬 수도 있습니다.

전투를 테스트하는 목적은 밸런스가 잘 맞는지 확인을 하기 위해서입니다. 롤플레잉 게임에서의 전투는 굉장히 중요한 요소 중 하나입니다. 레벨 디자인이라고도 부르는데 전투가 너무 일정하고 특색없이 벌어진다면 플레이어들은 지루함을 느끼게 될 것입니다. 턴제 전투는 턴을 돌아가면서 전투가 진행되기 때문에 지루함을 쉽게 느낄 수 있습니다. 따라서 전략을 잘 짜지 않고 대충 진행하면 게임이 어렵게 느껴지게끔 밸런스를 잘 맞추는 것도 중요합니다.

여러분들 중엔 분명 기본적으로 설정되어있는 데이터베이스 능력치를 바꾸지 않고 그대로 사용하는 분들도 있을 것입니다. 그러나 기본적으로 설정되어있는 데이터베이스 능력치들로 전투를 진행하는 일은 없길 바랍니다. 이 경우에는 대부분 플레이어가 별 생각없이 어택을 해도 적이 금방 나가 떨어지게 되고 그 상황이 반복되고 이변이란 없어서 난이도가 쉽기 때문에 전투에서 지루함을 느끼게 됩니다. 턴제 전투라 상관없다고 생각할 수 있겠지만 지루한 전투는 게임의 재미를 떨어뜨리는 일이기도 합니다.



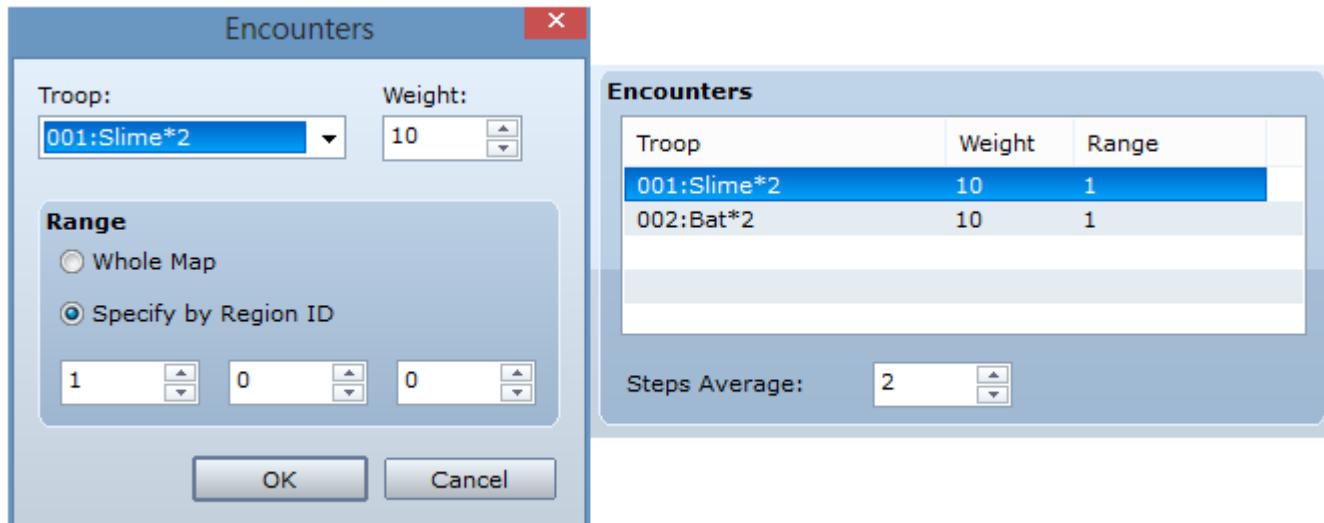
전투 테스트(Battle Test)는 전투 구성에 필요한 최소한의 스크립트만 구성하여 로드하는 경량화된 전투 테스트 전용 모드입니다. 기존에 데이터베이스에서 설정한 값과 달리 사전에 레벨과 무기 데이터를 교체할 수 있습니다. 게임이 시작되기 전에 BT_라는 접두어가 붙은 Data/BT_Actors.rvdata2 과 같은 형태로 데이터베이스 스크립트를 복제하고 이 스크립트들을 전투 테스트에 이용합니다. 이는 원본 데이터베이스를 보존하고 전투 테스트에 필요한 레벨과 무기 조건을 충족시키기 위해서입니다.



파티원들의 능력치를 확인할 수 있고 장비를 원하는 것으로 바꿀 수 있습니다. OK 버튼을 누르면 전투가 시작되게 됩니다.

이외에도 전투의 시작 조건이 하나 더 있는데 바로 엔카운터라는 기능입니다. 자동으로 전투가 시작되는 엔카운터 설정이라는 기능에 의해서도 시작되기도 합니다. 이 기능은 이미 2강에서 배운적이 있기 때문에 잘 알고 있을 것입니다.

포켓몬스터라는 게임에서는 몬스터가 없어도 필드에서 자동으로 전투가 발생합니다. 이런 방식의 전투 시작 방법은 엔카운터라고 부릅니다. 엔카운터 시스템은 자동으로 게임 무대가 전투 필드로 변하기 때문에 주인공들은 순식간에 게임 세계와는 동떨어진 전투 필드로 툭 떨어지게 됩니다. 또한 이동 중에 갑자기 발생하기 때문에 게임의 맥이 끊기게 되고 게임 진행 속도가 늦춰진다는 단점이 있습니다. 동떨어진 세계에서 전투를 하기 때문에 빨리 게임 세계로 돌아가고 싶다는 느낌이 생기지만 이러한 점을 잘 이용하면 오히려 긴장감을 줄 수도 있습니다. 그러나 잘못하면 지루함을 느끼게 될 수도 있으니 적당한 선을 맞춰줘야 합니다.



▲ 맵 속성에서 엔카운터를 설정한 모습

3가지 방법 중 하나로 전투가 되면 아래와 같은 화면이 뜨게 될 것입니다. 기본 전투 시스템은 아래와 같은 모습을 하고 있습니다. 상단에는 적 그룹이 스프라이트로 표시되고 하단에는 전투를 돋기 위한 커맨드 창이 나열되어있습니다. 커맨드 창을 통해서 파티원들에게 공격 명령이나 방어 명령을 내릴 수 있습니다. 이외에도 특수 능력 사용이나 마법 사용 그리고 아이템 사용등의 명령도 내릴 수 있습니다.



1번 윈도우에는 커맨드 버튼 2개가 나열되어 있습니다. 커맨드는 한글로 명령이라는 뜻으로 결정키를 누르게 되면 반응하는 버튼기능입니다. 결정키는 키보드 상에서 Enter, Space Bar, Z 키를 뜻합니다. 결정키를 누르면 해당 버튼에 입력되어 있는 내용이 수행됩니다. Fight 커맨드 버튼을 누르게 되면 1번 윈도우가 비활성화되고 3번 윈도우가 활성화되며 3번 윈도우에 있는 커맨드 버튼을 사용할 수 있게 됩니다. Escape 버튼은 탈출이라는 뜻으로 전투를 끝내고 맵으로 돌아가기 위한 기능입니다. 선제 공격을 받지 않았다면 100% 탈출이 가능하지만 몬스터로부터 선제 공격을 받았을 경우엔 탈출 확률에 의해 탈출할 수 있게 되어있습니다. 탈출 확률은 플레이어와 적 그룹 사이의 민첩력과 관계되어있습니다. 민첩력이 적보다 좋지 않으면 비율에 의해서 탈출 확률도 떨어지게 됩니다. 탈출 확률은 BattleManager 스크립트의 make_escape_ratio이라는 메소드에 정의되어있습니다. 탈출을 실패하면 탈출 성공 확률이 10% 상승합니다.



2번 윈도우에서는 현재 파티의 상태를 확인할 수 있습니다. 이 윈도우에는 체력(HP)와 마력(MP) 그리고 기술력(TP)이 게이지바와 텍스트 형태로 표시됩니다. 또한 현재 걸려있는 버프 또는 디버프 상태도 아이콘으로 확인할 수 있습니다.

HP는 Hit Points 와 Health Points의 약자입니다. 이 개념은 D&D 에서 유래된 개념으로 몬스터의 체력이나 아군의 체력을 나타내는 수치입니다. 당연한 소리이지만 적에게 공격을 당하면 체력이 줄어드는 것을 표현해야 하므로 이 수치도 줄어듭니다. 체력이 0이 되면 사망 상태가 됩니다.

MP는 Magic points의 약자입니다. Magic 은 마법이고 Points 는 힘을 뜻하는 말이기 때문에 한국어로는 마법력이라고 해석할 수 있습니다. 보통 줄여서 마력 또는 마나라고 불립니다. 마법력은 롤플레잉 게임에서 마법을 사용할 때 필요한 힘

입니다. 메이지 같은 마법 계열의 클래스들은 더 높은 MP를 가지고 있습니다. 이들은 마법 공격력을 바탕으로 적에게 공격을 가합니다. 이런 직업들은 마법력을 바탕으로 강력한 마법을 사용하기 때문에 마력이 없으면 힘을 발휘할 수 없습니다. 그래서 마력을 항상 신경써줘야 하는 직업들입니다. 턴제 전투 시스템에서는 매 턴이 끝나면 일정량의 MP를 회복됩니다. 회복률은 MRG라는 수치에 의해 결정됩니다. 그러나 이 수치가 낮으면 회복 속도도 더디기 때문에 MP를 회복시켜줄 수 있는 아이템으로 마나를 회복시켜줘야 할 수도 있습니다.

TP는 Technical Points의 약자입니다. 해석하면 기술력이라는 뜻입니다. 이 개념은 파이널 판타지라는 게임에서 유래되었습니다. 턴제 전투 시스템에서 마법을 사용하려면 MP 이외에도 일정량의 TP가 소비됩니다.

따라서 전투 초반에 TP를 반드시 획득해놓아야 합니다. TP는 적을 가격하거나 적에게 피격을 당할 시에 바로 획득할 수 있으며 매 턴이 종료시에도 획득합니다. TP는 100이 최대값입니다.

초기의 TP 수치는 0 ~ 25 이하의 난수로 결정되어지고 소비 TP값은 스킬탭에서 설정할 수 있습니다.

강력한 마법을 쓰려면 많은 TP가 필요하기 때문에 초반에 TP 채울 수 있는 마법과 아이템들도 만들어놓을 수도 있습니다. TP의 충전값은 TCR이라는 비율값에 의해서 결정됩니다. TP는 매 턴이 끝나면 TRG(TP 회복률)이라는 수치에 의해 일정량이 채워집니다.

3번 윈도우는 공격 스킬을 선택하거나 방어를 하거나 아이템을 고를 수 있는 윈도우입니다. 마법을 사용할 수 있는 직업은 마법 커맨드가 추가로 주어집니다. 그러나 그 이외의 직업들은 Special이라는 스킬 타입만 존재합니다. 스킬 타입은 데이터베이스 Terms 탭의 Skill Types라는 리스트 박스에서 추가할 수 있습니다. Terms 탭엔 속성, 무기 타입, 방어구 타입을 추가하고 줄일 수 있는 기능이 있는 리스트박스도 있습니다.

전투는 데이터베이스에 설정된 데이터를 바탕으로 만들어집니다. 실제로 확인해보면 아무것도 설정하지 않은 상태에서도 모든 것이 정상적으로 진행되는 것을 알 수 있습니다. 그 이유는 데이터베이스에 이미 입력된 데이터들이 있어서 그런 것입니다.

이렇게 기본적으로 미리 입력된 데이터만으로도 게임을 만들 수 있습니다. 그러나 아무것도 조작하지 않은 상태의 주인공은 매우비범한 용사로 설정되어있습니다. 그야말로 판타지 세계의 몬스터들을 모두 도륙하고 다닐 수 있는 초능력자 용사로 설정되어있기 때문에 모든 전투가 매우 시시하게 끝나버릴 것입니다. 그렇게 되면 턴제 전투의 매력이 사라지게 되고 플레이어들은 쉽게 질려버리게 됩니다.

여러분들의 게임에 나와야 하는 주인공이 초능력자 용사라면 그냥 이용해도 되겠지만 그게 아니라면 아무래도 기본 데이터를 이용하는 것은 무리입니다. 그래서 전투를 만들기 전엔 사전 설정이 필요합니다. 기본적으로 플레이어의 기본 능력치, 적의 능력치, 스킬을 어떻게 설정하고 바꿀 수 있는지 정도는 알고 있어야 전투를 잘 다룰 수 있게 됩니다.

이러한 일은 모두 데이터베이스에서 할 수 있습니다. 데이터베이스는 많은 탭들로 구성되어있는데 이러한 데이터들은 모두 전투 시스템과 관련되어있습니다. 전투 후에 아이템을 보상으로 주려고 해도 마찬가지입니다. 보상으로 받은 아이템과 골드는 추후 전투에도 영향을 끼치기 때문에 다음 전투를 봉괴시킬 위험이 있습니다.

기본 데이터를 그대로 쓰게되면 게임 밸런스는 포기해야 될지도 모릅니다. 게임 제작자는 플레이어가 전투를 통해 어떤 아이템을 받았는지도 모를 것입니다. 플레이어가 다음 던전을 클리어하기 위해 어떤 것을 준비해야 하는지도 모를 것입니다. 플레이어는 던전 클리어에 필요한 물품을 미리 공수하기 위해 마을 상인들을 만나볼 것입니다. 그러나 그 물건을 판매하지 않을 수도 있겠죠. 판매한다해도 밸런스가 이미 봉괴되어있을 수도 있습니다.

그래서 데이터베이스의 조작은 중요합니다. 데이터베이스에선 무엇을 할 수 있고 어떤 데이터를 바꿀 수 있는지 그것도 잘 알아야 합니다. 아무래도 데이터베이스 기본 데이터를 그대로 쓰기에는 밸런스가 봉괴될 위험성이 크기 때문입니다.

직접 데이터베이스를 조작해야 전투에서 필요한 적절한 데이터를 만들 수 있습니다.

전투에 필요한 데이터는 대부분 플레이어의 현재 능력치 정보입니다. 전투 시스템에서는 현재 플레이어의 능력치를 종합해서 가져오게 됩니다. 능력치는 직업(Class) 탭에서 최초로 생성되는 정보이지만, 각 능력치들은 착용하고 있는 아이템이나 특성에 따라 유동적으로 달라집니다. 그래서 최종 능력치는 현재 플레이어의 상황에 따라 달라집니다. 그러나 쪼꼬르툴답게 최종 능력치의 합산도 스크립트에서 대신 처리해줍니다.

알만툴이 능력치 데이터를 간단하게 보거나 조작하는 기능을 제공하지 않았다면 우리들은 아마 지금도 엑셀을 공부하고 있었을지도 모릅니다. 데이터베이스라는 기능이 없었다면 직접 온갖 계산식을 만들어야 하고 비율을 맞추기 위해 공식을 짜야하며 스크립트 파일들도 직접 만들어야 했을 겁니다. 또한 레벨 별로 수치를 다르게 하기 위해 온갖 비율을 만들어야 하는데 그것을 하는 데에 대부분 시간을 투자 했을 것입니다. 아마 전문가가 되었을 수도 있습니다.

하지만 RPG Maker VX Ace에서는 데이터를 만들고 조작하는데 유용한 데이터베이스라는 에디터를 제공하기 때문에 굳이 위 과정의 전문가가 되기 위한 시간을 투자할 필요가 없습니다. 이미 복잡한 부분은 다 만들어져있기 때문에 데이터베이스에서 간단히 설정만 해주면 되는 것들이 많습니다. 하지만 지금은 그것도 처음 보기 때문에 생소해서 어렵게 느껴질 뿐인 것입니다.

참고

- [참고 영상\(다음tv팟\)](#)

[09강] 기본 전투의 처리 2편 (응용편) - RPG 만들기 VX Ace 강좌

시작하기

2015/02/04 21:22

<http://blog.naver.com/biud436/220263027161>

RPG Maker VX Ace 왕초보탈출 9

"RPG 만들기 VX Ace - 기본 전투의 처리 2편 (응용편)"

안녕하세요. 러닝은빛입니다.

이전 강좌에서는 RPG 의 정의에 대해서 알아보았습니다. 또 기본 전투 처리 시스템이 왜 터제 전투인지 장점과 단점에 대해서도 알아보았으며 기본 인터페이스 구성이 어떻게 되어있는지도 개략적으로 살펴보았습니다. 그런 내용들 자체는 흥미로웠지만 터제 전투 시스템을 구성하는데 실질적으로 마음에 와닿거나 도움이 되는 내용들은 아니었습니다. 실전 중심의 강좌를 펼치다 뜬끔없이 갑자기 이론 위주의 강좌가 이어지니 그런 괴리감이 생긴 것입니다. 하지만 안타깝게도 이번에 하게 될 강좌도 지난 강좌와 마찬가지 성격의 강좌입니다. 사실 이론이라고 해봤자 그렇게 어려운 내용도 아닙니다. 전투 시스템을 직접 만들고 처리하는 기술을 가르치는 강좌가 아니기 때문입니다. 전투 시스템은 이미 개발사가 만들어놓은 상태입니다. 그렇기 때문에 어려운 내용은 하나도 없습니다. 다만 생소한 단어들이 많아서 익숙하지 않을 뿐입니다.

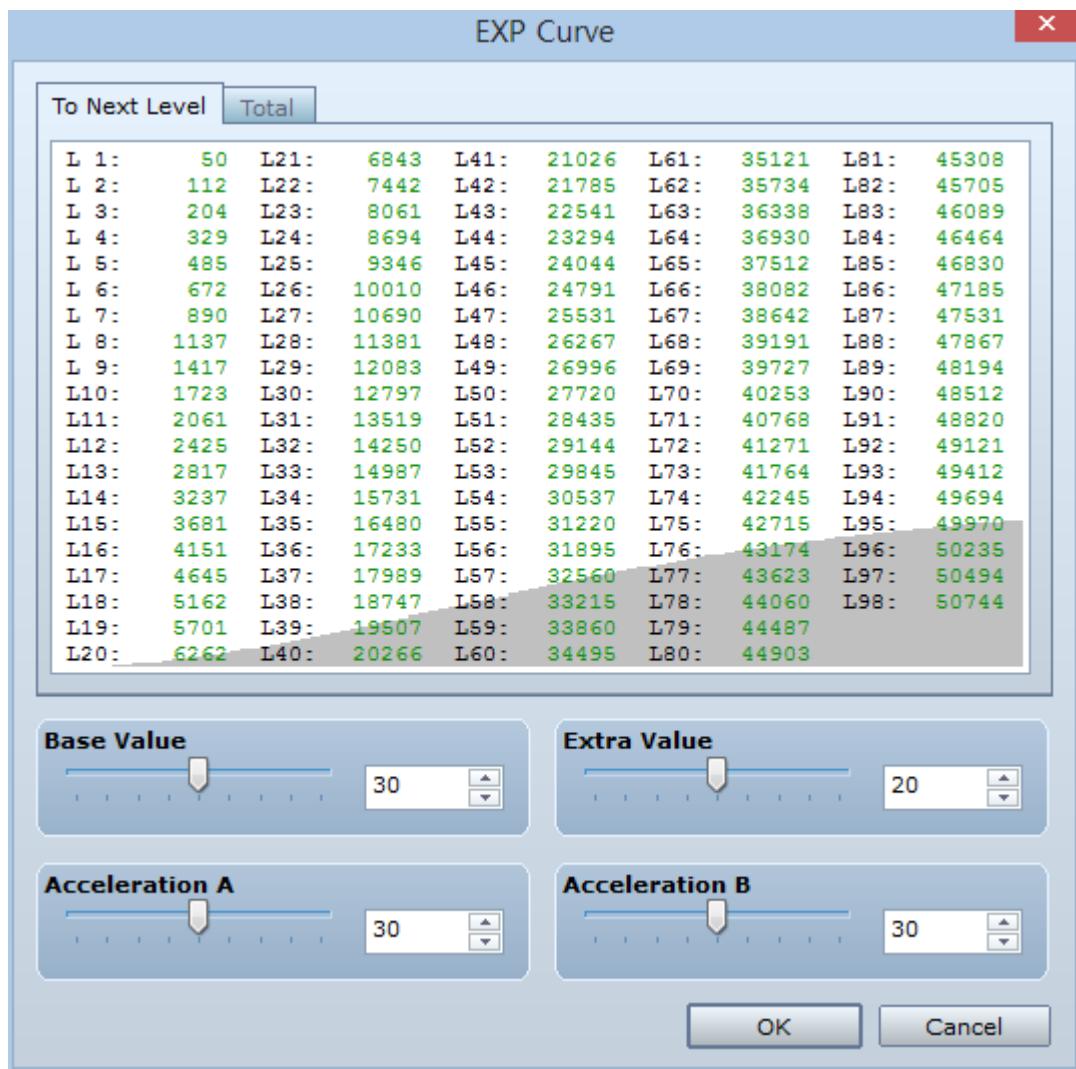
전투 시스템은 데이터베이스 활용의 끝판왕 격이라고 볼 수 있습니다. 모든 개념이 총 집합되어있습니다. 그래서 먼저 경험치 곡선 분석부터 들어가려고 합니다. 경험치 곡선은 자동 완성되는 부분이라 자세히 살펴보지 않으면 잘하는 분들도 그냥 넘어가는 부분입니다. 하지만 경험치 곡선에 적용되어있는 수식을 이해하면 다른 게임 제작툴을 이용하여 게임을 만들 때에도 같은 원리를 적용해 유용하게 사용할 수 있습니다.

1. 경험치곡선(Exp Curve)의 분석

주인공이 가지고 있는 기본 능력치와 확장 능력치들은 Classes 탭에 정의되어있습니다. 경험치는 Classes 탭에서 설정할 수 있는 경험치 곡선(Exp Curve)에 따라 정해집니다. 최대 경험치는 기본값, 보정치, 증가율 A값과 증가율 B값에 의해 최종적으로 결정됩니다.

The screenshot shows the RPG Maker MV game editor interface with the 'Classes' tab selected. On the left, a list of existing classes is shown: 001:Soldier, 002:Monk, 003:Paladin, 004:Spellblade, 005:Samurai, 006:Archer, 007:Thief, 008:Priestess, 009:Witch, and 010:Sage. The '001:Soldier' entry is highlighted with a blue background. The main workspace displays the configuration for the 'Soldier' class under the 'General Settings' tab. The 'Name:' field contains 'Soldier'. The 'EXP Curve:' field shows '[30, 20, 30, 30]' with a three-dot ellipsis button. Below this, the 'Parameter Curves' section shows eight triangular graphs for MHP (orange), MMP (blue), ATK (red), DEF (green), MAT (magenta), MDF (green), AGI (cyan), and LUK (yellow). The 'Skills' section lists two skills: 'Lv 5 Strong Attack' and 'Lv 10 Cleave'. To the right, a vertical sidebar titled 'Features' lists various options like 'Kind', 'Sp-Parameter', and 'Equip Weapon', with 'Equip Weapon' currently selected.

Classes 탭을 번역해서 직업 탭이라고 부르겠습니다. 우리가 알아볼 부분은 영어로 Exp Curve 라고 부르는 경험치 곡선 부분입니다. 전투 시스템을 다루지 않더라도, 경험치 곡선을 볼 줄을 알아야 합니다. 경험치 곡선도 벨런스를 맞추기 위해 중요한 개념 중 하나이기 때문입니다.

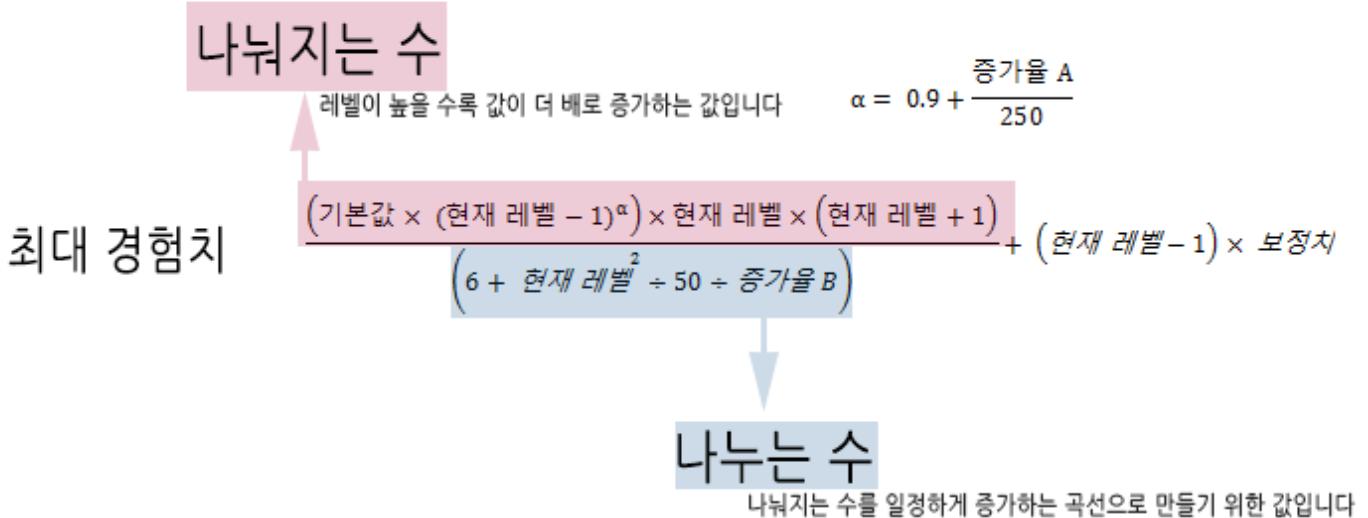


세부 창으로 들어가보면 먼저 위와 같은 장면을 보실 수 있게 됩니다. 데이터가 도출되어있는 표와 표 뒤에 숨어있는 잣빛으로 빛나는 그래프를 보실 수 있습니다. 하단에는 경험치 곡선의 데이터 값인 기본값(Base Value)과 보정치(Extra Value) 그리고 증가율 A(Acceleration A)와 증가율 B(Acceleration B) 수치를 조절할 수 있는 스크롤바와 업다운 컨트롤 상자가 제공됩니다. 경험치 곡선이 그려지는 세부 공식은 도움말에 아래와 같이 나와있습니다.

```
def exp_for_level(level)
  lv = level.to_f
  basis = @exp_params[0].to_f
  extra = @exp_params[1].to_f
  acc_a = @exp_params[2].to_f
  acc_b = @exp_params[3].to_f
  return (basis*((lv-1)**(0.9+acc_a/250))*lv*(lv+1)/
    (6+lv**2/50*acc_b)+(lv-1)*extra).round.to_i
end
```

▲ 도움말에 나와있는 최대 경험치 산출 공식

하지만 이를 해석하려면 루비 스크립트에서의 연산자 우선 순위와 결합 순서를 알아야 하고, 수식도 꽤 길고 알아보기가 힘들어서 눈으로 보고 판단하기에는 무리가 있다는 것을 알 수 있습니다. 그래서 스크립트에 정의되어있는 최대 경험치 산출 공식을 우리 인간이 알아볼 수 있는 수식으로 정리해보면 아래와 같은 수식으로 정리할 수 있습니다.



▲ 인간이 알아볼 수 있는 수준으로 다시 정리한 수식

위 수식도 인간이 알아보기에는 너무 길어서 일부 수식은 알파라는 기호로 치환되어있는데 위 수식은 크게 나눠지는 수, 그리고 나누는 수, 그리고 보정치로 분리할 수 있습니다.

먼저, 핵심 값으로 볼 수 있는 나눠지는 수를 보겠습니다. 나눠지는 수에서 알파값은 증가율(Acceleration) A에 의해 결정되는 배율값입니다. 증가율 A 값에 따라 최소 0.94에서 최대 1.1 범위 내로 결정됩니다. 증가율 A 값이 1 올라갈 때마다 최종 알파값은 0.004씩 증가하고, 현재 레벨 - 1이라는 값은 알파값에 의해 늘어나기도 하고 줄어들기도 합니다. 이렇게 만들어진 배율값에 기본값을 곱하면 최종적으로 나눠지는 수가 만들어지게 됩니다. 나눠지는 수는 배율에 의해 결정되는 값이기 때문에 레벨이 증가할 수록 배로 증가합니다.

		최소	최대	기본값
기본	basis	10	50	30
보정치	extra	0	40	20
증가율A	acc_a	10	50	30
증가율B	acc_b	10	50	30

레벨	연산 결과	보정치	최종 경험치
1	0	0	0
2	30	20	50
3	122	40	162
4) / 50 / \$H\$7)	60	366

▲ 최대 경험치 산출 공식을 엑셀로 옮겨놓은 것

그러나 나눠지는 수는 나누는 수에 의해 일정한 수치로 나눠집니다. 나누는 수는 증가율 B의 값이 관건입니다. 증가율 B는 위 수식의 $(6 + \text{현재 레벨의 제곱값})$ 조합에 얼마를 나눠야 하는지를 결정하는 수치입니다. 이 증가율 B 값을 더 높은 값으로 나누면 나눌 수록 최종적으로 도출되는 나누는 수가 작아집니다. 반대로 증가율 B 수치가 낮으면 나누는 수가 증가하게 됩니다. 나누는 수가 증가한다면 최종 경험치의 값도 줄어들게 됩니다. 최종 경험치의 값이 높아지려면 나누는 수가 낮게 나와야 높은 최종 경험치 값을 얻을 수 있습니다.

나눠지는 수가 레벨이 높을 수록 값이 배로 증가하는 형태이므로, 증가율 B값에 의해 만들어지는 값은 레벨이 낮으면 수치가 너무 미미하므로 별 다른 영향을 줄 수 없지만 주인공의 레벨이 커지면 증가율 B값도 커지게 됩니다.

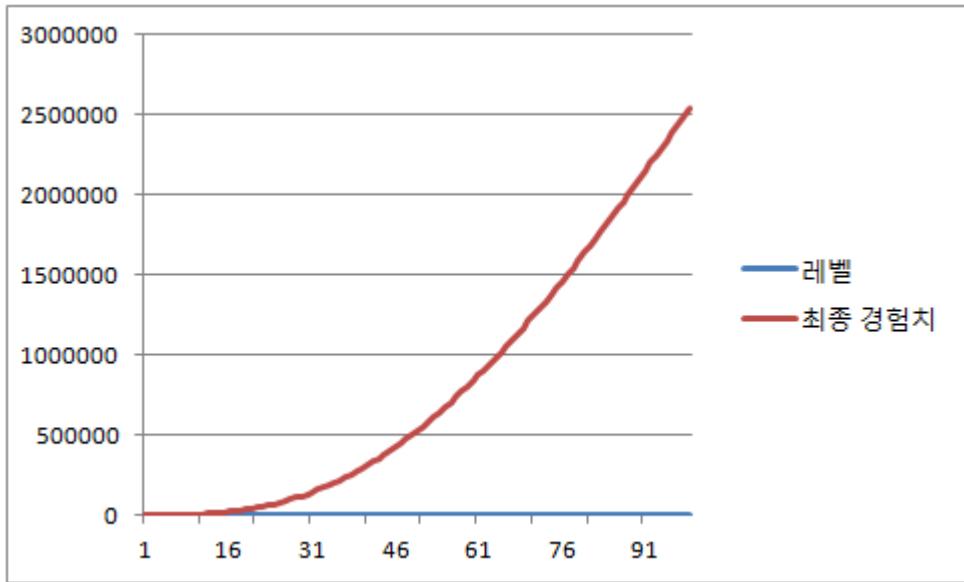
마지막에 더해주는 보정치는 단순히 현재 레벨에 1을 감산한 값에 보정치를 곱해준 값입니다.

따라서, 최종적으로 스크린샷과 같은 결과가 도출됩니다.

레벨	연산 결과	보정치	최종 경험치
1	0	0	0
2	30	20	50
3	122	40	162
4	306	60	366
5	615	80	695
6	1080	100	1180
7	1732	120	1852
8	2602	140	2742
9	3719	160	3879
10	5116	180	5296
11	6819	200	7019
12	8860	220	9080
13	11265	240	11505
14	14062	260	14322
15	17279	280	17559
16	20940	300	21240
17	25071	320	25391
18	29696	340	30036
19	34838	360	35198
20	40519	380	40899
21	46761	400	47161
22	53584	420	54004
23	61006	440	61446
24	69047	460	69507
25	77721	480	78201
26	87047	500	87547
27	97037	520	97557
28	107707	540	108247
29	119068	560	119628
30	131131	580	131711
31	143908	600	144508
32	157407	620	158027
33	171637	640	172277
34	186604	660	187264
35	202315	680	202995
36	218775	700	219475
37	235988	720	236708
38	253957	740	254697
39	272684	760	273444
40	292171	780	292951
41	312417	800	313217
42	333423	820	334243
43	355188	840	356028
44	377709	860	378569
45	400983	880	401863
46	425007	900	425907
47	449778	920	450698
48	475289	940	476229
49	501536	960	502496
50	528512	980	529492
51	556212	1000	557212
52	584627	1020	585647
53	613751	1040	614791
54	643576	1060	644636
55	674093	1080	675173
56	705293	1100	706393
57	737168	1120	738268
58	769708	1140	770848
59	802903	1160	804063
60	836743	1180	837923
61	871218	1200	872418
62	906319	1220	907539
63	942033	1240	943273
64	978351	1260	979611
65	1015261	1280	1016541
66	1052753	1300	1054053
67	1090815	1320	1092115
68	1129437	1340	1130777
69	116608	1360	1169968
70	1208315	1380	1209695
71	1248548	1400	1249948
72	1289296	1420	1290716
73	1330547	1440	1331987
74	137228	1460	1373751
75	1414516	1480	1415996
76	1457211	1500	1458711
77	1500365	1520	1501885
78	1543968	1540	1545508
79	1588008	1560	1589568
80	1632475	1580	1634055
81	1677358	1600	1678958
82	172264	1620	1724266
83	1768331	1640	1769971
84	1814400	1660	1816060
85	1860844	1680	1862524
86	1907654	1700	1909354
87	1954819	1720	1966539
88	2002330	1740	2004070
89	2050177	1760	2051937
90	2098351	1780	2100131
91	2146443	1800	2148643
92	2195643	1820	2197463
93	2244744	1840	2246584
94	2294136	1860	2295996
95	234810	1880	2345690
96	2395760	1900	2395660
97	2443975	1920	2445895
98	2494449	1940	2496389
99	2545173	1960	2547133

▲ 1 ~ 99 레벨의 최대 경험치와 보정치

위 경험치 데이터들은 도움말에 있는 공식을 활용하여 엑셀로 결과를 구해본 것입니다. 내친김에 엑셀로 그래프를 그려보면 아래와 같이 완만한 곡선을 보실 수 있습니다. 별 다른 어려움 없이 레벨업을 할 수 있다는 증표이기도 합니다.



▲ 경험치 곡선(빨간색)

사실 굳이 경험치 곡선까지 분석하면서 스스로 어려운 일을 만들 필요는 없습니다. 아직 알아야 할 것이 태산 같이 쌓여있어서 저처럼 강의를 해야 하는 강의자 입장이 아니라면 한가롭게 경험치 곡선이나 분석 할 시간은 없기 때문입니다.

2. 능력치 곡선(Parameter Curves)의 분석

경험치 곡선 다음으로 보실 항목은 능력치 곡선(Parameter Curves)입니다. 우리가 눈으로 볼 수 있는 체력이나 마력 게이지, 게임이 시작되기 전까진 그저 숫자로만 되어있는 수치 데이터에 불과합니다. 다른 툴에서는 수치 데이터를 직접 엑셀 등으로 위 경험치 곡선 공식 같은 것을 만들어서 직접 만들어내야 하지만 RPG 만들기 툴에선 놀랍게도 능력치 곡선이라는 편리한 기능을 제공하고 있습니다.

사실 이 능력치 곡선은 데이터베이스 작업 탭에서 가장 눈에 크게 들어오는 부분일 것입니다. 경험치 곡선과는 다르게 칙칙하지도 않고 알록달록한 색상으로 장식되어있기 때문입니다. 또한 그래프가 너무 직관적이라 이 기능을 모르는 사람도 몇 번 눌러보면 형태에 맞는 능력치를 짤 수 있습니다.

General Settings

Name:	Soldier
EXP Curve:	[30, 20, 30, 30] ...

Parameter Curves

MHP	MMP	ATK	DEF
MAT	MDF	AGI	LUK

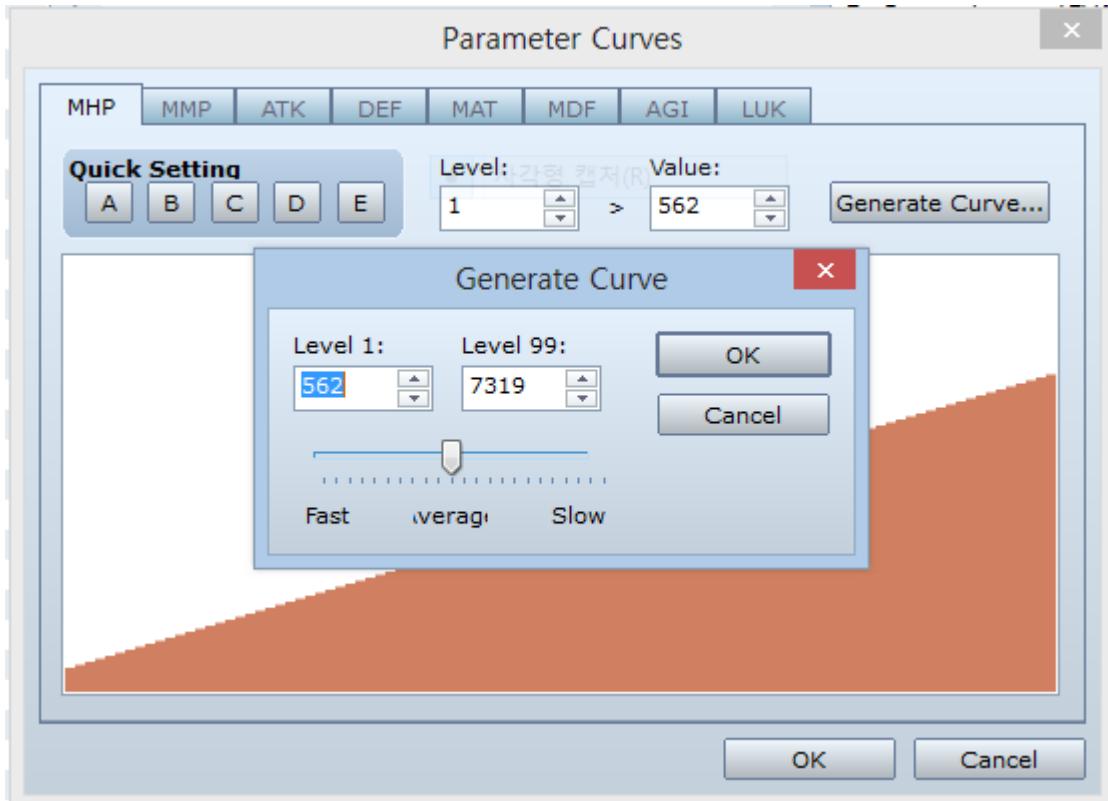
Skills

Level	Skill	Note
Lv 5	Strong Attack	
Lv 10	Cleave	
Lv 15	Berserker's Roar	
Lv 20	Berserker's Dance	
Lv 25	Giant's Rampage	

Features

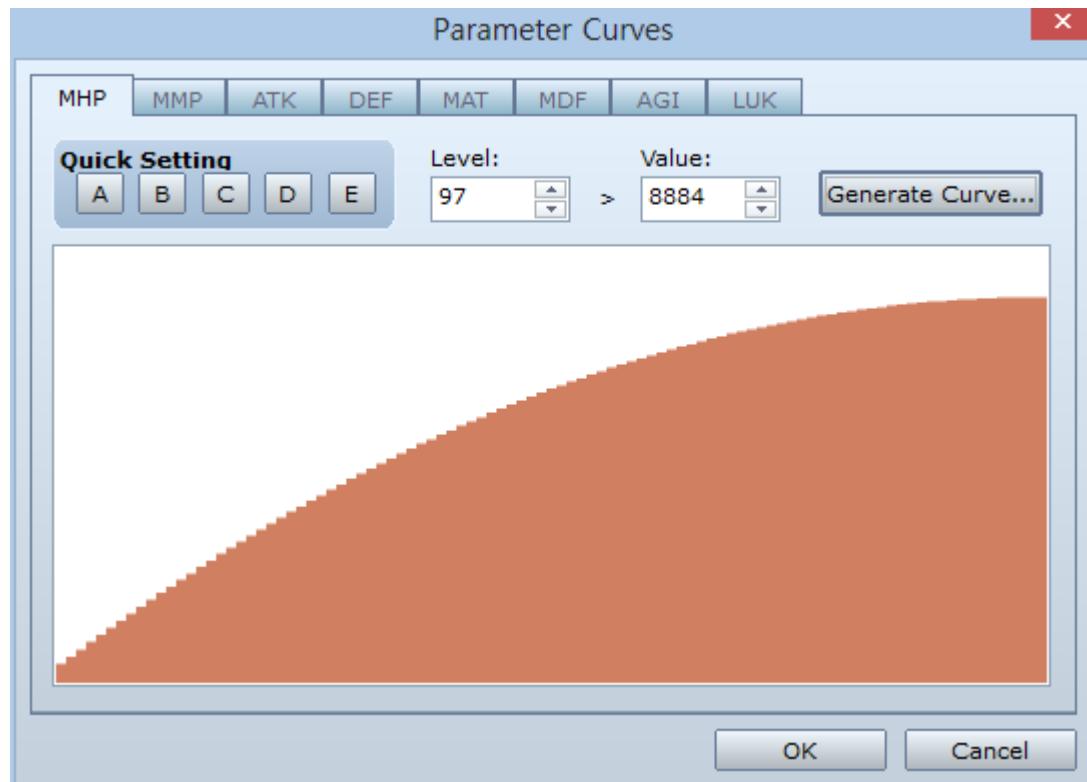
Kind	Content
Sp-Parameter	[TGR] * 100%
Ex-Parameter	[HIT] + 0%
Ex-Parameter	[EVA] + 5%
Ex-Parameter	[CRI] + 4%
Add Skill Type	[Special]
Equip Weapon	[Axe]
Equip Armor	[General Armour]
Equip Armor	[Light Armour]
Equip Armor	[Heavy Armour]
Equip Armor	[Small Shield]
Equip Armor	[Large Shield]

그래프 총 8개가 놓여져있는데 그 이유는 메인 능력치가 8개이기 때문입니다. 메인 능력치는 게임을 많이 해본 사람이라면 누구나 알 수 있는 명칭으로 되어있습니다. 잘 알고 계시듯 MHP¹는 최대 체력입니다. MMP²는 최대 마력을 뜻합니다. ATK³는 공격력이며 DEF⁴는 방어력입니다, MAT⁵는 마법 공격력의 약자고, MDF⁶는 마법 방어력의 약자입니다. 그리고 AGI⁷는 민첩력 수치를 나타냅니다. 마지막으로 LUK⁸는 행운 수치입니다. 이 수치들은 모두 전투 시스템과 깊은 관련이 있습니다. 또한 오른쪽에 Features라는 부분을 보시면 알겠지만 전투 시스템과 메인 능력치 같은 데이터와 연관되어있는 부가 능력치들도 있습니다.



▲ Generate Curve

능력치 곡선을 더블 클릭하고 들어오면 위와 같은 창이 뜨게 됩니다. 직관적으로 알 수 있는 그래프가 보이는데 마우스 왼쪽 버튼으로 그래프를 클릭해서 수치를 조절할 수 있게 되어있어서 매우 편리합니다. 또한 마우스 오른쪽 버튼을 누르면 값도 말풍선으로 보여줍니다. 또한 클릭하는 것 조차 귀찮은 분들을 위해 빠른 설정(Quick Setting)과 능력치 곡선 생성(Generate Curve)과 같은 기능도 제공되고 있습니다. 그래프를 해석하는 일도 꽤 쉬운 편인데 Fast 는 말그대로 빠른 성장을 나타냅니다, 빠른 성장을 위해 데이터가 짜여지는 것입니다. 중간에 있는 글자는 약간 짤린 상태인데 Average 라고 해서 평균값을 나타내는 말입니다. 또한 Slow 는 느린 성장을 나타냅니다.



▲ MHP의 빠른 성장 곡선

빠른 성장의 경우, 단 기간에 높은 능력치까지 도달하게 되는 것을 알 수 있고, 느린 성장의 경우 높은 능력치까지 도달하기 위해 많은 레벨업이 필요함을 그래프를 통해 알 수 있습니다. 상단에 8개의 탭이 있는 것도 보이는데 한 번 창을 띄우면 다른 능력치까지 조절할 수 있다는 것을 본능적으로 알 수 있습니다. 사실 능력치는 아래와 같은 공식을 채워서 만들어집니다.

```
@params = Table.new(8,100)
(1..99).each do |i|
  @params[0,i] = 400+i*50
  @params[1,i] = 80+i*10
  (2..5).each { |j| @params[j,i] = 15+i*5/4 }
  (6..7).each { |j| @params[j,i] = 30+i*5/2 }
end
```

▲ 메인 능력치의 할당 공식

이런 스크립트 코드 구문이 도움말에 나와있긴 하지만 초보자가 보기엔 인간이 알아볼 수 있는 수준이 아닌 것처럼 보일 수 있으므로 간단하게 인간이 알아볼 수 있는 수식으로 만들어보면 아래와 같습니다. 수식의 형태는 기본값에 레벨에 맞는 보정치를 더한 값으로 되어있습니다.

α = 레벨

$$MHP = 400 + (\alpha \times 50)$$

$$MMP = 80 + (\alpha \times 10)$$

$$ATK = 15 + (\alpha \times 5 \div 4)$$

$$DEF = 15 + (\alpha \times 5 \div 4)$$

$$MAT = 15 + (\alpha \times 5 \div 4)$$

$$MDF = 15 + (\alpha \times 5 \div 4)$$

$$AGI = 30 + (\alpha \times 5 \div 2)$$

$$LUK = 30 + (\alpha \times 5 \div 2)$$

▲ 수식으로 정리된 공식

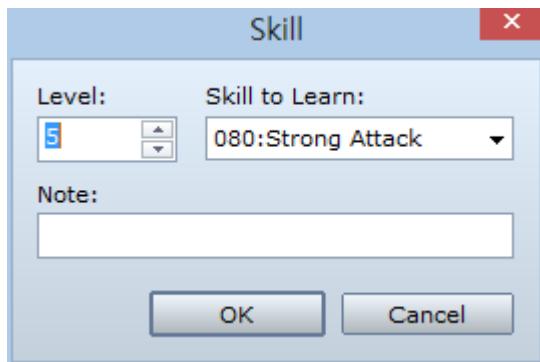
하지만 능력치 곡선의 빠른 성장과 느린 성장과 같은 곡선 형태에 따라 기본값과 보정치값이 달라지고 개발사에서 곡선의 형태에 따라 위 공식에 어떤 수치를 넣는지 자세하게 알 수 없기 때문에 당장은 위 공식을 중요하게 생각할 필요는 없으며 무엇보다 데이터베이스에서 손쉽게 만들어낼 수 있는 데이터들이기 때문에 신경을 쓸 필요가 전혀 없습니다.

데이터베이스로 설정할 수 있는 능력치의 한계는 MHP의 경우 1부터 9,999 까지입니다. (적군은 999,999 까지) MMP의 경우엔 9,999이며 나머지 능력치들은 999가 한계입니다. "이 한계를 돌파할 수 있나요?" 질문하고 싶으신 분들도 있을 겁니다. 물론 스크립트를 이용하여 레벨이나 능력치를 마음대로 확장할 수 있습니다. 그런 분들은 능력치 데이터를 더 만들기 위해 위 수식들을 참고를 해야 할 것입니다. 데이터베이스가 만들어낼 수 있는 데이터 범위를 넘어선 상태이기 때문입니다.

그런데 문제는 데이터 범위의 한계입니다. RPG Maker VX Ace에서는 능력치를 저장하기 위해 테이블(Table)이라는 곳에 데이터를 저장합니다. 이 테이블이라는 구조는 데이터의 범위가 -32,767부터 32,767까지가 한계로 정해져있습니다. 이 한계와 능력치가 인플레이션되는 현상을 고려하여 제작해야 하기 때문에 능력치를 무작정 높이기 보다는 개발 단계에서 능력치를 작은 값으로 압축해서 표현하는 것이 더 좋은 습관이 될 것입니다.

3. 스킬(Skills)

Classes 탭에서의 스킬 설정 부분은 눈으로 보면 누구나 알 수 있겠지만 해당 레벨이 되면 스킬을 자동 익힐 수 있게 설정하는 기능입니다. 스킬을 자동으로 익혀야 하는 게임에선 직업 탭에서 스킬을 설정하면 해당 레벨이 되면 자동으로 스킬을 습득하므로 매우 편리할 것입니다. 수동으로 익혀야 한다면 이벤트에서 Change Skills라는 기능을 제공하고 있으므로 그 기능을 이용하면 해결될 것입니다.

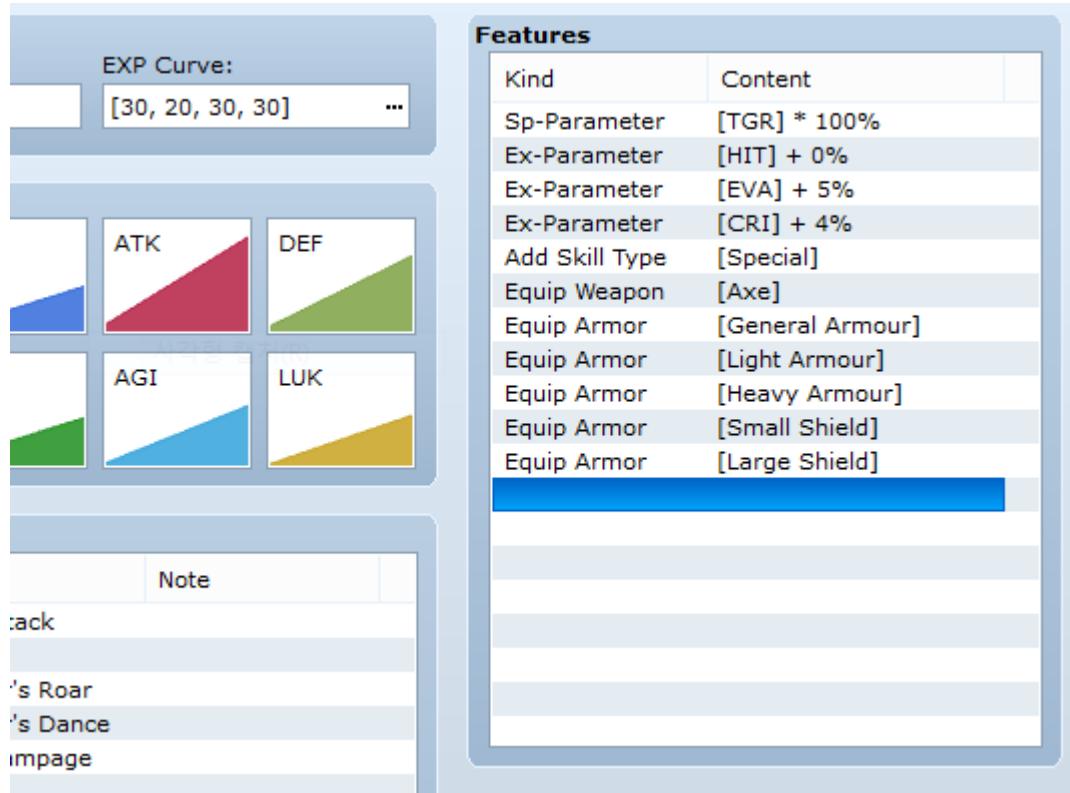


▲ Skill 창의 구조

내부적으로 Skill 창은 이런 구조로 되어있는데, 레벨 값은 1부터 99까지만 설정할 수 있고, 오른쪽은 배울 스킬을 선택할 수 있는 에디터 박스가 위치합니다. 하단에 Note라는 부분은 메모라고 해석할 수 있는 부분인데 주로 스크립트에서 기능을 확장하기 위해 사용합니다. 이 부분에 있는 내용도 변수에 저장되는 데이터이기 때문에 스크립트 에디터에서 읽을 수 있습니다. 그래서 일부 스크립트에선 정규 표현식이라거나 사용해서 숫자값이나 문자열 데이터를 뽑아갑니다. 그래서 해당 스킬에 대한 특별한 설정이나 확장 기능을 사용 할 수 있게 도와줍니다. 이 부분의 구체적 사용 방법은 스크립트를 모른다면 외계어로 보일 수 있는 부분이기 때문에 생략하고 넘어가겠습니다.

4. 특성(Features)

RPG를 하다보면 능력치가 항상 시시각각 바뀌는 상황에 있는 것을 알 수 있습니다. 파티 사냥 중에 다른 파티원이 버프를 걸어줘서 버프가 걸려서 모든 능력치 값이 상승해야 되는 경우도 있고 어그로를 끌다가 디버프 상태에 빠지게 되면서 모든 능력치 값이 저하돼 팀에 악영향을 끼치는 경우도 있습니다. 또한 던전에서 보스를 공략한 후에 보상으로 아이템을 얻을 수 있는데 툴팁Tooltip 정보에 나와있는 능력치 값들을 보니 같은 아이템이지만 어떤 아이템은 적중률을 높이고 어떤 아이템은 회피율을 높이고 두 가지 종류로 나뉘어 있기도 합니다. 그리고 얼음 동굴에 있는데 화염으로 공격을 하려고 하거나, 속성에 따라 공격력이 달라져야 할 경우도 있습니다. 게다가 도끼를 착용할 수 없는 직업도 있을 수 있고, 착용할 수 있는 직업군도 있을 겁니다. 그런 경우를 위해 존재하는 기능이 특성(Features) 기능입니다.



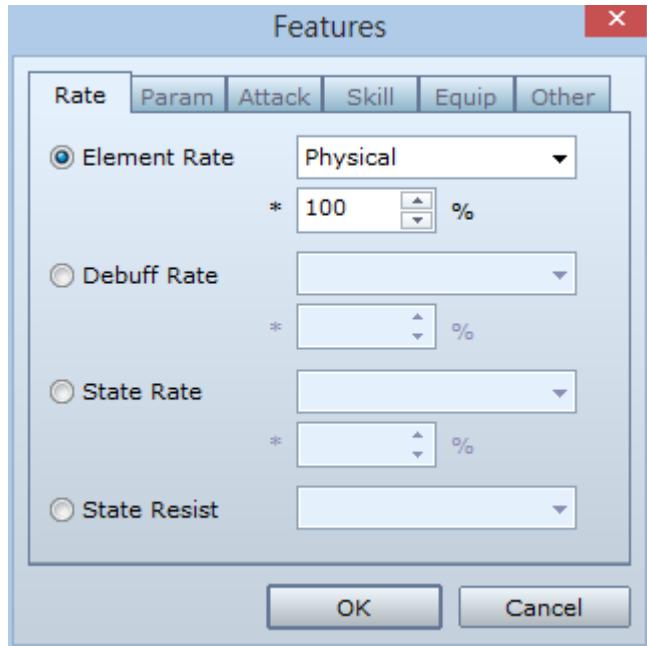
▲ 이미 많은 것이 미리 설정되어있는 것을 알 수 있습니다

이렇게 상황 별로 시시각각 바뀌는 능력치와 상황을 만들기 위해 특성(Features)이라는 기능이 제공됩니다. 또한 메인 능력치 이외의 RPG 만들기 툴에 존재하는 명중률, 회피율 같은 부가 능력치들을 이 특성 창에서 설정할 수 있습니다. 위 스크린샷에 많은 것이 미리 설정되어있는데 위에서 보는 것과 같이 실제로 많은 기능이 있어서 할 이야기가 많습니다.

특성(Features)이라는 기능은 직업(Clases) 탭 뿐만 아니라 액터(Actors)탭, 아이템(Item) 탭, 무기(Weapon)탭, 방어구(Armor)탭, 적(Enemies)탭 그리고 속성(States) 탭에도 존재합니다. 이렇게 능력치를 조작해야 하는 모든 탭에 존재합니다. 이렇게 능력치 변화에 직접적인 영향을 주는 기능이 바로 특성입니다. 무기나 방어구 탭의 부속 기능 정도로 생각하실 수도 있겠지만, 특성도 주인공이나 적 개인이 가지고 있는 일종의 능력치 변수 중 하나이며 최종적으로 누적, 합산된 결과가 전투에 반영됩니다.

4.1. Rate - 속성 유효도(Element Rate)

목록을 더블 클릭하거나 엔터를 누르면 아래와 같은 창이 표시됩니다. 기본적으로 6개의 탭이 존재 합니다. Rate, Param, Attack, Skill, Equip, Other 순입니다. 첫번째 탭은 Rate 라고 되어있는데 일본어판에서는 내성이라고 부르는 부분입니다.



Element Rate 는 속성 유효도입니다. 속성 유효도는 최종 데미지 계산에 사용되는 비율입니다. 데이터베이스에서 무기 (Weapon)와 적(Enemies)을 만들 때 공격 속성이라는 것을 추가로 지정해줘야 최종 데미지에서 속성 유효도라는 것이 적용됩니다.

적(Enemies) 탭에서 Element Rate 옵션 지정을 통해 플레이어의 공격 속성에 따라 데미지를 변동시킬 수 있습니다. 예를 들면, 특정 서식지나 던전에서 주인공이 가지고 있는 신성 속성의 무기가 큰 위력을 발휘하도록 설정할 수도 있습니다. 마왕성 같은 어둠이 깔려있는 우중충한 곳에 살고 있는 마물을 상대해야 한다면 상식적으로 신성 계열의 무기와 마법이 더 큰 위력을 발휘해야 한다는 것을 본능적으로 알 수 있습니다. 그럴 땐 속성 유효도 특성을 활용해줘야 합니다.

속성 유효도는 아래와 같이 Enemies 탭에서 활용됩니다.

General Settings

Name:	MHP:	MMP:
Zombie	1050	0
Graphic:	ATK:	DEF:
	75	40
MAT:	MDF:	
AGI:	LUK:	
50	50	

Rewards

EXP:	180
Gold:	320

Drop Items

(None)	...
(None)	...
(None)	...

Features

Kind	Content
Ex-Parameter	[HIT] + 95%
Ex-Parameter	[EVA] + 5%
Atk Element	[Physical]
Element Rate	[Absorb] * 0%
Element Rate	[Fire] * 150%
Element Rate	[Holy] * 200%

위 스크린샷에 강조된 부분을 보면, Element Rate [Holy] * 200% 이라고 되어있는 부분을 보실 수 있습니다. 속성 유효도가 생략되었거나 100 %로 지정되어있는 경우엔 기본 데미지가 들어갑니다. 그런데 만일 여러분들이 위 좀비를 신성 (Holy) 계열의 무기나 마법으로 공격을 가한다면 좀비에겐 평소보다 데미지를 2배 이상 줄 수 있을 것입니다.

예를 들면, Startlight라는 신성 계열의 마법이 있습니다. 이 마법은 실질 데미지가 400이며 좀비에게 큰 데미지를 줄 수 있습니다. 이 마법으로 좀비를 공격하게 될 땐 기본 데미지에 속성 유효도가 곱해져서 아래와 같이 데미지가 변동하게 됩니다. 사실 시전자의 마법 공격력과 적의 마법 방어력도 빼줘야 하지만 계산을 편하게 하기 위해 이해를 돋기 위해 아군의 마법 공격력과 적군의 마법 방어력이 0이라고 가정하면 결과는 아래와 같습니다.

$$\text{최종 데미지} = \text{Starlight의 데미지}(400) * [\text{속성 유효도 } 200\%]$$

$$\text{최종 데미지} = \text{Starlight의 데미지}(400) * (200\% / 100.0)$$

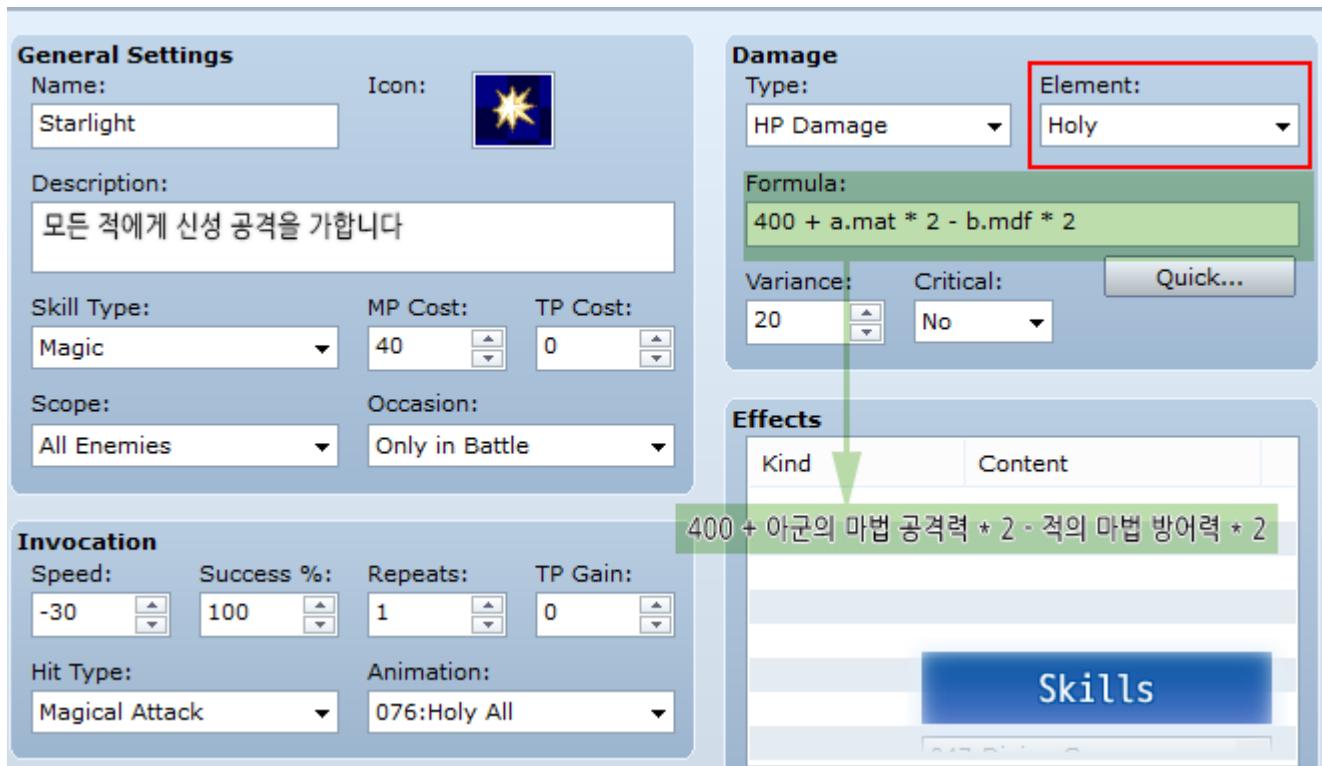
$$\text{최종 데미지} = 400 * 2.0$$

$$\text{최종 데미지} = 800$$

[속성 유효도 200%]는 $(200\% / 100.0)$ 과 같습니다. 200에서 100을 나누면 2가 됩니다. 따라서 최종 결과는 800임을 알 수 있습니다. 마법 데미지가 Holy 속성인 경우에만 속성 유효도에 의해 데미지가 위와 같이 2배로 변동되는 것입니다. 참고로 속성 유효도는 최대 1000%까지 설정할 수 있게 되어있습니다. 1000%는 평소 데미지 수치보다 10배 더

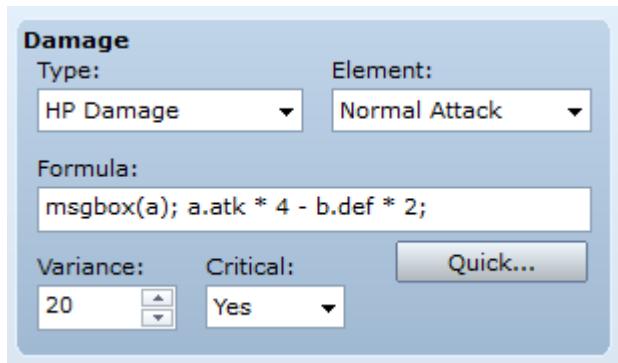
큰 데미지를 줄 수 있는 수치입니다. 10배라는 값은 10.0이라는 실수로 반환되어 데미지에 곱해집니다. 곱해지게 되면 10배 이상의 데미지를 줄 수 있게 되는 것입니다.

속성은 보통 무기 설정에서 속성 타입을 지정하거나 스킬의 데미지 부분에서도 속성을 설정할 수 있습니다. 마법 데미지는 Formula라는 수식을 통해 결정되는데 내친김에 이것까지 곁들어 설명하겠습니다.



▲ 데미지(Damage) 부분에 Holy 속성이 지정되어있는 것을 볼 수 있습니다

Formula는 수식이라는 뜻입니다. 전투 스킬에 지정되어있는 위 데미지 공식에 의해서 최초로 데미지라는게 생성됩니다. 여기서 a는 아군이며 b는 적 대상입니다. 최초 데미지가 이 수식에 의해 생성됩니다. 주의해야 할 점은 이 수식들은 실제 루비 스크립트의 코드로 인식되는 문자열들 9이라는 것입니다. 그래서 문법 검사도 실시됩니다. 실제 루비 스크립트의 문자열로 인식되기 때문에 실제 변수값 10도 가져올 수 있고 메시지박스로 메시지 11까지 띄울 수도 있습니다.



▲ Formula 박스는 실제 루비 스크립트를 적는 공간입니다

Formula 박스는 한 칸 밖에 없습니다. 그래서 긴 스크립트를 작성할 순 없는데 프로그래밍 언어에서 많이 보던 세미 콜론 `;` 12 으로 라인을 구분할 수 있습니다. 딱히 그렇게 까지 할 필요는 없지만, 이 좁쌀만한 공간을 활용해서 뭔가를 만들고 싶다면 필요할 수도 있을 것 같습니다. 그런 작업을 할 땐 루비 스크립트에서 마지막 라인을 반환한다는 사실을 기억하시기 바랍니다. 따라서 마지막에 수식만 반환하게 설정해주면 되는 것입니다. `return 13` 같은 키워드 필요없이 그냥 위 스크린샷과 같이 마지막에 수식을 적어주면 됩니다.

코드	뜻
<code>x.atk</code>	공격력
<code>x.def</code>	방어력
<code>x.mat</code>	마법 공격력
<code>x.mdf</code>	마법 방어력
<code>x.agi</code>	민첩력
<code>x.luk</code>	행운
<code>x.mhp</code>	최대 HP
<code>x.mmp</code>	최대 MP
<code>x.hp</code>	현재 HP
<code>x.mp</code>	현재 MP
<code>x.tp</code>	현재 TP
<code>x.level</code>	레벨

▲ 능력치 코드

Damage 부분에서 Variance는 수치 분산도를 나타냅니다. 기본값이 20인데 그냥 20의 수치 분산도를 적용하게 되면 데미지가 80% ~ 120% 사이로 분산됩니다. 분산된다라는 것은 데미지가 매 턴마다 항상 같지 않음을 나타냅니다. 항상 같은 100% 데미지로 공격을 가하는게 아니라 난수값이 적용돼 매 턴마다 달라집니다. 따라서 매 턴마다 데미지가 항상 달라지게 됩니다. 평소보다 적은 80%에 가까운 데미지를 입힐 수도 있고, 운이 좋으면 120%에 가까운 높은 데미지를 입힐 수도 있게 됩니다. 아래는 스크립트에 정의된 내용입니다. 인간이 알아볼 수 있는 수준으로 꾸며본 것입니다.

```
# * Applying Variance (수치 분산도)
#
def apply_variance(damage, variance)
    amp = [damage * variance / 100, 0].max.to_i 추가데미지는400
    var = rand(0..400 + 1) + rand(0..400 + 1) - 400 추가데미지=-400~400사이의수(난수)
    damage >= 0 ? damage + var : damage - var 최종데미지=800+추가데미지
end
#
```

▲ 수치 분산도의 스크립트 정의

수치 분산도는 사용자의 입맛에 맞게 0% 부터 100%까지 적용할 수 있습니다. 100% 를 적용하게 되면 최종 데미지는 0% ~ 200% 사이로 결정되는 것입니다. 아예 데미지가 발생하지 않을 수도 있는 것입니다. 수치 분산도가 적용되는 시기는 데미지 공식을 확인해야 알 수 있는데 데미지 공식은 아래와 같습니다. 이 공식도 인간이 알아볼 수 있는 수준으로 오른쪽에 주석을 달아봤습니다. 주석 이외의 코드는 굳이 해석 할 필요는 없습니다.

```
#-----  
# * Calculate Damage(데미지 공식)  
#-----  
def make_damage_value(user, item)  
    value = item.damage.eval(user, $bf, $game_variables) #Formular 의해 최초 데미지가 만들어집니다.  
    value *= item_element_rate(user, item) #속성 유효도 값이 데미지에 곱해집니다  
    value *= pdr if item.physical?  
    value *= mdr if item.magical? #데미지 종류에 맞는 비율을 적용합니다(물리 데미지, 마법 데미지, 회복률)  
    value *= rec if item.damage.recover?  
    value = apply_critical(value) if @result.critical #크리티컬 데미지라면, 3배의 데미지를 입힙니다  
    value = apply_variance(value, item.damage.variance) #수치 분산도를 적용합니다  
    value = apply_guard(value) #방어 시엔 데미지에 방어 효과율이 적용돼 데미지가 감소합니다  
    @result.make_damage(value.to_i, item)  
end
```

▲ 데미지 공식의 스크립트 정의

일반적인 데미지 공식은 위와 같은데, 사실 이렇게 간단하진 않고 도와주는 메소드들이 잔뜩 있습니다. 그 메소드들까지 소개하기에는 인내심에 한계가 있기도 하고 강의를 읽다가 지쳐버릴 수도 있기 때문에 생략하겠습니다.

1. Max HP
2. Max MP
3. Attack power
4. Defense power
5. Magic attack
6. Magic defense
7. Agility
8. Luck

9. '루비' 같은 인터프리터 언어들은 해석기가 따로 있으며 컴파일이 필요 없습니다. 프로그램이 실행되는 런타임 시간에도 해석 명령을 내릴 수 있습니다. Kernel 모듈의 eval(string) 메소드를 통해 해석기가 문자열을 해석하기 시작합니다.
10. \$game_variables[변수번호]
11. RGSS3 에선 msgbox(string)라는 메소드를 통해 메시지 박스창을 띠울 수 있습니다.
12. 프로그래밍 언어에서 주로 코드의 끝을 표시하기 위해 사용하는 문자입니다. 루비에선 필요없지만 Formula 박스처럼 한 줄에 다 적어야 할 때, 줄바꿈의 의미로 사용할 수 있습니다.
13. 프로그래밍 언어에서 주로 반환의 의미로 사용합니다. 루비에선 마지막에 실행된 코드가 반환됩니다.

[10강] 기본 전투의 처리 3편 (심화편) - RPG Maker VX Ace 강좌

시작하기

2015/08/05 21:51

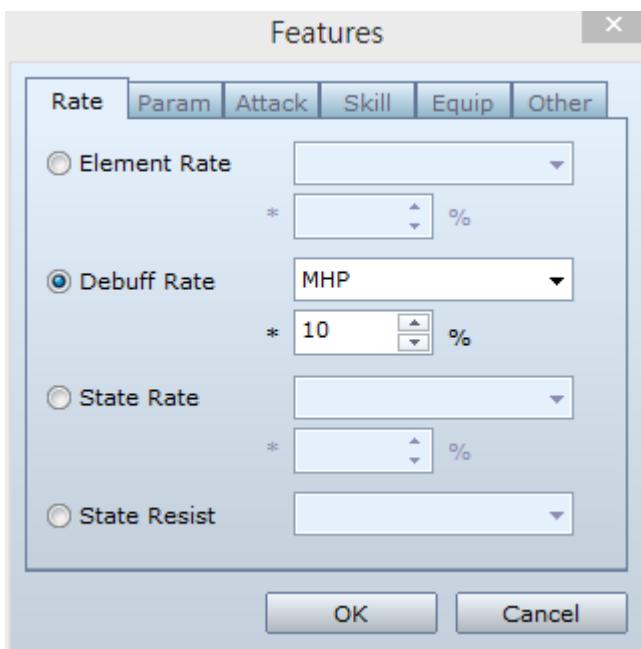
<http://blog.naver.com/biud436/220442229649>



"기본 전투의 처리 3편 (심화편) - RPG Maker VX Ace"

4.2. Rate - Debuff Rate

디버프 효과는 디버프 효과가 설정된 아이템이나 스킬을 사용할 때 일정 확률로 발동됩니다. Debuff Rate 값은 그 일정 확률에 관련된 값입니다. 이 값을 변경하면 디버프 효과의 성공 확률에 영향을 줄 수 있습니다.



1000% 에 가까울 수록 디버프에 성공할 가능성이 높아집니다. 행운과 관련이 있는 luk_effect_rate라는 비율도 디버프 확률에 반영되는데 이 비율값은 적군과 아군의 행운 수치에 따라 바뀌게 되며 아래와 같이 계산됩니다. 즉,

`luk_effect_rate` 는 현재 파티원의 행운 수치에서 적군의 행운 수치를 뺀 값을 일정 비율(0.001 ~ 0.999)로 만든 값이라는 것을 알 수 있습니다.

```
#-----#
# * Get Effect Change Rate by Luck
#
def luk_effect_rate(user)
  [1.0 + (user.luk - luk) * 0.001, 0.0].max
end
```

Debuff Rate 는 스크립트 내부에서는 실질적으로 0.0 ~ 10.0 로 환산되고 이 값이 행운 비율과 곱해지게 되면 `rand` 라는 난수와 값을 비교해 디버프 효과 적용 여부를 판단하게 됩니다. 따라서 디버프 효과가 적용되려면 아군의 행운 수치가 적군보다 높아야 하고 Debuff Rate 가 1000% 에 가까워야 한다는 것을 최종적으로 알 수 있습니다.

```
#-----#
# * [Debuff] Effect
#
def item_effect_add_debuff(user, item, effect)
  chance = debuff_rate(effect.data_id) * luk_effect_rate(user)
  if rand < chance
    add_debuff(effect.data_id, effect.value1)
    @result.success = true
  end
end
```

버프나 디버프 효과가 발동되면 아래 공식에 의해 능력치가 바뀌게 됩니다. 이때 능력치는 아래 공식에 의해 50% 이상 150% 사이의 값으로 변동하게 됩니다. 버프나 디버프는 최대 2번, 중첩 사용이 가능하기 때문에 공식은 아래와 같습니다.

[디버프(-2 또는 -1) 또는 버프(1 또는 2)] * 0.25 + 1 # 0.5 ~ 1.5 (50% ~ 150%)

예를 들어 스킬 목록에 있는 Berserker's Roar(버서커의 포효)란 스킬을 사용하면 5턴 동안 파티원의 공격력이 증가하게 됩니다. 이때 버프 효과가 한 번 적용되었으므로 공격력은 5턴 간 기존 공격력보다 25% 만큼 증가하게 됩니다.

Skills

- 076:Fire Weapon
- 077:Ice Weapon
- 078:Thunder Weapon
- 079:-----
- 080:Strong Attack
- 081:Cleave
- 082:Berserker's Roar**
- 083:Berserker's Dance
- 084:Giant's Rampage
- 085:Tackle
- 086:Chakra
- 087:Tiger Stance
- 088:No Shadow Kick
- 089:Claw Dance
- 090:Cover
- 091:Provoke
- 092:Super Guard
- 093:Restrengthen
- 094:Zero Storm
- 095:Aura Blade
- 096:Dispel
- 097:Magic Barrier
- 098:Reinforce
- 099:Radiant Blade

General Settings

Name: Berserker's Roar Icon: 

Description: For 5 turns, increase ATK of party.

Skill Type: Special MP Cost: 0 TP Cost: 25

Scope: All Allies Occasion: Only in Battle

Invocation

Speed: 0 Success %: 100 Repeats: 1 TP Gain: 0

Hit Type: Certain Hit Animation: 034:Shout

Damage

Type: None Element:

Formula:

Variance: Critical: Quick...

Effects

Kind	Content
Add Buff	[ATK] 5 turns

Nightmare (나이트메어)라는 스킬을 쓰면 적에게 디버프를 걸 수 있게 되는데 그렇게 되면 적의 마법 방어력이 대폭 감소하게 됩니다. 이때 디버프가 두 번 걸리게 되므로 적군의 마법 방어력(MDF)은 반으로 줄어들게 됩니다.

General Settings

Name: Nightmare Icon: 

Description: Puts all enemies to sleep and greatly lower their MDF.

Skill Type: Special MP Cost: 0 TP Cost: 50

Scope: All Enemies Occasion: Only in Battle

Invocation

Speed: 0 Success %: 100 Repeats: 1 TP Gain: 0

Hit Type: Certain Hit Animation: 056:Death

Damage

Type: None Element:

Formula:

Variance: Critical: Quick...

Effects

Kind	Content
Add State	[Sleep] 100%
Add Debuff	[MDF] 5 turns
Add Debuff	[MDF] 5 turns

4.3. Rate - State Rate (상태 유효도)

아이템이나 스킬의 사용을 통해 적이나 아군의 상태를 바꿀 때 영향을 주는 상태 이상 관련 비율값입니다. State Rate 가 높으면 적에게 상태 이상 마법을 성공적으로 걸 수 있는 확률이 증가합니다. 이 값도 Debuff Rate와 마찬가지로 0% 이상

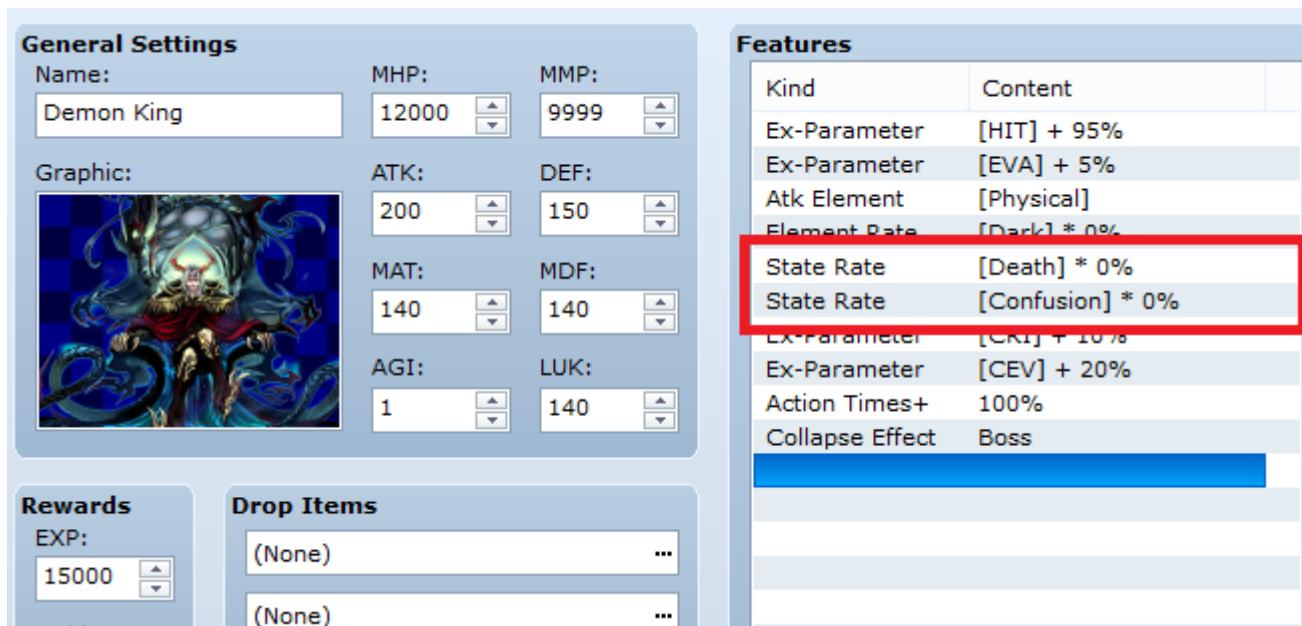
1000% 사의의 값을 지정할 수 있습니다. 아이템이나 스킬의 사용 대상이 절대적 관계일 경우 아이템이나 스킬의 성공 확률이 100%에 가까워야 하고, 적보다 행운 수치도 높아야 하고, State Rate도 1000%에 가까워야 성공 확률이 더 높아집니다. 하지만 사용 대상이 '절대적 관계'에 있지 않으면 이 확률과 관계 없이 아이템이나 스킬의 성공 확률만을 가지고 판단합니다.

```

#-----#
# * [Add State] Effect: Normal Attack
#
def item_effect_add_state_attack(user, item, effect)
    user.atk_states.each do |state_id|
        chance = effect.value1
        chance *= state_rate(state_id)
        chance *= user.atk_states_rate(state_id)
        chance *= luk_effect_rate(user)
        if rand < chance
            add_state(state_id)
            @result.success = true
        end
    end
end
#-----#
# * [Add State] Effect: Normal
#
def item_effect_add_state_normal(user, item, effect)
    chance = effect.value1
    chance *= state_rate(effect.data_id) if opposite?(user)
    chance *= luk_effect_rate(user)      if opposite?(user)
    if rand < chance
        add_state(effect.data_id)
        @result.success = true
    end
end

```

▲ Game_Battler 스크립트에서 상태 추가와 관련된 부분



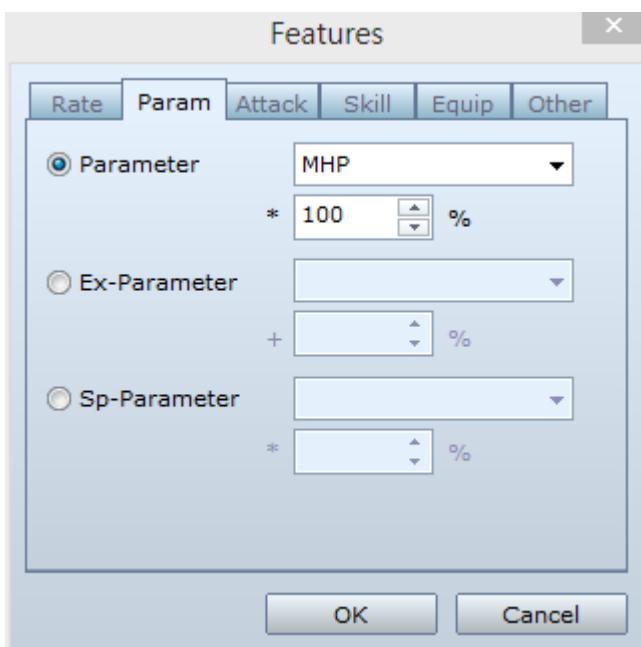
▲ 'Demon King'은 'Death' 와 'Confusion' 상태에 빠지지 않습니다.

4.4. Rate - State Resist (상태 저항)

특정 상태에 대해 저항력을 가질 수 있는 옵션입니다. 이 옵션을 가지고 있을 경우 특정 상태에 면역 상태가 됩니다. 상태 이상 마법에 당해도 상태 저항인지 체크하는 코드에 의해 상태가 바로 해제됩니다.

4.5. Param - Parameter (능력치)

데이터베이스에서는 MHP, MMP, ATK, DEF, MAT, MDF, AGI, LUK 등 8개의 능력치를 레벨 별로 결정할 수 있습니다. 가장 처음에 데이터베이스를 통해 정해놓은 능력치 값이 100%라고 하고 기준이라고 본다면 기준으로부터 비율값을 높이면 능력치가 상승하고 줄이면 능력치가 데이터베이스에서 주어진 것보다 줄어들게 됩니다.



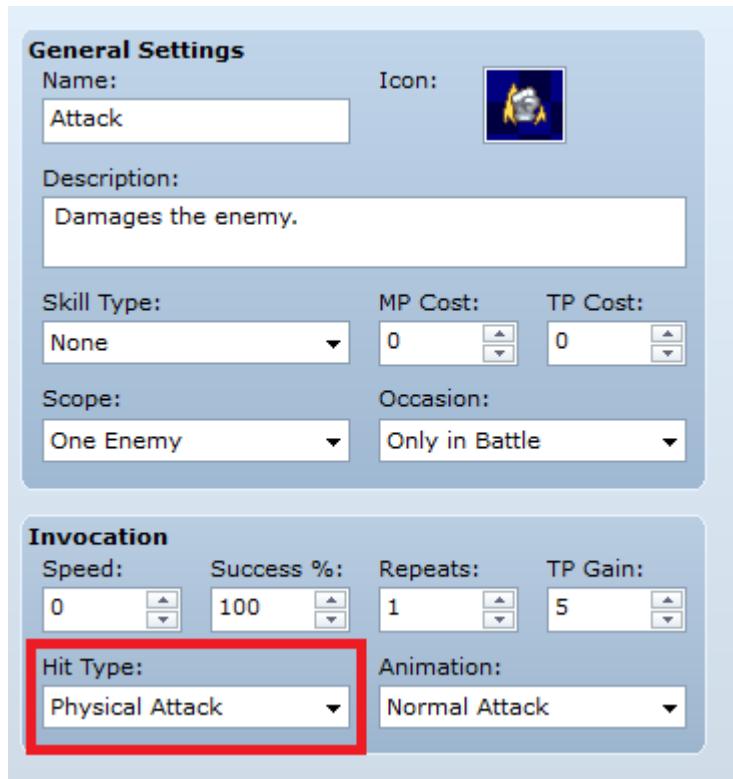
비율값은 0% ~ 1000% 까지 조정할 수 있습니다. 기준 능력치를 기준으로 곱해지는데 100% 가 실질적으로 1.0 으로 처리되므로 1000% 면 수치적으로 기준 능력치의 10배가 됩니다.

4.6. Param - Ex Parameter (부가능력치)

부가 능력치(Ex-Parameter)는 전투 진행에 있어 지대한 영향을 끼치는 능력치들이며 기본적으로 0%로 설정되므로 새로운 직업을 만들거나 적을 설정할 때 반드시 특징 탭에서 퍼센트 값을 설정해줘야 합니다.

Game_BattlerBase의 xparam 메소드는 Ex-Parameter로 분류되어있는 값들을 무기나 방어구, 상태, 적 탭에 설정되어 있는 모든 부가 능력치 설정값들은 모두 수집하여 최종적인 누적된 수치를 만들어냅니다. 이렇게 누적된 데이터들은 턴제 전투 시스템에서 어떤 전투 요소의 성공 여부를 판단할 때 판단값으로 요긴하게 사용됩니다.

HIT rate (적중률)과 EVAsion rate (회피율), Magic EVasion rate (마법 회피율)은 스킬이나 아이템을 사용할 때 히트 판정에 영향을 주는 비율입니다. 스킬이나 아이템을 사용하려면 먼저 히트 판정이 나야 하는데 히트 판정은 적중률과 회피율 값을 검사해서 판정합니다. 적중률이 높아 적중에 성공해도 상대방의 회피율이 높으면 히트 판정이 나질 않게 되는 것 이죠. 즉, 회피율이 높으면 스킬이나 아이템 사용에 실패할 확률이 높아집니다. (이 부분은 Game_Battler 클래스의 item_apply 메소드에 정의되어있습니다)



그리고 히트 타입(Hit Type)이 '물리 공격'이라면 EVAsion rate (회피율)이 적용되고 마법 공격이라면 'Magic EVasion rate (마법 회피율)'이 적용됩니다. 적중률과 회피율은 난수적인 부분이 있어 예측하기 힘든 부분이 있긴 하지만 대부분은 데이터베이스에 설정되어있는 대로 작용하기 때문에 전투를 설계하기 전에 적중률과 회피율을 확인해 아이템이나 스킬로 영향을 줄 수 있는지 확인해야 할 것입니다.

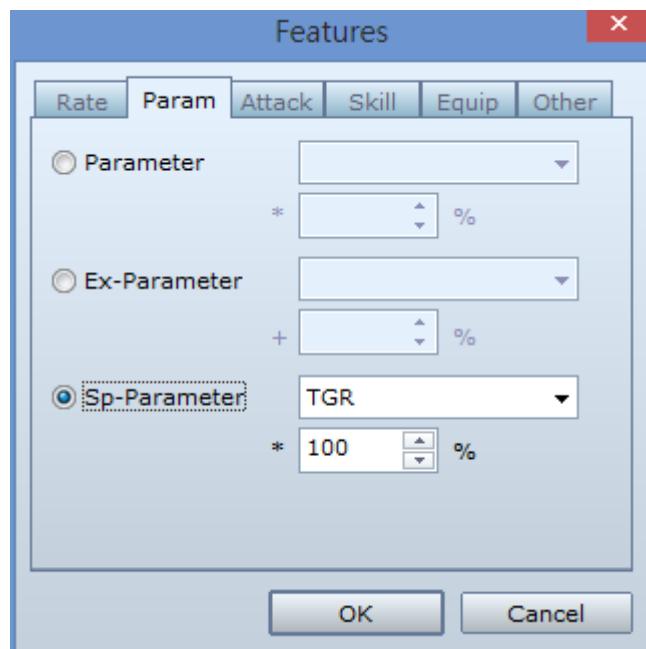
Critical rate 과 Critical EVasion rate은 결정적인 타격을 가할 수 있는지와 결정적인 타격을 피할 수 있는지를 판단할 때 사용되는 비율값입니다. 크리티컬 비율이 크리티컬 회피율(CEV)보다 높아야 크리티컬 공격을 성공적으로 가할 수 있는 확률이 증가합니다. 이 판정은 Game_Battler의 item_cri라는 메소드에서 관리합니다. 또한 크리티컬 확률은 기본적으로 일반 공격의 3배에 해당하는 공격력을 가집니다. 이 3배라는 수치를 고치고 싶다면 Game_Battler 스크립트에서 apply_critical 메소드의 값을 변경하면 됩니다.

Magic ReReflection rate (마법 반사)와 CouNTer attack rate (반격 확률)은 아군이 아이템이나 스킬을 사용할 때 적군이 마법 반사나 반격을 취할 확률과 관계된 수치입니다. 아군이 물리 스킬 공격을 가했을 때 임의 난수값보다 CNT 비율이 높을 경우 적군은 반격을 하게 되고 아군이 마법 스킬 공격을 가했을 때 임의 난수값보다 MRF 비율이 더 높을 경우 마법 반사를 하게 됩니다. 위에서 말한 두 경우에 해당하지 않을 경우 성공적으로 아이템이나 스킬을 성공적으로 사용하게 됩니다.

Hp ReGeneration rate (HP 재생 비율)과 Mp ReGeneration rate (MP 재생 비율) 그리고 Tp ReGeneration rate (TP 재생 비율)은 매 턴이 끝나면 자동적으로 HP, MP, TP 가 일정량 상승하게 되는데 이 비율값들은 그 값과 관계되어 있는 비율입니다. 재생값은 최대 HP, 최대 MP, 최대 TP(100)에서 비율값을 곱해서 구하게 되며 이 수치값들은 Game_Battler 스크립트의 regenerate_hp, regenerate_mp, regenerate_tp 메소드에 정의되어 있습니다.

4.7. Param - Sp Parameter (특별 능력치)

Sp Parameter (특별 능력치)는 기본적으로 100%로 설정되어 있습니다. 하지만 특징(Features) 탭에서 게임 세계관과 캐릭터 특성에 맞게 비율값을 변경할 수 있습니다.



TarGet Rate (타겟 비율)은 적에게 타겟이 될 확률에 영향을 주는 비율입니다. Game_Unit 스크립트의 random_target 메소드에서 공격 타겟을 설정할 때 사용됩니다. 이 비율값이 낮으면 공격 타겟이 될 확률이 줄어듭니다. 예를 들면 Hide 상태에서는 적의 타겟으로 잘 설정되지 않으므로 공격 받을 확률이 줄어들게 됩니다.

GuaRD effect rate (기본 방어 효과율)은 턴제 전투에서 방어 커맨드를 선택한 경우 적용되는 비율입니다. 적군에게 타겟이 되어 공격을 받게 될 때 방어 커맨드를 선택한 경우 방어 효과율이 적용되면서 적에게 받는 데미지가 감소하게 됩니다. 이 비율이 높으면 방어 시 적에게 조금 더 적은 데미지를 받게 됩니다. 이 비율은 Game_Battler의 apply_guard 메소드에 영향을 줍니다.

RECovery effect rate (기본 회복 효과율) 회복 효과율은 회복 계열의 마법이나 아이템을 사용했을 때 얻는 최초의 회복 수치에 곱해지는 비율값입니다. 쉽게 말하면 '기본 회복 수치에 영향을 주는 비율값'입니다. 영향을 준다는 것은 회복 스킬의 영향력을 크게 늘리거나 줄일 때 사용할 수 있다는 소리입니다. 기본 회복 효과율이므로 회복 스킬 마법, 포션 류 아이템의 회복 수치에 모두 영향을 주게 됩니다.

PHarmacology (약리학 = 약물로 체력이나 마력을 회복할 경우의 회복 효과율)은 약리학은 간단히 말하면 체력 포션이나 마나 포션 등과 같은 아이템으로 HP나 MP를 회복할 때 그 회복 수치에 영향을 주는 비율값입니다. 최초 회복 수치는 기본 회복 효과율과 데이터베이스에 적힌 값을 곱해서 만들어지게 되는데 그 최초 회복 수치 값에 pha 비율값이 곱해집니다. 이 비율값을 잘 이용하면 '아이템'을 사용하여 체력이나 마력을 회복할 때 더 많은 회복 효과를 누리게 만들 수 있습니다. 반대로 아이템을 이용하면 회복 효과를 덜 누리게 만들 수도 있습니다.

MP Cost Rate (MP 소비율)은 스킬을 사용했을 때 스킬 소비량에 곱해지는 비율값으로 이 비율값이 높으면 스킬을 사용할 때 드는 MP 소비량이 증가하게 되고 낮으면 MP 소비량이 감소하게 됩니다.

Tp Charge Rate (TP 충전율)은 아이템이나 스킬을 사용하여 TP를 충전할 때 획득하는 최종 TP 값에 영향을 주는 비율값입니다. TP(테크니컬 포인트)는 일부 직업군에서 특수 마법을 사용할 때 필요한데 TCR 수치가 낮으면 아이템이나 스킬을 통해 획득할 수 있는 TP량이 낮아지게 되므로 전투 초기에 TP 회복 스킬을 써서 많은 TP를 확보한 후 강력한 스킬을 쓰게 되는 행동을 막을 수 있습니다. 즉, TP 충전율이 낮으면 많은 TP가 필요한 직업군에겐 치명적일 것이며 이는 전투 난이도로 이어질 수 있습니다.

Physical Damage Rate (물리 데미지 비율)은 물리 공격을 가했을 때 곱해지는 비율 값으로 물리 공격량에 영향을 주는 비율입니다. 이 값이 높으면 물리 공격 시 더 높은 데미지를 줄 수 있게 됩니다.

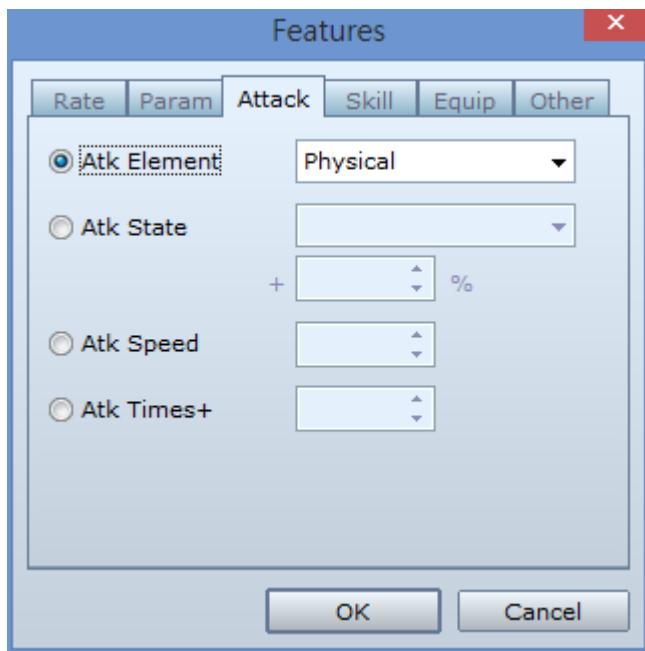
Magical Damage Rate (마법 데미지 비율)은 마법 공격을 가했을 때 영향을 주는 비율값으로 마법 공격력에 영향을 주게 됩니다. 이 비율값이 높으면 마법 스킬을 사용했을 때 더 높은 데미지를 가할 수 있습니다.

Floor Damage Rate (바닥 데미지 비율)은 데미지 바닥 타일(Damage Floor)이 설정된 타일을 밟았을 때 플레이어에게 얼마큼의 데미지를 줘야 할지를 결정하는 비율입니다. 기본 데미지 값 10에 이 비율값(FDR)이 곱해지게 됩니다. 따라서 비율값이 높아지면 바닥 데미지 타일을 밟을 경우 데미지를 더 많이 받게 된다는 것을 알 수 있습니다.

EXperience Rate (경험치 획득 비율)은 전투 종료 후 획득 할 최종 경험치량에 영향을 주는 비율입니다. 이 비율값은 최종 획득 경험치 계산에 막대한 영향을 주므로 높을 경우 더 많은 경험치를 받게 되고 낮을 경우 더 적은 경험치를 얻게 할 수 있습니다.

4.8. Attack 탭

공격과 관련된 전투 요소를 설정할 수 있는 탭입니다.



Atk Element 는 공격 속성을 뜻합니다. 공격 속성을 부여하면 일부 몬스터에게 큰 데미지를 입힐 수 있게 됩니다. 이 개념은 속성 유효도를 설명할 때 이미 설명하였습니다.

Atk State 는 공격 시 상태를 부여할 수 있는 옵션입니다. (상태는 여러가지 특성에 영향을 줍니다)

Atk Speed 는 공격 속도와 관련되어있습니다. 먼저 전투에서 일반 공격을 선택하게 되면 민첩력이 높은 멤버순으로 정렬된 상태에서 순서대로 적에게 공격을 퍼붓게 됩니다. 민첩력이 높은 멤버가 먼저 공격을 하려 가게 되는 구조로 되어있는 데 Atk Speed 를 설정하면 일반 공격 시 Atk Speed 에 설정된 값도 민첩력과 함께 더해지게 되므로 공격 행동을 취하게 되는 순서를 바꿀 수 있습니다. Atk Speed 는 -999 ~ 999 사이의 값을 설정할 수 있습니다.

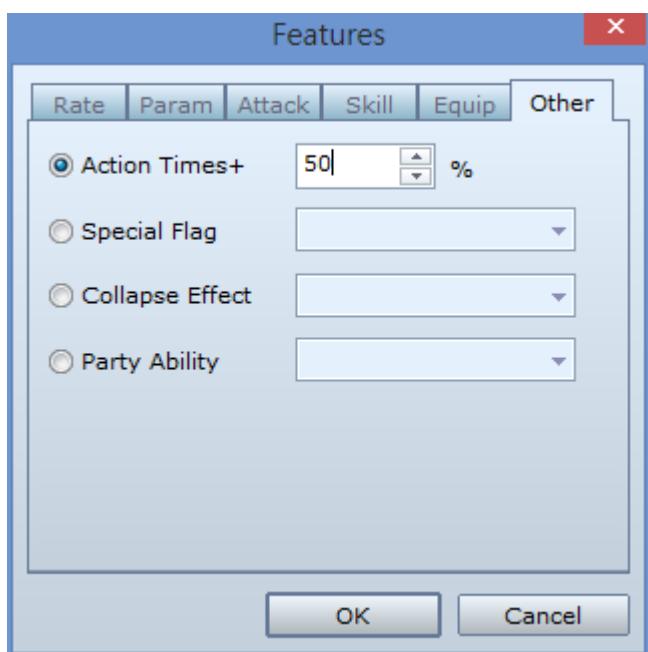
Atk Time + 는 공격 기회를 나타냅니다. 기본적으로 타겟당 최소 한 번의 공격 기회를 가집니다. 하지만 Atk Time + 를 설정하면 추가 공격 기회를 가질 수 있게 됩니다. 9로 설정하면 최대 10번의 공격 기회를 가질 수 있기 때문에 최대 10번 반복 공격을 하게 됩니다.

4.9. Skill, Equip 탭

특정 스킬을 추가하거나 사용하지 못하게 만들 수 있습니다. 이 탭을 통해 어떤 신체 부위에 장비를 착용하지 못하게 만들 수도 있습니다. 직관적으로 되어있기 때문에 이 부분은 사실상 추가 설명이 필요없을 것 같습니다.

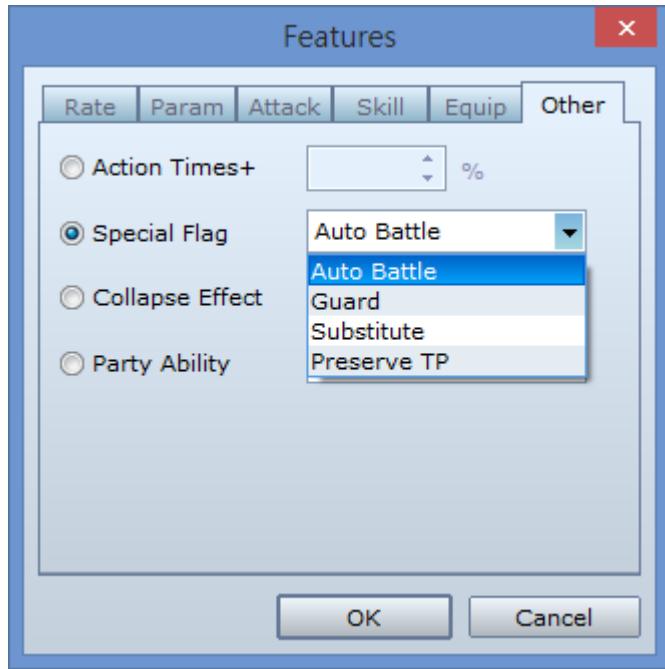
4.10. Other 탭 - Action Time+

보통 턴제 전투 시에는 파티원 한 명 당 한 번의 행동(일반 공격, 스킬 사용, 아이템 사용 등) 기회를 가집니다.



Action Time+ 를 설정하지 않으면 보통은 한 번만 선택할 수 있습니다. 하지만 Action Time+ 을 통해 확률을 설정하면 일정 확률로 추가 행동을 선택할 수 있습니다. 이렇게 되면 총 2번 선택할 수 있게 됩니다.

4.11. Other 탭 - Special Flag (특수 플래그)



Auto Battle 플래그가 설정되어있으면 전투를 자동으로 진행하게 됩니다.

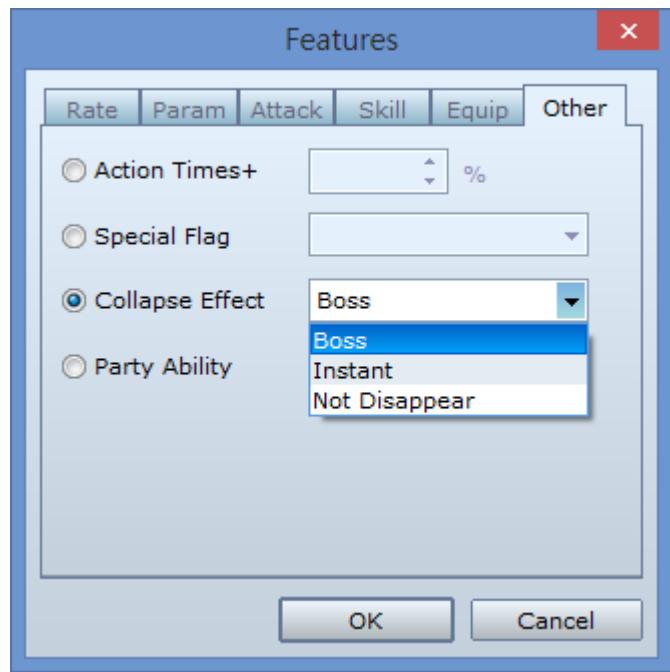
Guard 플래그는 방어 커マン드를 선택한 것과 동일한 효과로 방어 커マン드를 설정했을 때 상태가 방어 상태로 바뀌면서 이 플래그가 자동으로 설정됩니다. Game_Battler 스크립트의 apply_guard 메소드에 적힌 내용대로 데미지가 줄어들게 됩니다.

Substitute 플래그는 HP가 적은 아군 대신 피해를 받게 하는 플래그입니다. 정확히 말하면 '체력이 최대 체력의 4등분보다 작은 아군'을 대상으로 합니다.

Preserve TP 플래그는 TP를 유지하는 플래그로 원래 전투 시작 시, 0 ~ 25 사이의 TP를 난수값으로 받지만 이 플래그가 설정된 캐릭터는 이전 전투의 TP (테크티컬 포인트, 기술을 사용하려 할 때 필요한 값) 값을 그대로 이어받을 수 있게 됩니다.

4.12. Other 탭 - Collapse Effect

적이 죽을 때 적군의 그래픽이 어떻게 사라져야 하는지를 결정하는 옵션으로 주로 Enemy 탭에서 설정됩니다.



Boss 방식은 적군의 그래픽이 땅으로 천천히 꺼지면서 사라지는 방식입니다.

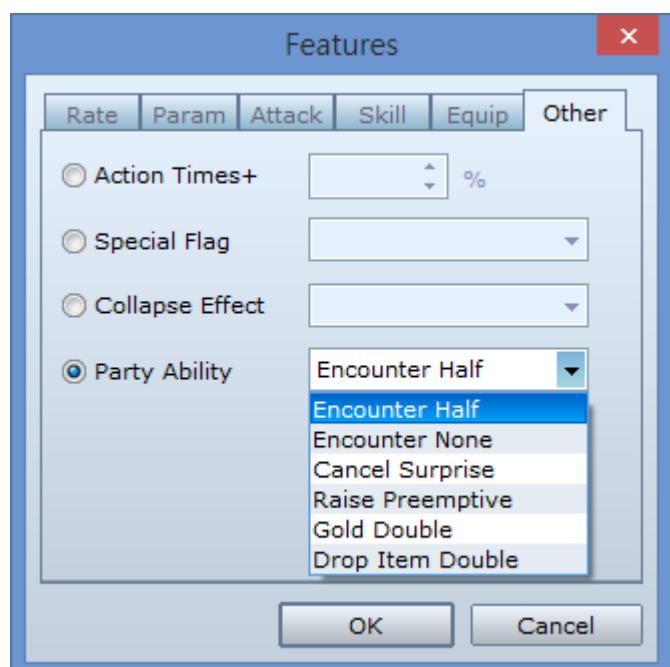
Instant 방식은 적군의 그래픽을 즉시 제거하는 방식입니다.

Not Disappear 방식은 적군의 그래픽을 그대로 유지하는 방식입니다.

방식을 설정하지 않으면 투명도가 감소되면서 천천히 사라지게 됩니다.

4.13. Other 탭 - Party Ability

전투 시작과 관련된 옵션과 보상과 관련되어있는 옵션입니다.



Encounter Half 와 Encounter None 은 엔카운터 시스템과 관련된 옵션입니다.

Encounter Half (엔카운터 반감)은 전투 발생 빈도를 절반으로 줄이는 옵션입니다. 30 발자국에 한 번 전투가 발생하는 시스템에서 한 걸음 당 1번씩 카운팅된다면 이 옵션이 체크되어있으면 0.5 걸음씩 카운팅되면서 결과적으로 전투 발생 빈도가 줄어들게 됩니다.

Encounter None (엔카운터 없음)은 엔카운터 시스템을 사용하지 않는 옵션이며 엔카운터가 설정되어있어도 전투가 발생하지 않습니다. 이 효과는 테스트 플레이에서 컨트롤(Ctrl) 키를 누르고 걷는 것과 동일한 효과입니다.

Cancle Surprise (기습 공격 중지)는 턴제 전투에서 기습 공격을 당하지 않게 해주는 옵션입니다. 이 옵션이 참이라면 파티의 민첩력이 '확률 비교를 위한 난수값'보다 항상 낮기 때문에 기습 공격을 절대 당하지 않게 됩니다. (100% 확률)

Raise Preemptive (선제 공격 확률 올리기)는 선제 공격 확률을 높이는 옵션입니다. 해당 파티의 민첩력이 4배 상승하게 되면서 선제 공격권을 가지게 될 확률이 높아지게 됩니다. (12% ~ 20% 의 확률로 선제 공격권을 가지게 됩니다)

Gold Double 은 말 그대로 전투 보상금을 더블로 받는 옵션이며 Drop Item Double 도 마찬가지로 말 그대로 전투 보상 아이템을 더블로 받는 옵션입니다.