Course Glossary

The following is a list of concepts covered in each lesson as is intended to be used as a glossary of the course. Use the 'Find In Document' feature to locate a keyword or concept and refer to the associated lesson when required.

Contents

Lesson 1: Setting Up a Title Screen

- Creating a New Project
- Creating a New Scene
- Assigning a Background to a Scene
- Assigning a Player Sprite Sheet to a Scene
- Assigning an On-Screen Menu Cursor using the Player
- Animating a Stationary 'Press Start' Actor in a Scene
- Shifting the Camera View of a Scene using Player Input
- Running a Build of your Project

Lesson 2: Preparing an Options Menu

- Triggers
- Attaching Scripts to Buttons
- Variables
- Custom Scripts
- Adding Sound Effects
- If Statements
- Adding Music
- Sound Effect Formats (VGM)
- Sound Effect Formats (FX Hammer)
- Sound Effect Formats (WAV)
- Copy/Pasting Scripts
- Calling Custom Scripts
- Changing Scenes
- Finalizing the Title Screens 'On Init' Script

Lesson 3: Implementing and Tracking Options

- Tracking Menus With A Global Temporary Variable
- Debugging Your Game
- Avoiding Two Or More Menu Triggers Executing At Once
- A Note On The SFX On/Off Menu Setting
- Saving Sprite Frames In A Scene When The Player Is Hidden

Lesson 4: Implementing Cutscenes

- Engine Field Updates - Fade Style
- Skipping Cutscenes
- The Overlay Layer
- The 'Monochrome Palette' (Or Switching Between "Light Green" And "Dark Green" Tone On DMG)
- Parallax
- A Note On Placing Actors Over Areas Of Background Parallax
- Using Event Groups To Improve Script Navigation & Code Readability
- Using Actor Directions To Manage Actor Animation
- Changing The 'Wait' Event's Duration From Seconds To Frames For More Precise Control
- Setting The Speed And Font Of Text Within A 'Display Dialogue' Event
- Setting Actor Positions By Pixel Instead Of Tile
- Assigning Sound Effects To Character Dialogue

Lesson 5: Player Animation States and the Platformer Scene

- Setting Up A Platformer Character's Animation States
- The Scanline Sprite Limit
- The Platformer Scene Type
- Sprite Limits In A Platformer Game
- The Platformer Player Character Animation States
- Applying Collision To A Platformer Scene
- Button Prompts (One Of Many Design Solutions)
- (Partially) Fixing The "Walking On The Spot" Problem
- Dialogue Avatars
- Editing The ASCII Font Sheet
- The De-Loaded Actor Problem
- The (Very Important) 'Script Lock' Event
- Tracking NPC Dialogue Using Variables
- Using Switches To Manage Three Or More Conditional Scripts
- Emote Bubbles

Additional concepts covered in Lesson 5 Overview Document:

- Default Platformer Player Characters Animation States
- The Scanline Sprite Limit
- The Platformer Engine Settings

Lesson 6: The Platformer+ Plugin

- Fixing the "Head Bonk" Problem
- A Note on Background Parallax
- The Platformer+ Plugin
- Replacing (Swapping) Tiles
- Locking the Camera Horizontally in a Platformer Scene
- Attaching scripts to the Platformer+ states
- Using projectiles to create particle effects
- Attaching SFX to P+ States to Improve "Game Juice"
- Adding a 'Jump Out of Dash' Mechanic
- Pau-Tomas' Player Fields Plugin
- Precise Control Over Player Animation States
- Editing Engine Fields
- Prefab Actors
- The Benefits of Using Flags to Track Boolean (True or False) States
- 'What' vs. 'How Many' (Flags vs. Values)
- Launch Projectile Presets
- An Important Note on Hidden Actors
- Creating One-Way Platforms Using Actors (and P+)
- Changing Collision Bounding Boxes (or Hitboxes)

- Calling Variables in Character Dialogue

Lesson 7: Implementing a HUD

- Storing the Player Location Throughout a Game
- A Note on Continuity Between Scenes (or Paying Attention to Design Details)
- Updating Variables to Suit Checkpoints
- Displaying a HUD Using the 'Set Submap' Method
- Displaying the Player's Current Location in the HUD
- Displaying Icons and Other Static Information in the HUD
- Displaying Number Strings in a HUD
- The 'Modulus' Operator
- Replacing Tiles According to Tileset Values
- Conditional HUDs
- GBVM Scripts
- GBVM & Variable Value Ranges (or What's INT8 & INT16?)
- HUD Positioning, Top or Bottom?
- Why the 'P+ States' Custom Script Needs To Be In All Scene 'On Inits'

Additional concepts covered in the Lesson 7 Overview Document:

- Color Modes
- How to Set Palettes to "True Color" when Exporting to Web

Lesson 8: Collectibles and Updating the HUD

- Updating and Displaying a Value in the HUD
- Implementing Collectibles Using the "Trigger System"
- Coding Collectibles, 'One Time Collection' or 'Refresh On Init Collection'
- A Note on Clearing and Adding Variable Flags (or Flag Values)
- Trigger Prefabs
- Avoiding Value Display Overflows in the HUD (or Using the min() and max() Maths Functions)
- Optimizing HUD Updates
- Muting Audio Channels
- A Note on Collectible Hitbox Sizes
- Pausing Gameplay for Cutscenes (or Other Events)
- The 'Script Lock' and 'Script Unlock' Events (a Method of "Hitstop" Implementation)
- Using Projectiles to Reduce Actor Use in a Scene
- Expanding the Player Move-set in a "Gated Exploration" Game

Lesson 9: Implementing a Shop

- Actor Animation State Names (a Handy Tip)

- A Note on Event Based Code Readability & Unnecessary Event Duplication
- Color Palettes
- Setting Sprite & Background Palettes
- Dialogue Menus
- Advanced Dialogue Menus
- Displaying Conditional Prices in a Shop Menu
- Jumping to Labels Using GBVM
- Pitfalls of Advanced Menus (Or What Happens When The Player Presses 'B'?)

Lesson 10: Creating a Level Select Menu

- Implementing A World Map Level Select System
- Managing the Level Select Screens Sate According To Player Progress
- Avoiding GB Studio's "Camera Shake Glitch"
- Designing The User Interface Of A Level Select Screen
- Implementing A Level Select Route Reveal Sequence
- Looking For Ways to Add Flair To A Level Select Screen
- Implementing A Level Select Route Reveal Sequence
- Using Threads To Run Looping Background Layer Animations
- Using Triggers To Navigate A "World Map" Style Level Select Screen
- Displaying Context Sensitive Level Information In A Level Select Screen
- Declaring Constant And Non-Zero Values To Manage Collectibles (And More)
- Implementing The Platformer Win State (Or How To Exit Stage Right)
- Pixel-Shifting Actors (Or The Easy Way To Adjust Sprite-Based Background Graphics)

Lesson 11: Creating Obstacles

- Creating A Level Template Scene
- Implementing A Pause Game System Using The "Display Dialogue Method"
- Efficient Soundtrack Implementation
- Implementing A Player Fail State Using Triggers (Or Instant Death)
- Moving An Actor Using Actor Properties And Value Modifiers
- Setting Up A New Level Scene
- Trigger Based "Instant Death" Obstacles
- Actor Based "Instant Death" Obstacles
- Creating A Falling Platform Using Local Variables
- The Dangers Of Using Local Variables
- A Few Pathways To Solving Bugs (Including An Example)
- A Note On Non-Modal Dialogue Boxes When Changing Scenes

Lesson 12: Creating Enemies

- Implementing Background Animations In Platformer Scenes
- Simulating Foreground And Background Graphics Using Background Tile Priority
- Offsetting The Platformer Scene Camera View (And Why It's So Important!)
- Implementing Enemy Projectiles (By Applying Scripts To A Scene's Collision Groups)
- Implementing Enemy Behavior Using An Actor's 'On Update' Script
- Maximizing Unique Enemies With Minimal Sprites In A Scene (Or Efficient Enemy Visual Design)
- Creating A Universal Enemy Self-Destruct Script
- Creating Bouncing Enemies Using The 'Actor Move To' Event
- Creating Shooting Enemies Using An Actor's 'On Update' Script
- Using The Actor Direction Property To Create Enemy Variation
- Creating Flying Enemies Using The 'Actor Move To' Event
- Creating Flying Enemies Using The 'Launch Projectile' Event

Lesson 13: Taking Your Game To The Next Level

- Menus, Shops and Cutscenes (Course Concepts Reviewed)
- A Note On Player Knock-back Systems
- Implementing Health Points (HP)
- Implementing Player Invincibility Frames (i-frames)
- The Importance Of Commenting Code And Custom Script Summary Notes
- Iterating On Existing Mechanics
- Implementing Complex Player Moves (Ground Pound)
- Enabling The Ground Move Only When It Is Unlocked
- Preventing Bugs Using The 'Lock Out Gate Variable' Method
- Precise Use Of The Platformer State Engine
- Debugging The Lock Out Gate And Player Attack State
- Updating Engine Fields, Player Fields, & Engine Variables To Temporarily Modify The Platformer Engine
- Polishing The Ground Pound Move (Yes, More Debugging)
- A Deeper Understanding Of The Platformer Animation States
- Music and SFX (Course Concepts Reviewed)