

Intro Construct 2

เว็บสำหรับการสอนการใช้โปรแกรม Construct 2 สำหรับมือใหม่และผู้ใช้ทั่วไป

นำเสนอ

ผศ.ดร.มานพ พันธโคกรวด

รายงานนี้เป็นส่วนหนึ่งของวิชา 06016312 เทคโนโลยีเว็บ (WEB TECHNOLOGY)

นักศึกษาชั้นปีที่ 2 คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 1 ปีการศึกษา

ชื่อรายงาน : Intro Construct 2

ผู้จัดทำ : 1. นายธนา สามารถ 61070079
2. นายชิตวร สถาพรณิชย์ 61070041
3. นายกิตติภูมิ ชูอมร 61070015
4. นางสาวณิชนันท์ สาครน้อย 61070057
5. นายสุฤตสิน นุพงศ์ 61070253

นักศึกษาชั้นปีที่ 2

คณะ : เทคโนโลยีสารสนเทศ

สาขา : เทคโนโลยีสารสนเทศ

ที่ปรึกษา : ผศ.ดร.มานพ พันธุ์โคกกรวด

ภาคเรียนที่ 1 ปีการศึกษา 2652

ที่มา

การสร้างหรือออกแบบเกมนั้นต้องอาศัยหลายองค์ประกอบและลำดับขั้นตอนทั้ง การเริ่มตีกรอบความกว้างของเนื้อหาเกมและแนวเกม (Scope) การคำนึงถึงความสามารถของทีมผู้พัฒนา (Skill) การออกแบบกราฟิกเกม (Design) และส่วนสำคัญคือ เครื่องมือ(Tool หรือ Game Engine) ผู้พัฒนาเกมหรือ เดฟ(Dev.) ต้องมีความรู้ความเข้าใจในตัวเครื่องมือ เพื่อเมื่อเกมเกิดปัญหาจะสามารถแก้ได้อย่างรวดเร็วและมีประสิทธิภาพ

ในปัจจุบันมี เครื่องมือในการพัฒนาเกมนั้นมีให้เราได้เลือกใช้มากมาย ทำให้ผู้พัฒนาสามารถเลือกได้ตามความเหมาะสมให้ตรงกับทรัพยากรที่มี โดยโปรแกรม Construct 2 ก็เป็นหนึ่งในโปรแกรมพัฒนาเกมที่เข้าถึงง่ายและเหมาะสมในการเป็นเครื่องมือเริ่มต้นของผู้พัฒนาเกมหน้าใหม่ที่สามารถเข้าถึงได้ง่ายเพียงแค่เข้าใจหลักการเงื่อนไขของ if และ else ด้วยเหตุนี้คณะผู้จัดทำจึงได้จัดสร้างเว็บไซต์สำหรับเป็นสื่อการเรียนการสอนวิธีการสร้างเกมด้วยโปรแกรม Construct 2 ในชื่อ Intro Construct 2

โดยจะมีเนื้อหาที่เรียงลำดับความสำคัญดังนี้ เริ่มต้นด้วยการทำความรู้จักกับโปรแกรม Construct 2 เพราะผู้จัดทำได้ให้ความสนใจในความรู้และความเข้าใจในเครื่องมือที่เลือกใช้ เป็นสำคัญ และตามด้วยเนื้อหาอื่น ๆ เช่น การสร้างและการกำหนดคุณสมบัติของ Object, การจัดการเหตุการณ์และความเป็นไปภายในเกม การสอนแนวคิดพื้นฐานในการสร้างเกมเพื่อเพิ่มความเข้าใจให้กับผู้ใช้งานมากยิ่งขึ้น

วัตถุประสงค์

1. สามารถสร้างบทสรุปข้อมูลการใช้โปรแกรม Construct2 ได้
2. เว็บไซต์ที่สร้างขึ้นสามารถใช้ประโยชน์ได้จริง
3. สมาชิกของผู้จัดทำได้ทบทวนเนื้อหา และวิธีการใช้งานโปรแกรม Construct2 ให้ชัดเจนยิ่งขึ้น

ใช้เทคโนโลยีที่ใช้ในการทำโครงการงาน

1. ด้านการสืบค้นข้อมูล

Internet โดยที่ 2 ทางหลักๆ เว็บไซต์ scirra.com และ youtube ใช้สำหรับสืบค้นข้อมูลพื้นฐาน และ ตัวอย่างเกี่ยวกับเป็นแรงบันดาลใจและแนวคิดในการพัฒนาเกม

2. ด้านการพัฒนาเกม

โปรแกรม Construct2

3. ด้านภาพและสื่อประสม

โปรแกรม Camtasia 9

โปรแกรม Snapping tool

โปรแกรม Paint

โปรแกรม Photo shop

ประโยชน์ของโครงการงาน

1. ประโยชน์ต่อคณะผู้จัดทำ

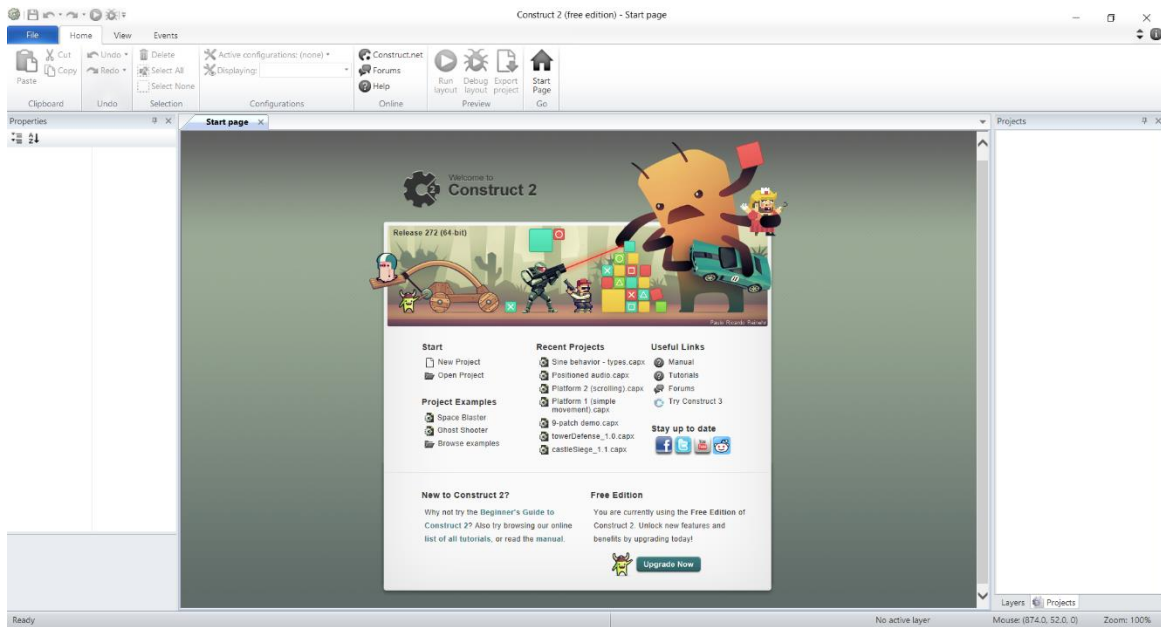
1. ได้บททวนและเรียนรู้เพิ่มเติมเกี่ยวกับการใช้งานโปรแกรม Construct2
2. ได้สรุปเนื้อหาและจัดทำสื่อการเรียนการสอนสำหรับใช้ในงาน ITCamp ปีถัดไป

2. ประโยชน์ต่อส่วนรวม

1. ผู้อ่านได้รับความรู้ในเรื่องการใช้งานโปรแกรม Construct2
2. ผู้อ่านได้มีความเข้าใจในแนวคิดและองค์ประกอบการสร้างเกมมากยิ่งขึ้น
3. ได้มีสื่อการเรียนการสอนสำหรับใช้งานในค่าย ITCamp

Chapter 1: Introduction

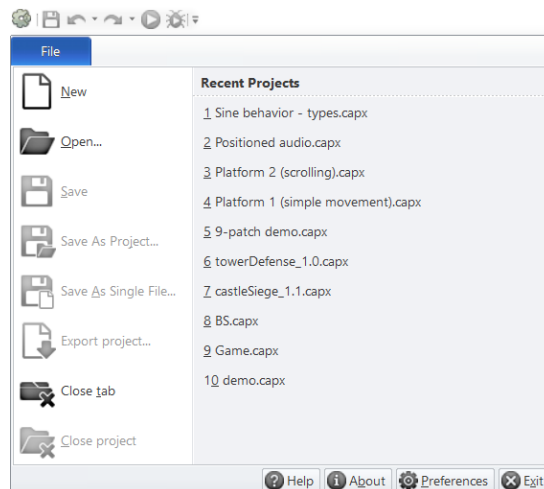
Construct 2 นั้นเป็นเครื่องมือที่ช่วยในการสร้างเกม สิ่งสำคัญในการใช้ ‘เครื่องมือ’ นั้นคือการเข้าใจเกี่ยวกับคุณลักษณะ และความสามารถต่าง ๆ ของมัน ดังนั้นเราจะเริ่มกันที่ Interface ของ Construct 2



(รูป 001 Construct 2 Interface)

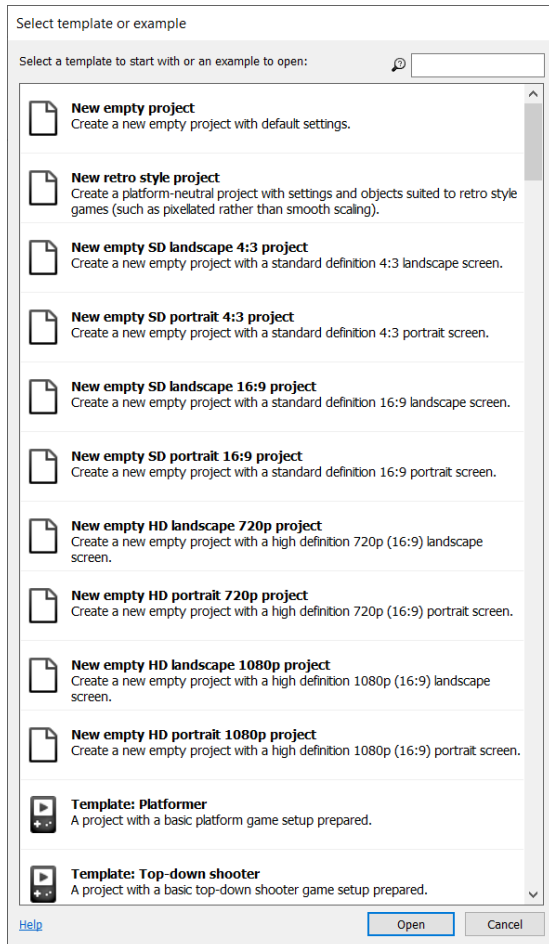
File

คลิกที่มุมขวาบนของโปรแกรม เมื่อเปิดจะแสดงหน้าต่างดังนี้



(รูป 002 หน้าต่าง File)

และให้กดที่ New จากนั้นจะเป็นการเลือกชนิดของ template โดยจะมีรายละเอียดดังนี้



(รูป 003 หน้าต่างเลือกชนิดของ template)

New empty project

สร้างโปรเจกใหม่โดยใช้ค่าพื้นฐาน (โปรเจกเปล่า)

New retro style project

สร้างโปรเจก โดยมีการเติมข้อมูลบางส่วนให้

New empty SD / HD

สร้างโปรเจกเพื่อให้เหมาะกับการแสดงผลผ่านอุปกรณ์ต่าง ๆ

Landscape	แนวนอน
Portrait	แนวตั้ง (เกมมือถือ)
SD	มีความละเอียดที่ 720 x 480p
HD	มีความละเอียดที่ 1,280 x 720p
Full HD	มีความละเอียดที่ 1,280 x 1080p
4 : 3, 16 : 9	อัตราส่วนภาพ ยาว * สูง

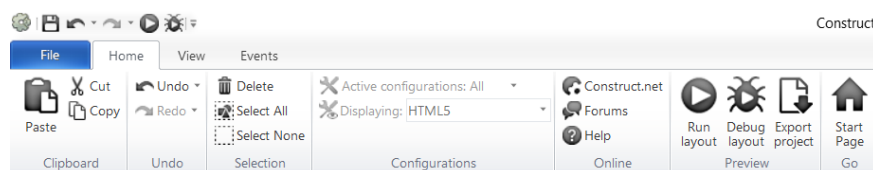
Example game or Template

ตัวอย่างเกม และ การใช้ฟังก์ชันต่าง ๆ

Ribbon tool

เป็นแถบเครื่องมือหลักจะมีทั้งหมด 3 ส่วนคือ home, view, และ event

Home



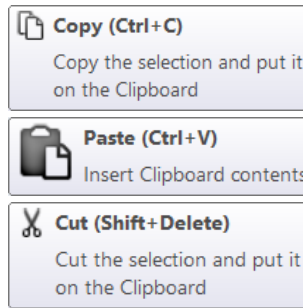
(รูป 004 Ribbon tool > Home)

Clipboard

Copy การคัดลอก เก็บไว้ใน คลิปบอร์ด

Paste การวาง

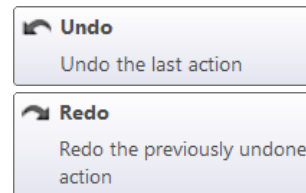
Cut ตัดออก แล้วเก็บไว้ใน คลิปบอร์ด



Undo

Undo ยกเลิกการกระทำนั้น ๆ และย้อนกลับ

Redo ทำซ้ำอีกครั้ง (ใช้หลังจากการกด Undo)

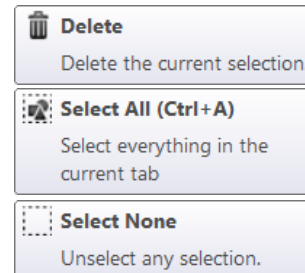


Selection

Delete กดเพื่อลบ Object ที่เลือก (delete)

Select All เลือกทุก Object ใน layout นั้น ๆ

Select None ยกเลิกการเลือกทุก Object



Preview

Run layout การสั่งเล่นไฟล์ปกติ (F5)

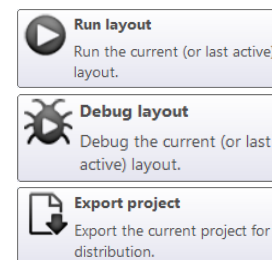
Debug layout การสั่งเล่นไฟล์ในโหมดดีบัค (Ctrl + F5)

ทำให้สามารถตรวจสอบข้อมูลทั้งหมดของทุก Object

ในขณะนั้น ใช้สำหรับหาจุดผิดพลาดของเกม

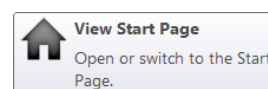
Export project ส่งออกไฟล์ ให้สามารถทำงานบน platform ต่าง ๆ

Web / Mobile /Mobile & Desktop /Desktop



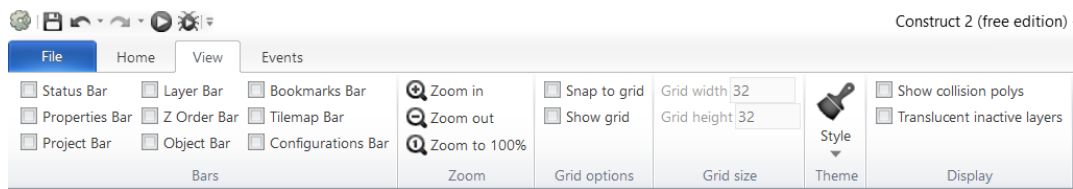
Go

Start Page เปิดหน้าเริ่มต้นของโปรแกรม



View

ส่วนที่ใช้ควบคุมภาพรวมของหน้าต่างโปรแกรม



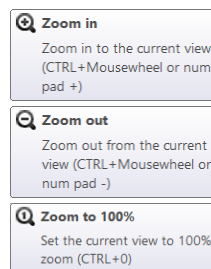
(รูป 005 Ribbon tool > View)

Bars

เป็น Check box หากติ๊กถูก จะเป็นการแสดง ส่วนนั้น ๆ ในหน้า Interface และสามารถจัดตำแหน่งได้

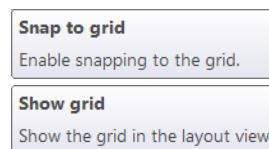
Zoom

Zoom in	ขยายเข้า
Zoom out	ย่อออก
Zoom to 100%	ปรับ ซูมเป็น 100%



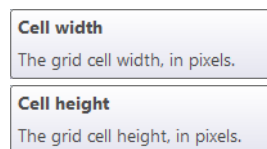
Grid options

Snap to grid	ทำให้การเคลื่อนย้าย Object จะขึ้นตรงกับ grid
Show grid	แสดง grid



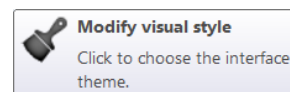
Grid size

Grid width	กำหนดความกว้าง
Grid height	กำหนดความสูง



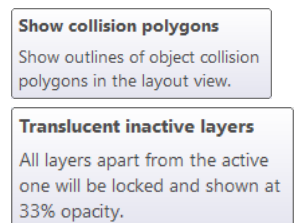
Theme

รูปแบบ Interface ของ โปรแกรม Construct 2

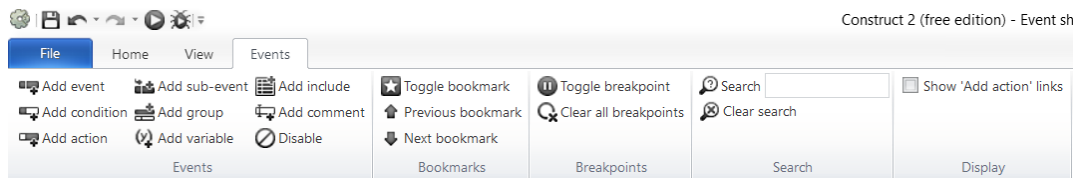


Display

Show collision polys	แสดง Polygon(hitbox) ของทุก Object
Translucent inactive layers	ทำให้สามารถ object นอกจาก layer ที่เลือกไม่ทำงาน



Events



(รูป 006 Ribbon tool > Events)

Events

Add event

สร้าง event ใหม่ใน event sheets

Add condition

สร้างเงื่อนไขเพิ่มเติมให้ event ที่เลือก

Add action

สร้างการตอบสนองกรณี มีเหตุการณ์เกิดขึ้นตรงกับเงื่อนไขของ event นั้น ๆ

Add sub-event

เงื่อนไขซ้อน

Add group

สร้างที่เอาไว้เก็บ รายการ event ต่าง ๆ เพื่อง่ายต่อการจัด / แก้ไข

Add variable

สร้าง Global variable

#Global variable

ตัวแปรของระบบและมีผลกับทุก Event sheets ของไฟล์

Add include

ใช้สำหรับการเรียกใช้ ชุด event เดิม โดยไม่ต้องเขียนใหม่

Add comment

เพิ่มคำอธิบาย

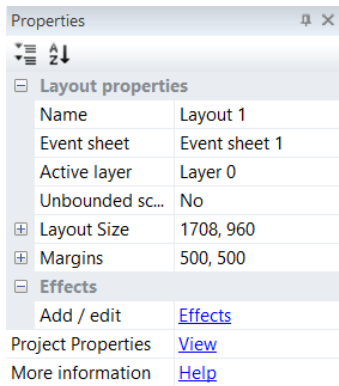
Disable

ทำให้ event นั้น ๆ ไม่ทำงาน

(รูป 007 Global variable)

Properties bar

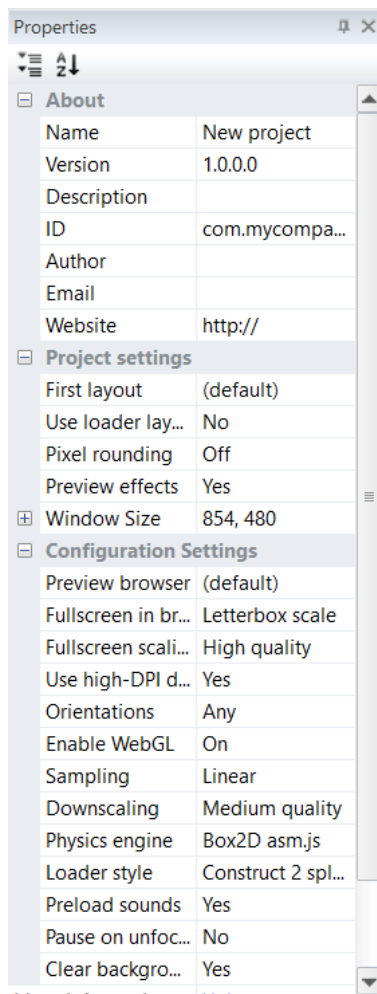
แถบแสดงสถานะ มีทั้งหมด 3 ประเภทคือ layout, Project, และ Objects



(รูป 008 Layout properties)

Layout properties

1. Name กำหนดชื่อของ layout
2. Event sheet เลือกใช้ Event sheet
3. Active layer แสดง layer ปัจจุบัน
4. Margins พื้นที่ว่างรอบ Layout
5. Effects เพิ่มและแก้ไข Effect ของ Layout



(รูป 009 Project properties)

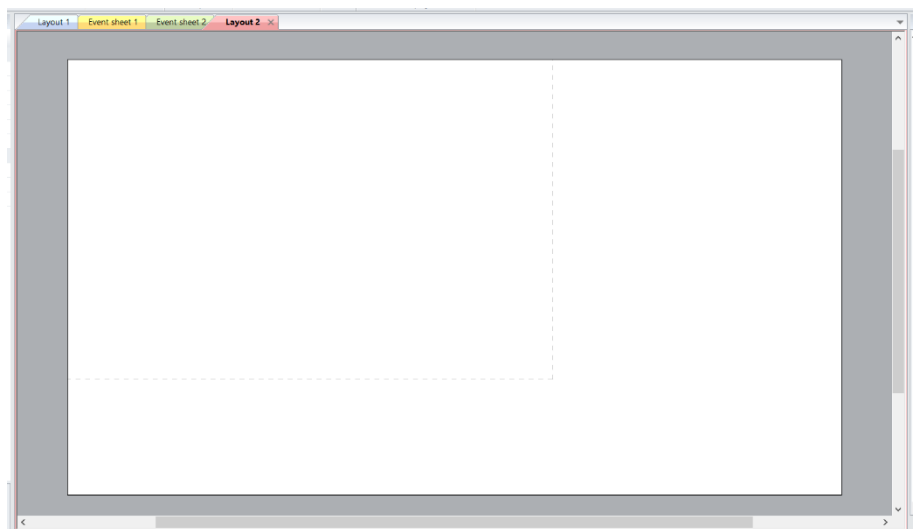
Project properties

About

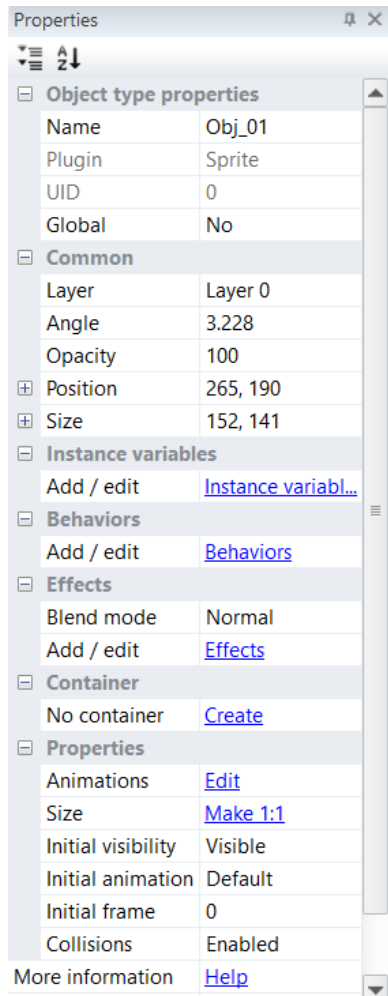
1. Name กำหนดชื่อของไฟล์เกม

Project setting

1. First layout เลือก layout แรกที่จะแสดงตอนเปิดเกม
2. Window size กำหนดขนาดของเกมที่แสดงผลในจอของผู้เล่น (เส้นปะ)



(รูป 010 เส้นปะแสดง Window size)



(รูป 011 Object properties)

Object properties

Object type properties

1. Name ชื่อของ object
2. Plugin ชนิดของ object
3. UID ลำดับประจำตัวของ object นั้น

Common

1. Layer แสดงว่า object นั้นอยู่ layer ไหน
2. Angle แสดงค่ามุมของ object
3. Opacity แสดงค่าความทึบแสง
4. Position แสดงตำแหน่ง
5. Size แสดงขนาด

Instance variables

1. Add/edit เพิ่ม/แก้ไข ค่าตัวแปรของ object นั้น ๆ

Behavior

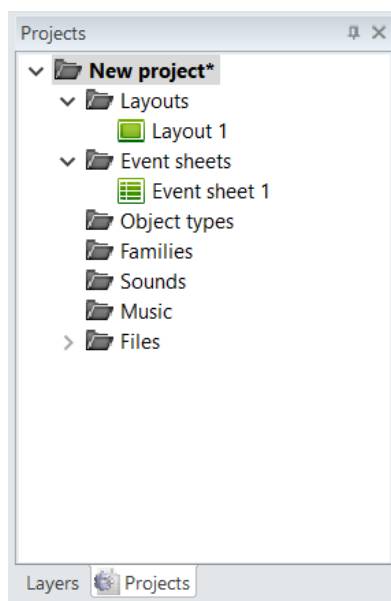
1. Add/edit เพิ่ม/แก้ไข attribute ของ object

Properties

1. Animations แสดง Animation ปัจจุบัน
2. Initial visibility ค่าการมองเห็น
3. Initial animation ตั้งค่า Animation เริ่มต้น
4. Initial frame กำหนด frame เริ่มต้นของ Animation
5. Collisions กำหนดใน object สามารถ มี การชน กับ object อื่นได้

Project bar

แถบแสดงข้อมูล object ของโปรเจค

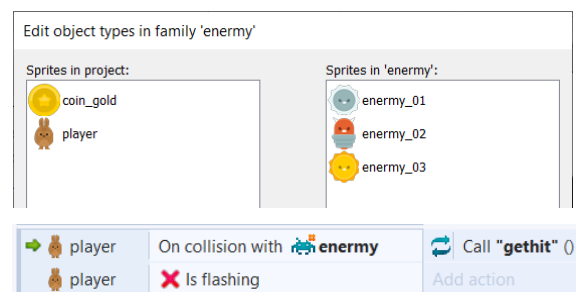


(รูป 012 Project bar)

- | | |
|-----------------|-------------------------------------|
| 1. Layouts | แสดง Layout ทั้งหมดของ project |
| 2. Event sheets | แสดง event sheet ทั้งหมดของ project |
| 3. Object types | แสดง Object ทั้งหมด ของ Project |
| 4. Families | กำหนดกลุ่มให้เรา object |
| 5. Sound | ที่เก็บไฟล์เสียงเอฟเฟค |
| 6. Music | ที่เก็บไฟล์เสียง ประเภเพลงพื้นหลัง |
| 7. Files | นำเข้าไฟล์ หรือ โฟลเดอร์ |



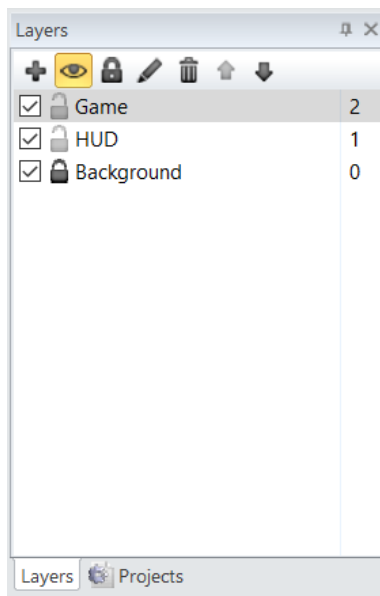
(รูป 013 ก่อนกำหนด Families)



(รูป 014 หลังกำหนด Families)

Layer bar

แถบแสดงข้อมูลของ layer



(รูป 015 layer bar)

Layer เป็นชั้นของข้อมูล(object)ที่ทับซ้อนกันอยู่ใน Layout นั้น ๆ



- | | |
|-------------------------|--------------------------------------|
| 1. Add layer to top | เพิ่ม layer |
| 2. Toggle layer visible | ตั้งค่าให้แสดง / ไม่แสดง |
| 3. Toggle layer locked | ทำให้ไม่สามารถเคลื่อนย้าย object ได้ |
| 4. Rename layer | แก้ไขชื่อ |
| 5. Delete layer | ลบ |
| 6. Move layer up | ย้าย layout ขึ้นด้านบน |
| 7. Move layer down | ย้าย layout ลงด้านล่าง |

Status bar

แถบแสดงข้อมูล ณ ปัจจุบันของไฟล์เกม

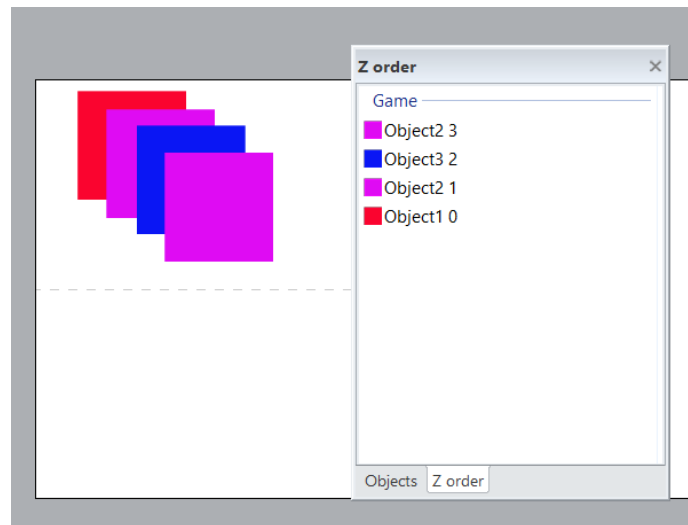


(รูป 016 Status bar)

1. Event แสดงจำนวน event ทั้งหมดใน project นั้น
2. Active layer แสดง layer ที่อยู่ในปัจจุบัน
3. Mouse แสดงพิกัด เมาส์ ณ ปัจจุบัน
4. Zoom แสดงขนาดการซูม

Z order bar

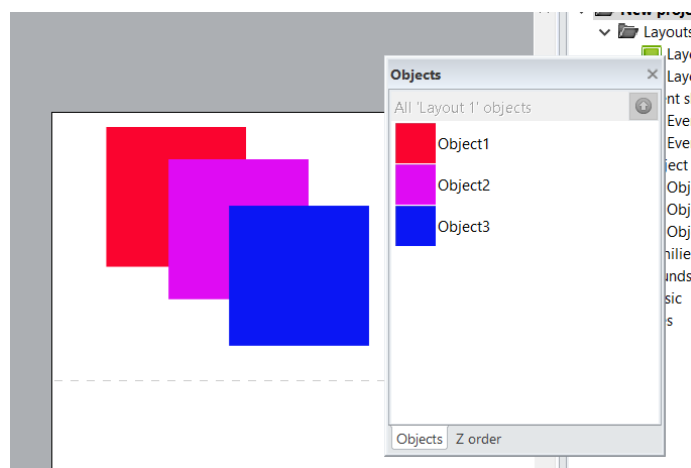
แสดงลำดับของ object ในแกน z (ความลึก)



(รูป 017 Z order bar)

Object bar

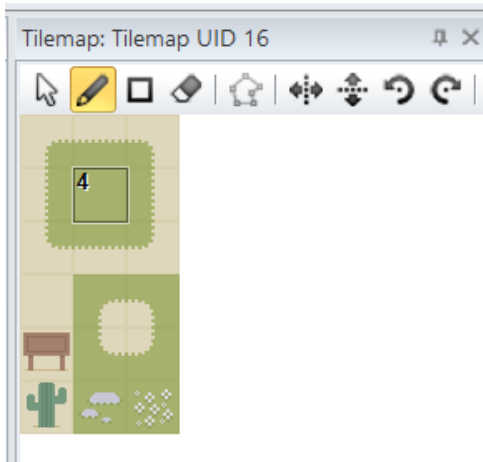
แสดง object ทั้งหมดที่ปรากฏใน layout



(รูป 018 Object bar)

Tilemap Bar

ใช้สำหรับการวาดแผนที่ หรือ ฉากของเกม โดยดึงภาพจาก Tilemap sheet



(รูป 019 Tilemap bar)

1. Normal layout view selection

เปลี่ยนเมาส์เป็นแบบปกติ

2. Pencil tile tool

เปลี่ยนเมาส์เป็นแบบสำหรับสร้าง tile

โดยเลือกจากตัว tilemap ที่ละช่อง

3. Rectangle tile tool

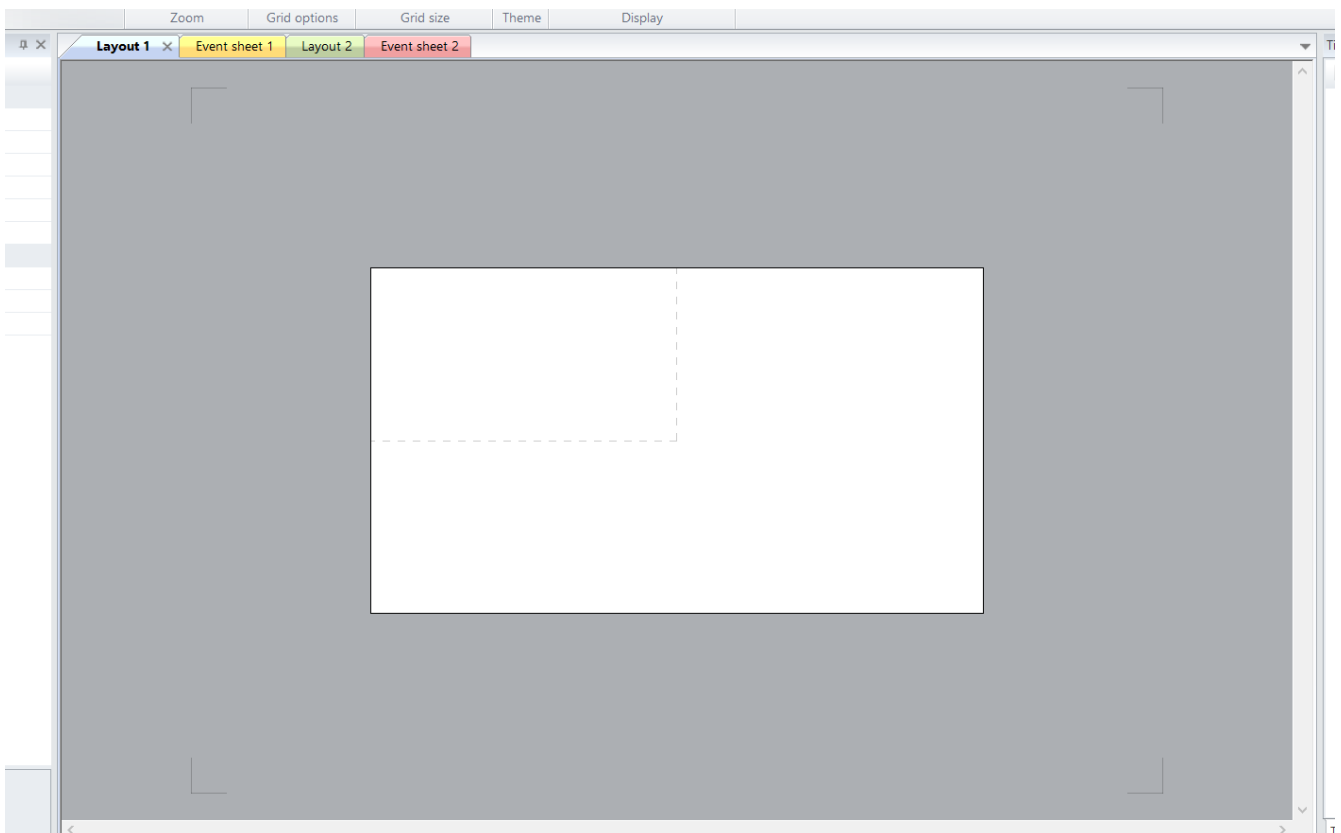
สร้าง tile จากที่เลือก ด้วยการลาก

4. Erase tile tool

ลบ tile ที่สร้างไว้

Workspace

พื้นที่สำหรับจัดวางส่วนต่าง ๆ ของเกม หากคลิกที่พื้นที่สีขาวจะแสดง Properties bar ของ Layout



(รูป 020 Workspace)

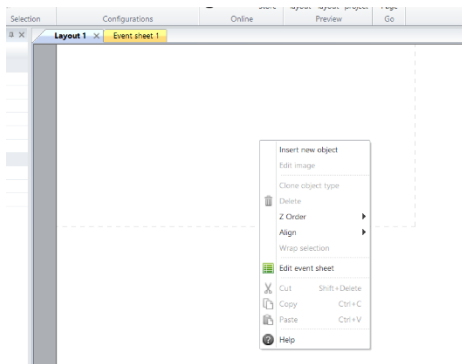
Chapter 2: Object

Object คือ สิ่งที่เป็นตัวแทนของสิ่งต่าง ๆ ที่เราสร้างขึ้นมา เช่น ตัวละคร, ช่องเก็บข้อมูล หรืออุปกรณ์ สำหรับเล่นเกม (คีย์บอร์ด, จอย, เม้าส์) ฯลฯ

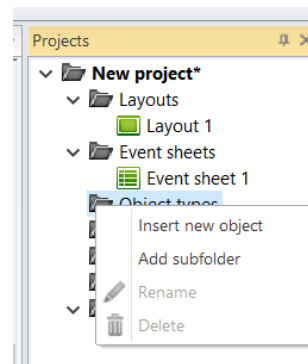
Insert new object

การสร้าง Object ในโปรแกรม Construct2 นั้นจะเรียกว่า การ Insert โดยมีวิธีการดังนี้

1. คลิกขวาที่ workspace เลือก Insert new object หรือ คลิกขวาที่ ไอคอนไฟล์ในแถบ project bar

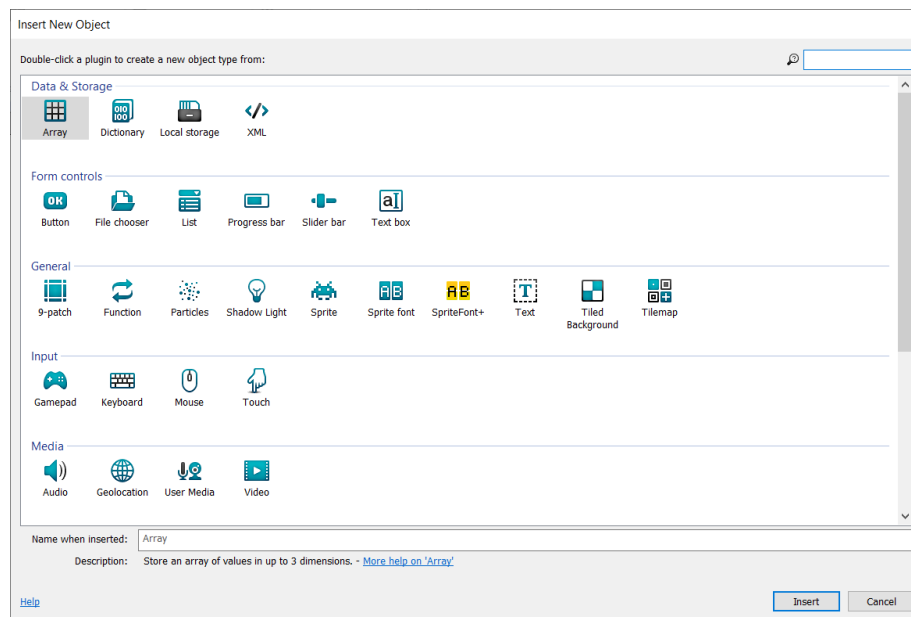


(รูป 021 Insert Object(1))



(รูป 022 Insert Object(2))

2. จะมีหน้าต่างใหม่ขึ้นมาสามารถเลือกประเภทที่เราอยากให้ Object เป็นได้โดยการเลื่อน แถบด้านขวา



(รูป 023 Insert Object(3))

3. คลิกซ้ายเพื่อเลือกประเภทที่เราอยากให้ Object เป็นอะไร
 4. หลังจากนั้นก็ตั้งชื่อให้ Object ในกรอบ สีเหลี่ยมด้านล่าง และ กดปุ่ม Insert เพื่อสร้าง
- คำเตือน :** การตั้งชื่อ Object นั้นควรตั้งให้สอดคล้องกับการใช้งาน หรือ ว่าอธิบายว่าคืออะไร

Object ของ Construct 2

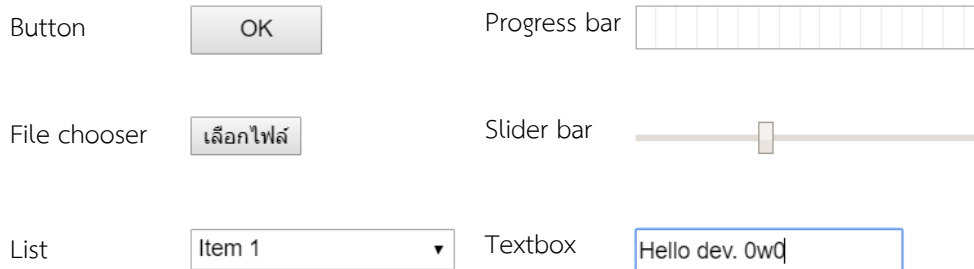
แยกออกเป็น 8 ประเภทดังนี้

1. Data & Storage คือ Object สำหรับเก็บข้อมูลในรูปแบบต่าง ๆ มีทั้งหมด 4 แบบ

1. Array เก็บข้อมูลในรูปแบบ แถว ได้ตั้งแต่แบบ 1 มิติ ถึง แบบ 3 มิติ ใช้สำหรับเก็บค่าตัวแปรต่าง ๆ
2. Dictionary เก็บข้อมูลในรูปแบบ ดิกชันนารี
3. Local storage เก็บข้อมูลแล้วบันทึกข้อมูลลงในเครื่องของผู้ใช้ ใช้สำหรับเก็บข้อมูลที่ต้องการบันทึก เช่น เซฟเกม, ค่าคะแนนสูงสุด เป็นต้น
4. XML เป็นการใช้ข้อมูลจาก ไฟล์เอกสารรูปแบบไฟล์ XML ใช้สำหรับ ดึงข้อมูลไฟล์ XML มาใช้

2. Form controls คือ Object สำหรับให้ผู้เล่นควบคุมสิ่งต่าง ๆ ส่วนใหญ่ใช้ในการตั้งค่าในตัวเกม (ความยาก, ความสวยงามในเกม) มีทั้งหมด 6 แบบ

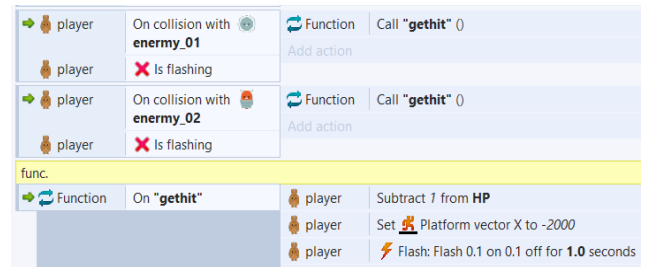
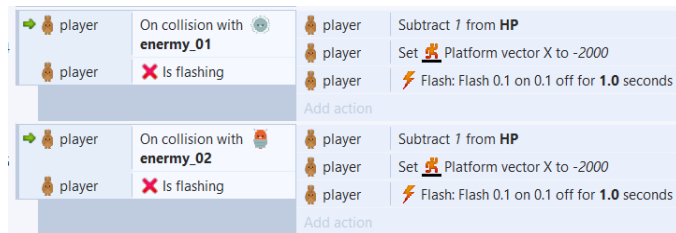
1. Button เป็นปุ่มสำหรับกด หรือ เป็นช่องทำเครื่องหมาย
2. File chooser เป็นโปรแกรมเสริม ที่อนุญาตเลือกไฟล์ในเครื่องของผู้เล่นมาใส่ในเกมได้
3. List เป็นกล่องแบบ list แบบต่าง ๆ
4. Progress bar เป็น object บาร์แสดงผลการทำงานว่าดำเนินไปทำอะไรแล้ว
5. Slider bar เป็น object บาร์เลื่อนซ้ายขวา
6. Textbox เป็น object ที่สร้างช่องเปล่าให้ผู้เล่นได้กรอก



(รูปที่ 024 Object > Form controls)

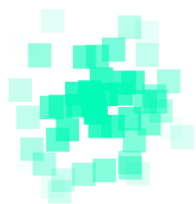
3. General คือ Object หลักที่ใช้ในเกม มีทั้งหมด 9 แบบ

1. 9-patch เป็นเปลี่ยนขนาดรูปโดยหารตัดบริเวณขอบและมุม ใช้สำหรับ Object ที่เปลี่ยนตามขนาดหน้าจอแสดงผลถูกต้อง
2. Function เป็นการเรียกใช้ action เดิมแต่ต่าง event ใช้เพื่อการเขียน action ซ้ำ ในต่าง event

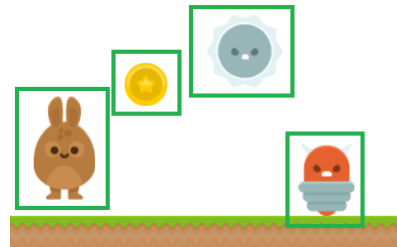


(รูปที่ 025 ก่อนและหลังการใช้ Function)

3. Shadow light สามารถทำให้ object นั้นสร้าง เงาแบบ real-time
4. Particles สำหรับสร้าง เอฟเฟค แบบฝุ่น หรือ ควัน
5. Sprite object ที่จะปรากฏในเกม เช่น ตัวละคร, สิ่งของ เป็นต้น



(รูปที่ 026 Particles)



(รูปที่ 027 Sprite)

6. Sprite Font เป็น object รูปภาพที่แสดงออกเป็นตัวอักษร
7. Text เป็น object สำหรับแสดงตัวอักษรในเกม
8. Tiled Background เป็น object ที่แสดงออกเป็นรูปภาพ ซ้ำ ๆ เป็นรูปแบบ
9. Tilemap เป็นนำรูปที่เป็นแผ่นมาต่อกันจน เป็นแผนที่ในเกม

Tilebackground



Sprite



(รูปที่ 028 Tile background VS Sprite)



(รูปที่ 029 Tilemap)

4. Input คือ การทำให้เกมสามารถรับคำสั่งต่าง ๆ และ ตั้งค่าว่าปุ่มใดทำให้เกิดอะไร ได้ มีทั้งหมด 4 แบบ

1. Gamepad สามารถทำให้รับคำสั่ง จาก จอย
2. Keyboard สามารถทำให้รับคำสั่ง จาก คีย์บอร์ด
3. Mouse สามารถทำให้รับคำสั่ง จาก เมาส์
4. Touch สามารถทำให้รับคำสั่ง จากการ สัมผัส ใช้กับมือถือ

5. Media คือ สื่อต่าง ๆ ที่ทำให้แสดง หรือ รับการทำงานได้ มีทั้งหมด 4 แบบ

1. Audio ใช้เพื่อให้สามารถใส่และใช้เสียงในเกม
2. Geolocation ใช้เพื่อประมาณการที่อยู่ของ ผู้เล่น
3. User Media อนุญาตให้ รับข้อมูลจาก กล้อง webcam หรือ microphone ของผู้เล่นในเกมได้
4. Video ทำให้แสดง Video ในเกมได้

6. Monetisation คือสร้างรายได้ผ่านทางวิธีต่าง ๆ มีทั้งหมด 3 แบบ

1. AdMob ads ทำให้เกม โฆษณา จาก AdMob
2. IAP หรือ In-App Purchases ทำให้เกมสามารถขายของในเกมแก่ผู้เล่น โดยซื้อภายในเกม
3. pubCenter ทำให้เกมโฆษณา จาก Windows Store

7. Platform specific คือการทำให้ ขาย, เล่น, ใช้งานหรือเก็บข้อมูลบน Platform ต่าง ๆ ได้ มีทั้งหมด 10 แบบ

- | | |
|-----------------------|------------------|
| 1. Facebook | 6. Scirra Arcade |
| 2. Game Center บน IOS | 7. Stride |
| 3. Google Play | 8. Twitter |
| 4. Kongregate | 9. Windows Store |
| 5. NW.js | 10. Xbox Live |

8. Web คือการทำให้เกมติดต่อแบบ Web มีทั้งหมด 4 แบบ

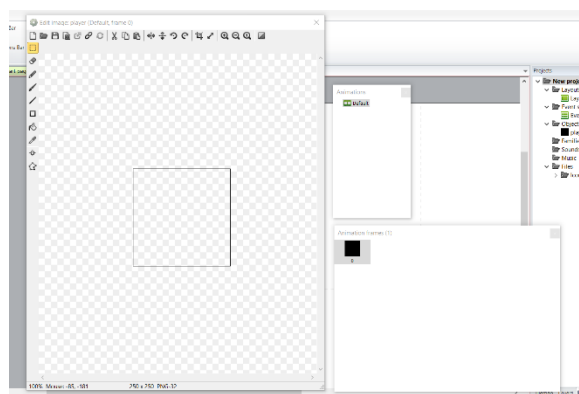
1. AJAX ทำให้สามารถโหว้ระหว่างเกมได้
2. Browser สามารถใช้งานรูปแบบ web ในการรันเกม
3. Multiplayer สามารถทำ real-time online multiplayer ได้
4. WebSocket ใช้รูปแบบการส่งข้อมูล real-time แบบ WebSocket protocol

Sprite

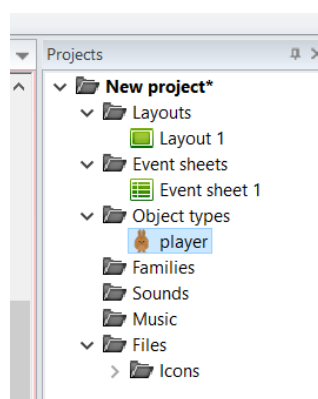
Sprite คือ object ที่จะปรากฏและมีผลในเกม เช่น ตัวละคร, มอนสเตอร์, อาวุธ เป็นต้น
การสร้าง Sprite

1. คลิกขวาในพื้นที่บริเวณตรงกลางหน้าจอ แล้วเลือก Insert new object
2. จะมีหน้าจอใหม่ขึ้นเลื่อนเมาส์ไปคลิกที่ Sprite หลังจากนั้นตั้งชื่อและ กดปุ่ม Insert เพื่อสร้าง / ใส่รูป
3. คลิกที่ workspace จะขึ้นหน้าต่างดังรูป (รูป 029) คลิกที่ไอคอนแฟ้ม และเลือกภาพให้ Sprite และกด enter
4. สามารถลาก Sprite จาก project bar มาเพิ่มใน workspace ได้ (รูป 030)

คำเตือน : การตั้งชื่อ Sprite นั้นควรตั้งให้สอดคล้องกับการใช้งาน หรือ ว่าอธิบายว่าคืออะไร



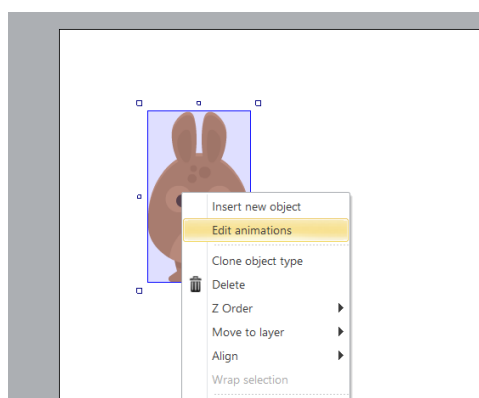
(รูปที่ 030 หน้าต่างวาด, ใส่รูปให้ Sprite)



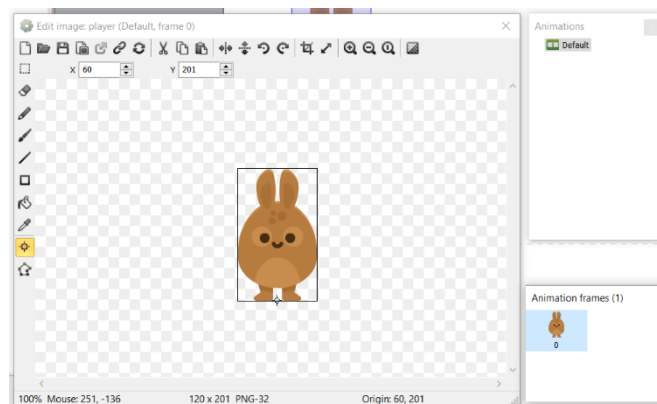
(รูปที่ 031 การแสดง sprite ใน project bar)

การกำหนด Animation ของ sprite

1. ดับเบิลคลิก หรือ คลิกขวาที่ sprite และเลือก edit animation (รูป 031) จะขึ้นหน้าต่างดังนี้ (รูป 032) โดยจะประกอบไปด้วย หน้าต่างตั้งค่ารูปภาพ (edit image) หน้าต่างอนิเมชัน(animation) และ หน้าต่าง อนิเมชันเฟรม (animation frames)

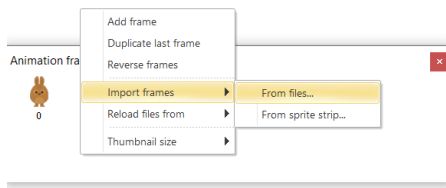


(รูปที่ 032 การเปิดหน้าต่าง การจัดการอนิเมชัน)

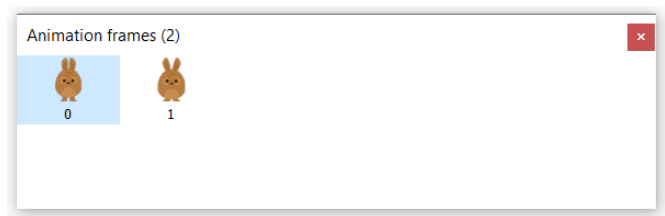


(รูปที่ 033 หน้าต่าง การจัดการอนิเมชัน)

2. คลิกขวาที่ หน้าต่าง animation frame > import frame > from file... > เลือกรูปภาพ > คลิก open (รูป 033) ผลลัพธ์ (รูป 034)

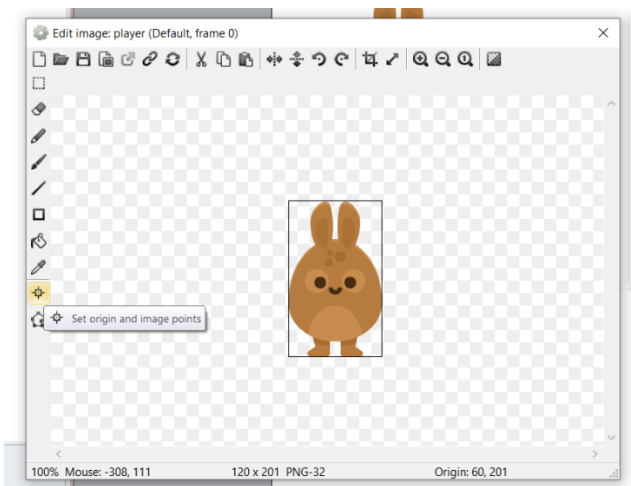


(รูปที่ 034 import frame: step)

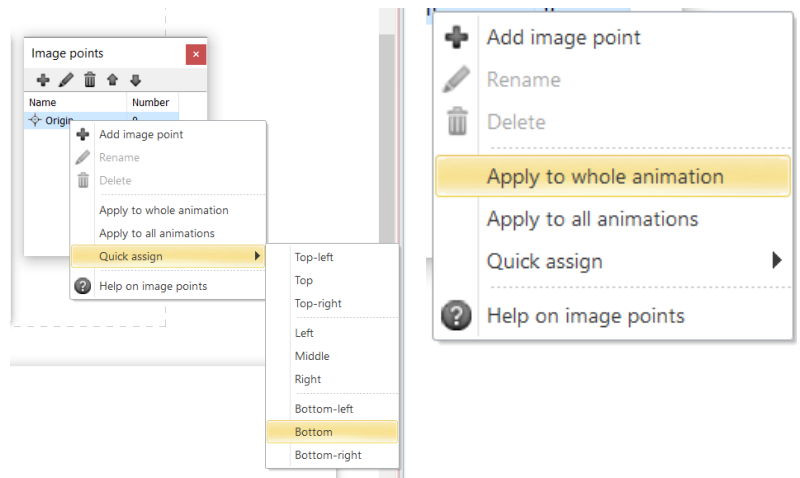


(รูปที่ 035 import frame: result)

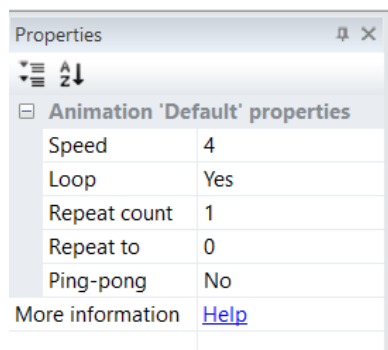
3. ตั้งค่าจุด origin ของ sprite ให้เท่ากัน ตามลำดับ



(รูปที่ 036 set up)



4. เมื่อคลิก อนิเมชัน ใน หน้าต่าง อนิเมชัน มีแถบ Animation properties ขึ้น (รูป 021)

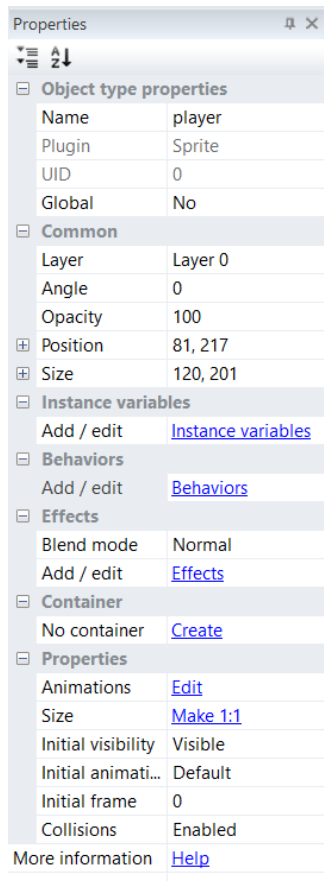


(รูปที่ 037 Animation Properties)

- | | |
|-------------------|--|
| 1. Speed | ความเร็วในการเล่นอนิเมชัน 1 รอบ |
| 2. Loop(No / Yes) | ให้ออนิเมชันนั้นเล่นซ้ำ |
| 3. Repeat count | ตั้งค่าให้ออนิเมชันวนซ้ำตามรอบที่กำหนด |
| | ใช้ได้กรณีที่ไม่ได้ตั้งค่าให้ลูป |
| 4. Repeat to | ตั้งค่าเฟรมเมื่อเล่นอนิเมชันจนจบ |
| 5. Ping-pong | เล่นอนิเมชันไปกลับ เช่น เฟรมที่ 1 2 3 2 1 เป็น |

Sprite Properties

การกำหนดคุณสมบัติให้กับ Sprite



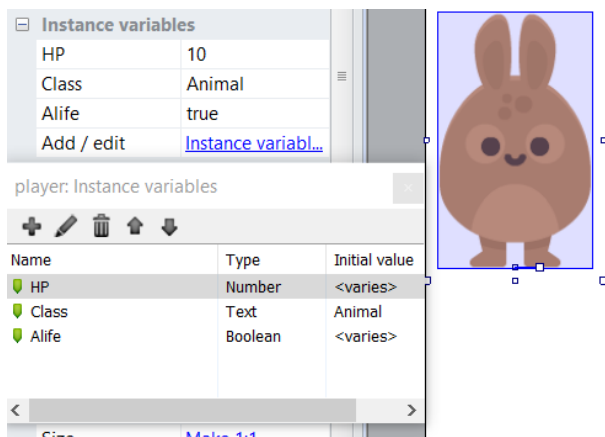
Object type properties

1. Name ชื่อของ Sprite
2. Plugin บอกว่า Object เป็นประเภทอะไร แต่ในที่นี้ คือ Sprite
3. UID ตัวเลขเฉพาะของ Object
4. Global ตั้งค่า Object ว่าจะถูกทำลายหลังจากออก Layout หรือไม่

Common

1. Layer บอกว่า Sprite อยู่ Layer ไหน
2. Angle บอก Sprite นั้นหมุนกี่องศา
3. Opacity บอกความทึบแสงของ Sprite
4. Position บอกว่า Sprite อยู่ตรงไหนตาม แกน x และ แกน y
5. Size บอกว่า Sprite นั้นขนาดเท่าไร มีหน่วยเป็น pixels

(รูปที่ 038 Sprite Properties bar)

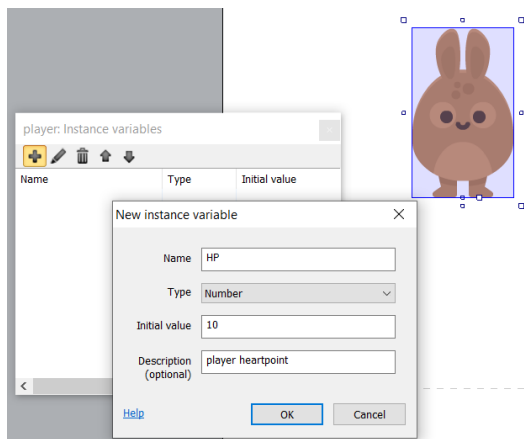


Instance variables คือ ตัวแปรของ Object / Sprite

โดยมีอยู่ Instance variables อยู่ 3 ประเภทคือ

1. Number มีค่าเป็นตัวเลข
2. Boolean เป็นค่าระหว่าง False(เท็จ) กับ True(จริง)
3. Text มีค่าเป็นตัวหนังสือ

(รูปที่ 039 หน้าต่าง Instance variables)



(รูปที่ 040 การสร้าง Instance variables)

โดยวิธีการสร้าง Instance variable ดังนี้

1. คลิกไปที่ Sprite ที่ต้องการคลิกไปที่ Instance variables ที่อยู่ทางด้านซ้ายของจอ
2. หลังจากนั้นจะมีหน้าจอใหม่ขึ้นมา ให้คลิกไปที่ สัญลักษณ์ + เพื่อเป็นการเพิ่ม

Instance variables

3. หลังจากนั้นจะมีหน้าจอใหม่ขึ้นมา ให้เราตั้ง ชื่อ เลือกประเภท Instance variables ค่าที่เราอยากให้เป็น
4. หลังจากกดปุ่ม OK ถือว่าเป็นอันเสร็จ

Text

Text คือ object สำหรับแสดงตัวอักษรในเกม

Properties	
Text	Text
Initial visibility	Visible
Font	Arial(12)
Color	0, 0, 0
Horizontal alignment	Left
Vertical alignment	Top
Hotspot	Top-left
Wrapping	Word
Line height	0
More information	Help

(รูปที่ 041 Text Properties bar)



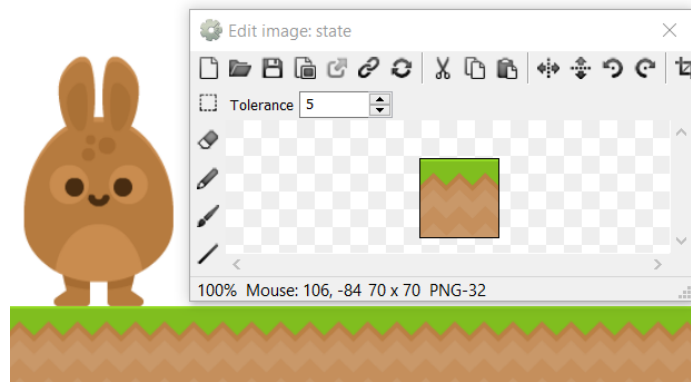
(รูปที่ 042 ตัวอย่าง Text)

Text Properties

1. Text ตัวอักษรที่จะโชว์บนจอ
2. Initial visibility ตั้งค่าว่าให้มาเห็นในเกมหรือไม่
3. Font รูปแบบ Font ที่ใช้
4. Color สีของตัวอักษร
5. Horizontal alignment ตำแหน่งตัวอักษรในแนวนอน
6. Vertical alignment ตำแหน่งตัวอักษรในแนวตั้ง
7. Hotspot เลือกจุดหมุนให้ Objects
8. Wrapping ล้อมข้อความ ด้วยคำหรือ ตัวอักษร
9. Line height ค่าความห่างแต่ละบรรทัด

Tile background

Tile background คือ object ที่แสดงออกเป็นรูปภาพ ซ้ำ ๆ เป็นรูปแบบ Tile background



(รูปที่ 043 ตัวอย่าง Tile background)

Tile map

Tile map คือ เป็นนำรูปที่เป็นแผ่นมาต่อกันจน เป็นแผนที่
ในเกม จะมีหน้าต่างใหม่ขึ้นมาหลังจากทำการ Insert tilemap
เพื่อให้เราดิงภาพที่ต้องการมาเพื่อวาด tilemap



(รูปที่ 044 ตัวอย่าง Tilemap)

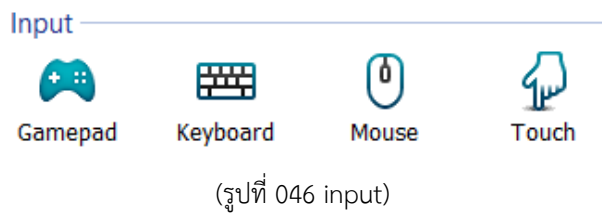
Properties	
Image	Edit
Initial visibility	Visible
Tile width	32
Tile height	32
Tile X offset	0
Tile Y offset	0
Tile X spacing	0
Tile Y spacing	0
Seamless mode	On
More information	Help

(รูปที่ 045 Tilemap Properties)

Tilemap Properties

1. Image รูปภาพสำหรับใช้ใน Tilemap
2. Initial visibility ตั้งค่าว่าให้มาเห็นในเกมหรือไม่
3. Tile width ความกว้างของ Tile
4. Tile height ความสูงของ Tile
5. Tile x offset รอยห่างแนวนอน ของ Tile
6. Tile y offset รอยห่างแนวตั้ง ของ Tile
7. Tile x spacing ระยะห่างแนวนอน ระหว่าง Tile
8. Tile y spacing ระยะห่างแนวตั้ง ระหว่าง Tile
9. Seamless mode ทำให้ภาพโรยต่อ

Input



Input คือ การทำให้เกมสามารถรับคำสั่งผ่านตัวรับต่าง ๆ โดยจะมีให้เลือก 4 ชนิด

1. จอยคอนโทรลเลอร์
2. คีย์บอร์ด
3. เมาส์
4. สัมผัส (มือถือ)

Audio

Insert เพื่อให้ โปรแกรมสามารถ อ่านไฟล์เสียงที่นำเข้ามาได้

Properties

1. Timescale audio เปลี่ยนเวลาในการเล่น audio
2. Save/load เล่นย้อนหลังเมื่อมีการโหลด และ เซฟ
3. Play in background เล่นเพลงหลัง background

Positioned Audio

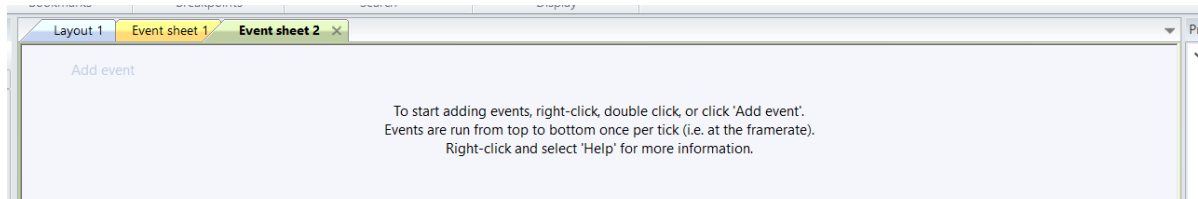
1. Panning model รูปแบบประมวลผลเสียง
2. Distance model รูปแบบเสียงที่ลดลงจากระยะห่าง
3. Listener Z height ระดับเสียงความดังตามลำดับ layout
4. Reference distance เสียงโดยวัดจากระยะเป็น pixels
5. Maximum distance ระยะเสียงที่จะไม่ลดลง
6. Roll-off factor เสียงที่ลดลงเร็วๆ แค่เมื่อห่างจากแหล่ง

Object type properties	
Name	Audio
Plugin	Audio
Properties	
Timescale audio	Off
Save/load	All
Play in background	No
Positioned audio	
Panning model	HRTF
Distance model	Inverse
Listener Z height	600
Reference distance	600
Maximum distance	10000
Roll-off factor	1
More information	Help

(รูปที่ 047 audio properties)

Chapter 3: Event sheets

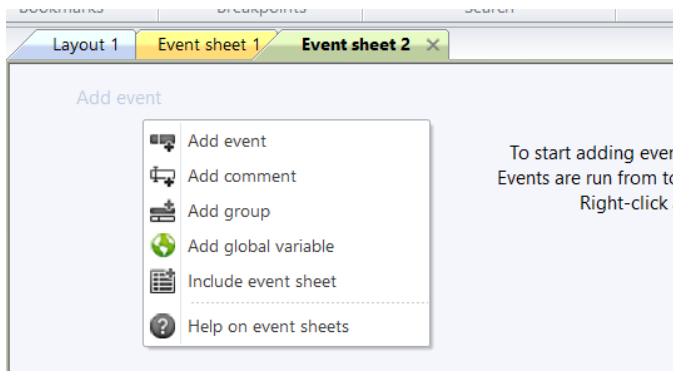
Event Sheet คือส่วนสำหรับการตั้งเงื่อนไขและความเป็นไปของเกม เป็นส่วนสำคัญที่ทำให้เกมมีความสมบูรณ์



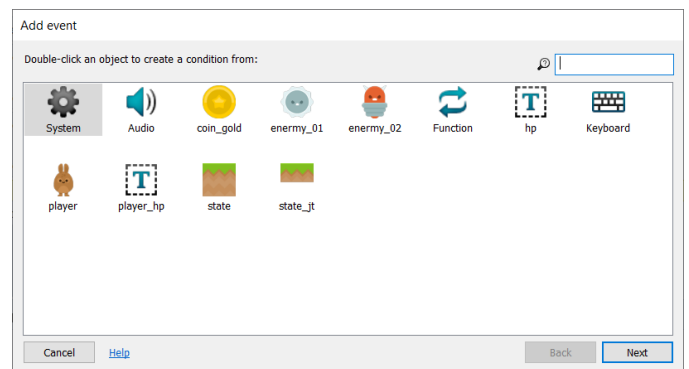
(รูปที่ 048 หน้าต่าง event sheet)

การสร้าง Event

1. คลิกขวาในหน้าของ event sheet หรือ คลิกที่ add event จะมีหน้าต่างขึ้น
2. เลือก Object เพื่อตั้งเงื่อนไข



(รูปที่ 049 add event)



(รูปที่ 050 หน้าต่าง add event)

สามารถแบ่งชนิดของ Event ได้ดังนี้

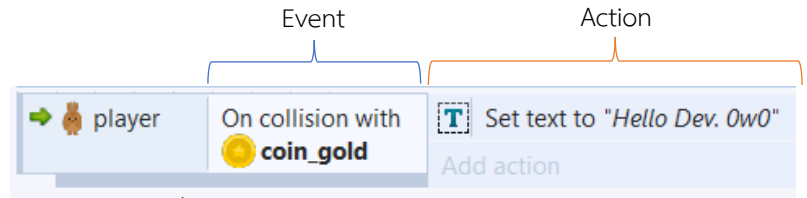
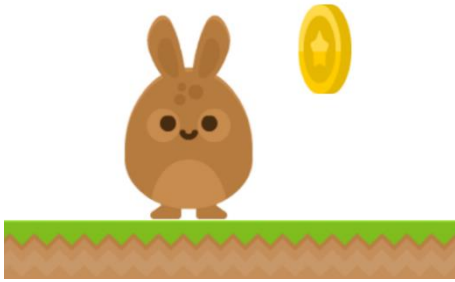
Basic Event
System Event
Sprite Event
Text Event
Input Event
Audio Event
Behavior Event

Event / Action

ใน Event แต่ละชนิดจะแบ่งเป็น 2 ประเภท ตามการทำงานคือ

1. Event จะทำงานเหมือน IF หรือ การกำหนดเงื่อนไข
2. Action จะทำงานเมื่อ มีเหตุการณ์เกิดขึ้นโดยเหตุการณ์นั้นต้องมีเงื่อนไขตรงกับ Event ของ Action นั้นเท่านั้น

ตัวอย่าง



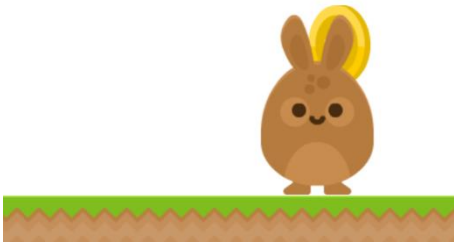
(รูปที่ 051 ตัวอย่าง event (1) event & action)

คำอธิบาย

Event: ถ้า player ชนกับ coin_gold(เหรียญทอง)

Action: ให้ ค่าตัวอักษรของ text เป็น “Hello Dev. Ow0”

Hello Dev. Ow0



(รูปที่ 052 ตัวอย่าง event (2) ก่อนและหลังเริ่ม event)

Expressions: Operators(ตัวดำเนินการ)

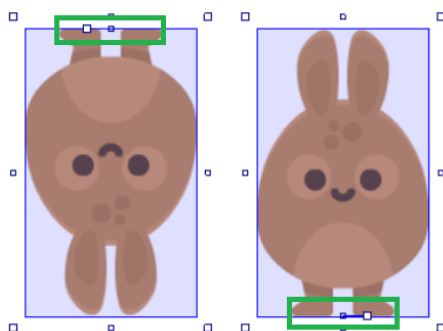
+	การบวก	*	การคูณ
-	การลบ	/	การหาร
% (modulo)			การมอด(หารเอาเศษ)
^ (raise to power)			การยกกำลัง
& (build strings)			การเชื่อมต่อตัวอักษร กับตัวแปรชนิดอื่น ๆ

Basic

เป็น event ที่ใช้ได้กับ Object เกือบทุกประเภท

แบ่งเป็น 9 ประเภทหลัก คือ Angle, Appearance, Instance variable, Mis, Size & Position, Web

Angle ใช้เช็คเงื่อนไขที่สัมพันธ์กับมุมมอง



(รูปที่ 053 origin ของ object)

คำอธิบาย

Object ด้านซ้ายมีมุมอยู่ที่ 180 องศา

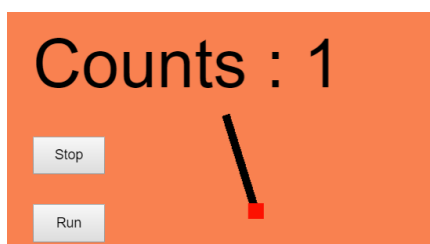
Object ด้านขวามีมุมอยู่ที่ 0 องศา

Event

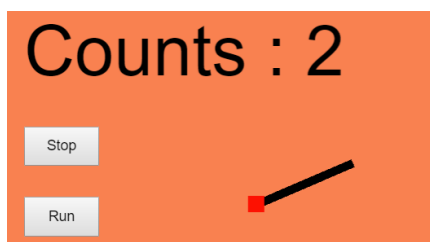
1. Is between angles ใช้ตรวจสอบว่า object มีองศาในช่วงที่กำหนดหรือไม่(อิงจาก origin)
2. Is clockwise from ใช้ตรวจสอบว่า object นั้นหมุนตามเข็มนาฬิกาจากมุม ที่กำหนดหรือไม่
3. Is within angle ใช้เช็คการหมุนของ object ว่าห่างจากมุมที่กำหนดเกินระยะที่ต้องการหรือไม่

Action

1. Rotate clockwise สั่งให้ object นั้น หมุนตามเข็มนาฬิกา โดยอิง จาก origin
2. Rotate counter-clockwise สั่งให้ object นั้น หมุนทวนเข็มนาฬิกา
3. Rotate toward angle สั่งให้ object นั้นหมุนตามองศาและความเร็วที่กำหนดให้
4. Rotate toward position สั่งให้ object นั้นหมุนไปตามตำแหน่งที่กำหนด
5. Set angle กำหนดค่ามุมให้กับ object



(รูปที่ 054 ตัวอย่าง expression &(1))



(รูปที่ 055 ตัวอย่าง expression &(2))

คำอธิบาย (รูป 54 – 56)

Clock_ned มี Behavior เป็น rotate (หมุนตามเข็ม)

Event ทุก ๆ 1/60 วินาที

Action ตั้งค่าตัว อักษร ของText

Event ทุกครั้งที่มันหมุนครบรอบ (360 องศา)

Action ให้เพิ่มค่าของตัวแปร counts 1

System	Every tick	Set text to "Counts : " & clock_ned.counts
		Add action
clock_ned	Is clockwise from 360	Add 1 to counts
		Add action
System	Trigger once	

(รูปที่ 056 ตัวอย่าง expression &(3))

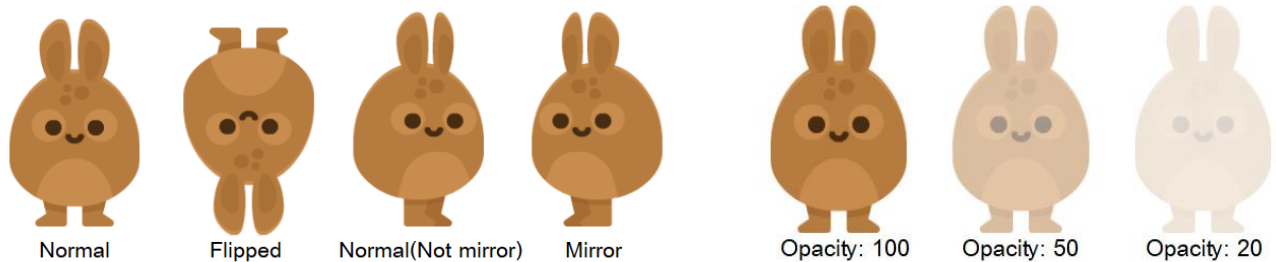
Appearance ใช้เช็คลักษณะรูปร่างของ object

Event

1. Compare opacity เช็คความทึบแสงของ object ในช่วง 0(โปร่งใส) ถึง 100(ทึบแสง)
2. Is flipped เช็คค่า object นั้น flip (กลับด้าน) บน/ล่างหรือไม่
3. Is mirrored เช็คค่า object นั้น Mirror (กลับด้าน) ซ้าย/ขวาหรือไม่
4. Is visible เช็คค่า Object มีการตั้งค่าการมองเห็นหรือไม่

Action

1. Set flipped กำหนดให้ object นั้น flipped / not - flipped
2. Set mirrored กำหนดให้ object นั้น mirrored / not - mirrored
3. Set opacity กำหนดค่าความทึบแสงของ object
4. Set visible กำหนดให้ object นั้นมองเห็นได้ / มองเห็นไม่ได้



(รูปที่ 057 ตัวอย่าง appearance(1))

(รูปที่ 058 ตัวอย่าง appearance(2))

Instance variables การจัดการกับค่าตัวแปร

Event

1. Compare instance variable นำค่าตัวแปรของ object มาเปรียบเทียบ
2. Is Boolean instance variable set เช็คค่าความจริงของตัวแปร

Action

1. Add to เพิ่มค่าให้กับตัวแปรที่เลือกตามจำนวนที่กำหนด
2. Set boolean กำหนดค่าความจริงให้กับตัวแปรชนิด boolean
3. Set value กำหนดค่าให้กับตัวแปร text และ number
4. Subtract from ลดค่าจากตัวแปรตามจำนวนที่กำหนด
5. Toggle boolean เช็คค่าความจริงของตัวแปรชนิด boolean

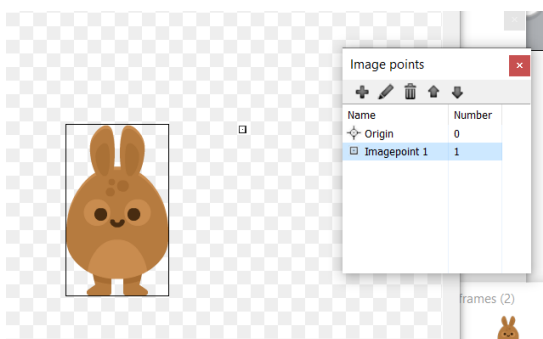
Miscs อื่น ๆ

Event

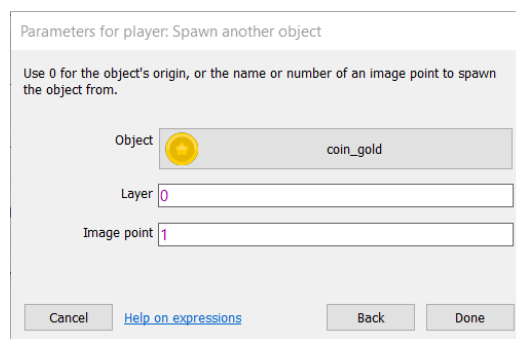
- | | |
|----------------------------|---|
| 1. On created | เมื่อ object นั้นถูกสร้างขึ้น |
| 2. On destroyed | เมื่อ object นั้นถูกทำลาย |
| 3. Pick by unique ID (UID) | ดึงค่า UID: Unique ID ของ object ที่เลือกมาเก็บในตัวแปร UID ของ object นั้น ๆ |

Action

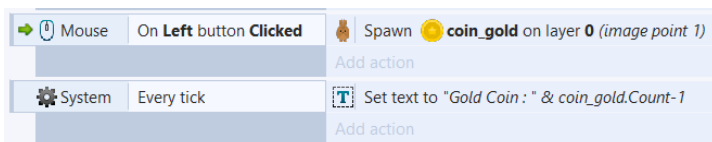
- | | |
|---------------------------|--|
| 1. Destroy | ทำลาย object (ลบ object ออกจากหน้าจอ) |
| 2. Set collisions enabled | กำหนดให้ object สามารถมี event จากชนกับ object อื่นได้ |
| 3. Set from JSON | กำหนด สถานะของ object จากไฟล์ json |
| 4. Spawn another object | กำหนดให้สร้าง object ขึ้นมาจากรูปภาพที่กำหนด |



(รูปที่ 059 ตัวอย่าง Spawn object (1))

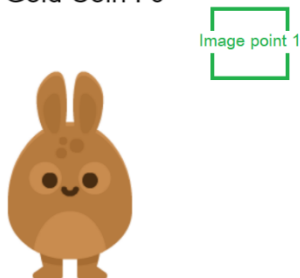


(รูปที่ 060 ตัวอย่าง Spawn object (2))



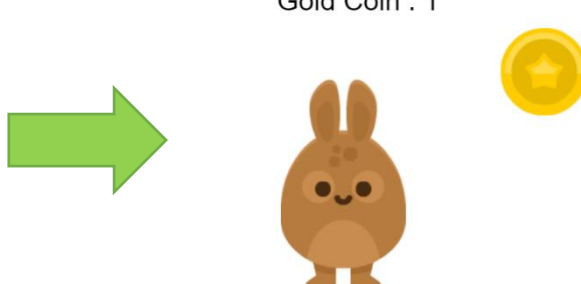
(รูปที่ 061 ตัวอย่าง Spawn object (3))

Gold Coin : 0



(รูปที่ 062 ตัวอย่าง Spawn object (4))

Gold Coin : 1



(รูปที่ 063 ตัวอย่าง Spawn object (5))

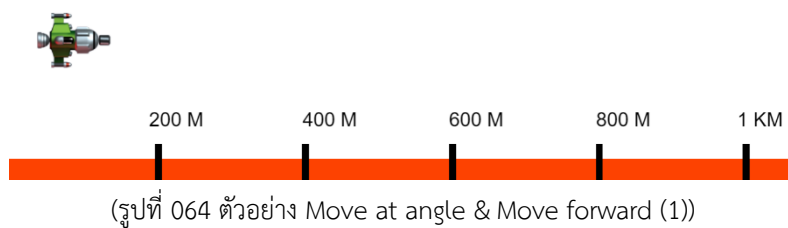
Size & Position




Event

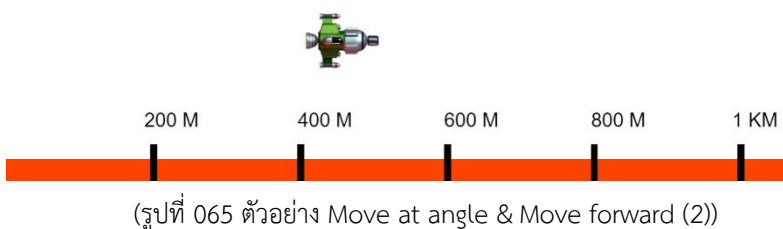
- | | |
|----------------------|--|
| 1. Compare height | เปรียบเทียบความสูงของ object |
| 2. Compare width | เปรียบเทียบความกว้าง object |
| 3. Compare X | เปรียบเทียบตำแหน่งของ object ในแกน x (แนวนอน) |
| 4. Compare Y | เปรียบเทียบตำแหน่งของ object ในแกน y (แนวตั้ง) |
| 5. Is on-screen | เช็คว่ายู่ object ในหน้าจอหรือไม่ |
| 6. Is outside layout | เช็คว่ายู่ object นอกจอหรือไม่ |

Action

- | | |
|-----------------------------------|---|
| 1. Move at angle | กำหนดให้ object เคลื่อนที่ไปในมุมความระยะทางที่กำหนด |
| 2. Move forward | กำหนดให้ object เคลื่อนที่ตามระยะทางที่กำหนด (มุมในการเคลื่อนที่จะขึ้นกับ origin) |
| 3. Set height | กำหนดความสูงของ object |
| 4. Set position | กำหนดตำแหน่ง x และ y ของ object |
| 5. Set position to another object | กำหนดตำแหน่งโดยอิงจาก object อื่นที่เลือกไว้ |
| 6. Set scale | กำหนดสเกลของ object |
| 7. Set size | กำหนดขนาดของ object |
| 8. Set width | กำหนดความกว้างของ object |
| 9. Set X | กำหนดตำแหน่งของ object ในแนวแกน x |
| 10. Set Y | กำหนดตำแหน่งของ object ในแนวแกน y |



Solution A			
	Every tick		Move 1 pixels at angle 0
		Add action	
Solution B			
	Every tick		Move forward 1 pixels
		Add action	



(รูปที่ 066 ตัวอย่าง Move at angle & Move forward (3))

#คำอธิบาย

ในกรณีนี้ Move at angle และ Move forward จะมีผลลัพธ์เหมือนกัน

Web

Event

1. On image URL loaded เช็คเงื่อนไขเมื่อ Object มีการโหลดรูปภาพจาก URL เสร็จสิ้น

Action

1. Load image from URL ดาวน์โหลด รูปภาพจาก URL

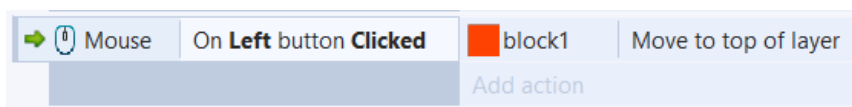
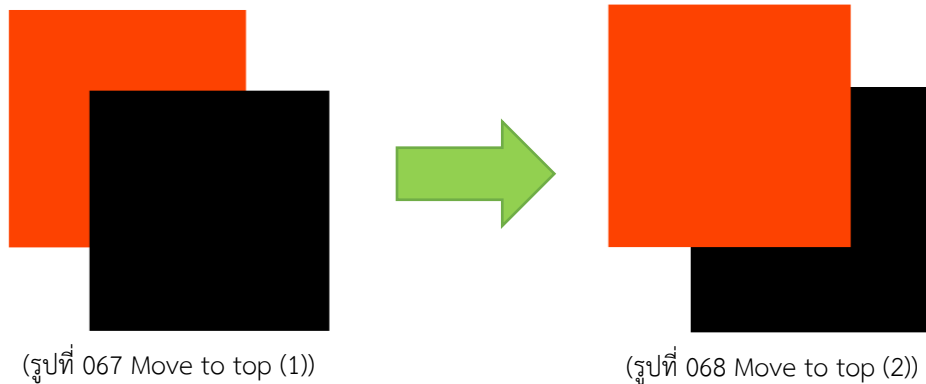
Z Order

Event

1. Is on layer เช็ค ว่า object นั้นอยู่ layer ไหน

Action

1. Move to bottom ย้าย object ไปใต้สุดของแกน z ของ layer ปัจจุบัน
2. Move to layer ย้าย object ไปยัง layer ที่เลือก
3. Move to object ย้าย object ไปอยู่บน/ล่าง object อื่นที่เลือก
4. Move to top ย้าย object ไปบนสุดของแกน z ของ layer ปัจจุบัน



(รูปที่ 069 Move to top (3))

Sprite

Event Animations

- | | |
|---------------------|--|
| 1. Compare frame | เช็คเฟรมอนิเมชันของ sprite |
| 2. Compare speed | เช็คความเร็วอนิเมชันของ sprite |
| 3. Is playing | เช็คว่ sprite นั้นกำลังเล่นอนิเมชันที่เลือกอยู่หรือไม่ |
| 4. On any finished | เมื่อ sprite เล่น animation ใด ๆ จบ |
| 5. On finished | เมื่อ sprite เล่น animation ที่กำหนดจบ |
| 6. On frame changed | เมื่อ animation ของ sprite นั้นมีการเปลี่ยนเฟรม |
-
- | | |
|------------------------|--------------------------------------|
| 1. Set animation | กำหนดให้ sprite เล่นอนิเมชันที่เลือก |
| 2. Set frame | กำหนดเฟรมของ sprite |
| 3. Set repeat-to frame | กำหนด repeat-to |
| 4. Set speed | กำหนด ความเร็วของ animation |
| 5. Start | เริ่มเล่น animation |
| 6. Stop | หยุดเล่น animation |

Collisions

- | | |
|-------------------------------------|---|
| 1. Collisions enable | กำหนดค่าของสถานะ Collisions |
| 2. Is overlapping another object | เมื่อมีการทับซ้อนกับ object อื่น |
| 3. Is overlapping at offset | เมื่อมีการทับซ้อนกับ object อื่นในตำแหน่งที่กำหนด |
| 4. On collision with another object | เมื่อชนกับ Object อื่นที่เลือก |



(รูปที่ 070 Sprite Event)

System

เป็น event ที่ระบบสามารถสร้างได้

General

Event

- | | |
|-----------------------|---|
| 1. Compare two values | นำตัวแปรสองตัวมาเปรียบเทียบกัน |
| 2. Every tick | จะทำงานในทุก ๆ tick (1 / 60 วินาที) |
| 3. Is between values | นำตัวแปร มาเทียบกันในรูปแบบ จำนวน (1) <= ตัวแปร <= จำนวน (2) |
| 4. Is group active | ใช้เช็คให้เกิด event เมื่อกลุ่ม event ที่เลือกมีการทำงาน |
| 5. Is number NaN | เช็คค่าของตัวแปรเมื่อ เกิดค่าที่ผิดพลาดทางคณิตศาสตร์ เช่น 1/0 |
| 6. Is value type | เช็คชนิดค่าของตัวแปร ตัวเลข / ตัวอักษร |
| 7. Object UID exists | เช็คค่า object นั้นมี UID ที่เลือกอยู่หรือไม่ |
| 8. Test regex | ใช้เพื่อค้นหา, แก้ไข และจัดการข้อความ |

Action

- | | |
|-------------------------------|--|
| 1. Create object | ให้ระบบสร้าง object ที่เลือกขึ้นมา |
| 2. Go to layout | ย้ายไปยัง layout ที่เลือก |
| 3. Go to layout (by name) | ย้ายไปยัง layout ที่เลือกด้วยชื่อ |
| 4. Go to next previous layout | ย้ายไปยัง layout ถัดไป/ก่อนหน้า (อิงลำดับในproject bar) |
| 5. Restart layout | เริ่มต้น layout เดิมใหม่ |
| 6. Set group active | กำหนดกลุ่มของ event ให้ ทำงาน / ไม่ทำงาน / เปรียบเทียบ |
| 7. Sort Z order | เรียงลำดับในแกน z ของ object โดยอิงจาก ค่าตัวแปรที่เลือก |
| 8. Stop loop | หยุดการทำงานของลูป |

Global & local variable

Event

- | | |
|---------------------|--|
| 1. Compare variable | นำค่าตัวแปรที่เลือกมาเปรียบเทียบกับค่าคงที่ หรือตัวแปรอื่น |
|---------------------|--|

Action

- | | |
|--------------------------|---|
| 1. Add to | เพิ่มค่าให้กับ Global variable ที่เลือก ตามที่กำหนด |
| 2. Reset global variable | รีเซ็ตค่าของ Global variable ทั้งหมด |
| 3. Set value | กำหนดค่าให้กับ Global variable ที่เลือก |
| 4. Subtract from | ลบค่าจากกับ Global variable ที่เลือก ตามที่กำหนด |

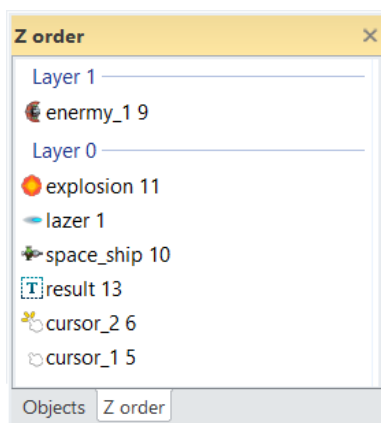
Layer & Layout

Event

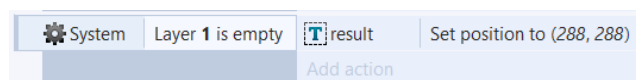
- | | |
|--------------------------|--|
| 1. Compare opacity | นำค่าความทึบแสงมาเปรียบเทียบ |
| 2. Effects are supported | เช็คว่ามีเปิดใช้งาน effect หรือไม่ |
| 3. Layer is empty | เช็คว่างเปล่าหรือไม่ |
| 4. Layer is visible | เช็คว่างเปล่าหรือไม่ |
| 5. On canvas snapshot | ทำงานเมื่อ Snapshot canvas พร้อมใช้งาน |

Action

- | | |
|---------------------------------|--|
| 1. Reset persisted objects | รีเซ็ต behavior persist ของทุก object |
| 2. Set layer angle | กำหนด องศาให้กับ layer ที่เลือก |
| 3. Set layer background color | กำหนด สีพื้นหลังให้กับ layer |
| 4. Set layer effect enabled | กำหนดใน effect ที่ตั้งค่าไว้ทำงาน/ไม่ทำงาน |
| 5. Set layer effect parameter | กำหนดค่าตัวแปรของ effect นั้น |
| 6. Set layer opacity | กำหนดค่าความทึบแสงของ layer นั้น |
| 7 Set layer parallax | กำหนด parallax ของ layer นั้น |
| 8. Set layer scale | กำหนด scale ของ layer |
| 9. Set layer scale rate | กำหนด scale rate ของ layer |
| 10. Set layer visible | กำหนดให้ layer ที่เลือกนั้น มองเห็น ได้/ไม่ได้ |
| 11. Set layout angle | กำหนดองศาให้กับ layout |
| 12. Set layout effect enable | กำหนดใน effect ที่ตั้งค่าไว้ทำงาน/ไม่ทำงาน |
| 13. Set layout effect parameter | กำหนดค่าตัวแปรของ effect นั้น |
| 14. Set layout scale | กำหนด scale ของ layout |



State 1 Clear !



(รูปที่ 071 ตัวอย่าง Layer is empty)

Loops

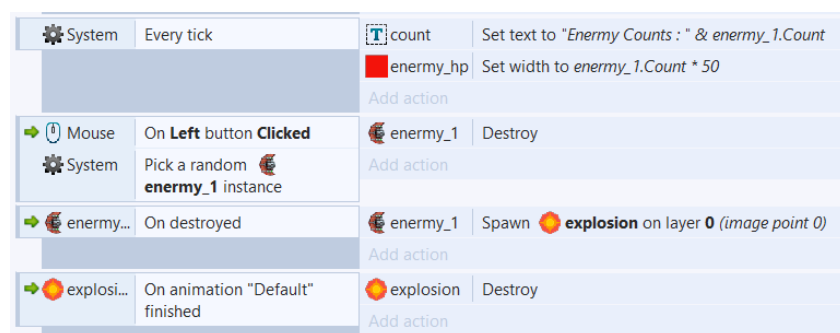
Event

1. For กำหนดการรูป(วนทำงาน) ตามจำนวนรอบที่กำหนด
2. Repeat ทำซ้ำตามรอบที่กำหนด
3. While ทำซ้ำจนกระทั่งเงื่อนไขเป็นเท็จ

Pick instances

Event

1. Pick all ใช้เช็คจำนวนของ object ที่เลือก
2. Pick by comparison นำค่าตัวแปรของ object ที่เลือกมาเปรียบเทียบ
3. Pick by evaluate นำค่าตัวแปรของ object ที่เลือกมาเปรียบเทียบ
4. Pick nth instance เลือกให้ object ทำงานแต่ตัวที่เลือก
5. Pick overlapping point กำหนดให้ ทำงานเมื่อ object ที่เลือกไปทับกับตำแหน่งที่กำหนด
6. Pick random instance สุ่มเลือกให้ object ทำงาน



(รูปที่ 072 ตัวอย่าง Pick instances(1))

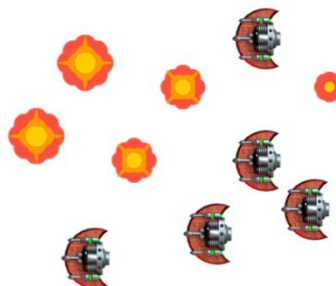
Enemy Counts : 10

Enemy HP



Enemy Counts : 5

Enemy HP



(รูปที่ 073 ตัวอย่าง Pick instances(2))

(รูปที่ 074 ตัวอย่าง Pick instances(3))

Save & Load

Event

- | | |
|---------------------|------------------------------------|
| 1. On load complete | ทำงานเมื่อคำสั่ง load ทำงานสำเร็จ |
| 2. On load failed | ทำงานเมื่อคำสั่ง load ทำงานล้มเหลว |
| 3. On save complete | ทำงานเมื่อคำสั่ง save ทำงานสำเร็จ |
| 4. On save failed | ทำงานเมื่อคำสั่ง save ทำงานสำเร็จ |

Action

- | | |
|-------------------|---|
| 1. Load | โหลดข้อมูลจากไฟล์เกมของเครื่องที่ใช้งาน |
| 2. Load from JSON | โหลดข้อมูลจากไฟล์ json |
| 3. Save | บันทึกข้อมูลเกม ณ ปัจจุบัน |

Special conditions

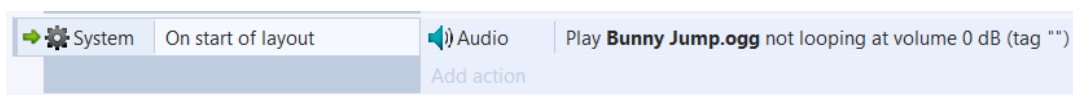
Event

- | | |
|----------------------------|---|
| 1. Else | ทำงานเมื่อเงื่อนไข event ด้านบนเป็นเท็จ |
| 2. Is preview | ทำงานเมื่อมีการรันไฟล์ในโหมด 프리วิว (Run, F5) |
| 3. Is on mobile device | ทำงานเมื่อรันไฟล์บนอุปกรณ์มือถือ |
| 4. Is on platform | ทำงานเมื่อรันไฟล์บนอุปกรณ์ platform |
| 5. Trigger once while true | ทำงานเพียงหนึ่งครั้งเมื่อเงื่อนไขนั้นเป็นจริง |

Start & end

Event

- | | |
|------------------------------|--------------------------------------|
| 1. On end of layout | ทำงานเมื่อมีการเปลี่ยน layout |
| 2. On loader layout complete | ทำงานเมื่อการโหลด layout ถัดไปสำเร็จ |
| 3. On start of layout | ทำงานเมื่อเริ่ม layout |



(รูปที่ 075 ตัวอย่าง On start of layout)

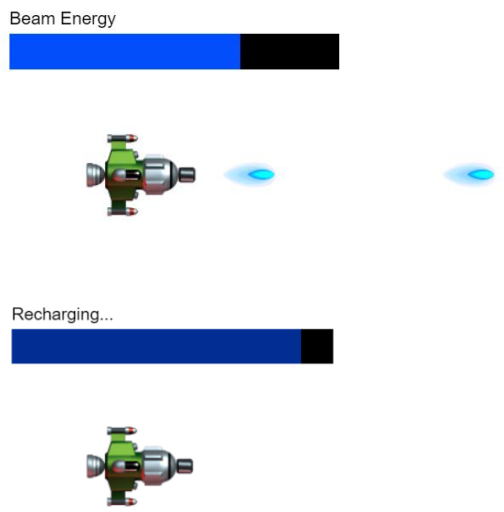
Time

Event

1. Compare time นำเวลามาเปรียบเทียบกับเงื่อนไข (เริ่มนับตั้งแต่เริ่ม layout)
2. Every X seconds ทำงานทุก ๆ วินาที ตามที่กำหนด

Action

1. Set time scale กำหนด time scale ของเกม
2. Wait ตั้งเวลา รอ ในการทำขั้นตอนต่อไป



System	Every 0.5 seconds	space_ship	Spawn laser on layer 0 (image point 1)
space_s...	energy > 0	space_ship	Subtract 1 from energy
		Add action	
System	Every tick	ss_energy	Set width to space_ship.energy * 30
		ss_rechar...	Set width to space_ship.recharge * 30
		Add action	
space_s...	energy = 0	beam_text	Set text to "Recharging..."
		Add action	
	Every 0.3 seconds	space_ship	Add 1 to recharge
		Add action	
	recharge < 10		
	recharge = 10	Audio	Play "Recharge_effect" not looping from Sounds at 0 dB (tag "")
	Trigger once	System	Wait 0.3 seconds
		space_ship	Set energy to 10
		beam_text	Set text to "Beam Energy"
		space_ship	Set recharge to 0

(รูปที่ 076 ตัวอย่าง Time Event)

System Action

Display

1. Set canvas size กำหนดขนาดของ canvas
2. Set fullscreen scaling กำหนดสเกลของภาพระดับ fullscreen
3. Set pixel rounding กำหนด enable / disable ของ pixel rounding

Scrolling

1. Scroll to object ย้ายมูมกลิ้งไปยัง object ที่เลือก
2. Scroll to position ย้ายมูมกลิ้งไปยังตำแหน่งที่เลือก
3. Scroll to X ย้ายไปยังพิกัดในแนวแกน x ที่เลือก
4. Scroll to Y ย้ายไปยังพิกัดในแนวแกน y ที่เลือก

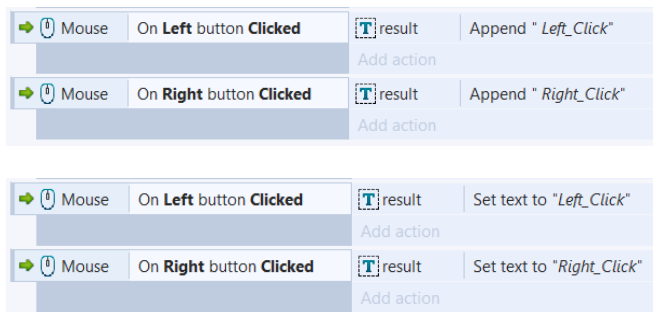
Text

Event

1. Compare text เปรียบเทียบข้อความของ text ว่าตรงกับที่กำหนดหรือไม่

Action

1. Append text เพิ่มข้อความต่อท้ายข้อความเดิม
2. Set text กำหนดข้อความให้ text



Result : Left_Click Left_Click Right_Click Right_Click

Right_Click

Left_Click

(รูปที่ 077 ตัวอย่าง Text Event)

Input

Gamepad

Event Gamepads

1. Gamepads are supported ทำงานเมื่อเครื่องที่รันไฟล์ สามารถต่อ gamepad ได้
2. Has gamepads เช็คว่ามีการต่อ gamepad หรือไม่
3. On gamepad connected ทำงานเมื่อ กำลังเชื่อมต่อ gamepad
4. On gamepad disconnected ทำงานเมื่อ ยกเลิกเชื่อมต่อ gamepad

Input (support X box gamepad)

1. Compare axis เปรียบเทียบระยะของ gamepad
2. Is button down เมื่อปุ่มของ gamepad ถูกกด
3. On any button pressed เมื่อกดปุ่มใด ๆ
4. On any button released เมื่อปล่อยปุ่มใด ๆ
5. On button pressed เมื่อกดปุ่มที่เลือก
6. On button released เมื่อปล่อยปุ่มที่เลือก

Keyboard

Key codes (key code for JavaScript > ex. Space bar > 32)

- | | |
|-------------------------|----------------------------|
| 1. Key code is down | เมื่อปุ่มที่เลือกถูกกดค้าง |
| 2. On key code pressed | เมื่อกดปุ่ม |
| 3. On key code released | เมื่อปล่อยปุ่ม |

Keyboard

- | | |
|------------------------|---------------------|
| 1. Key is down | เมื่อกดปุ่มค้าง |
| 2. On any key pressed | เมื่อกดปุ่มใด ๆ |
| 3. On any key released | เมื่อปล่อยปุ่มใด ๆ |
| 4. On key pressed | เมื่อกดปุ่มที่เลือก |
| 5. On key released | เมื่อกดปุ่มที่เลือก |

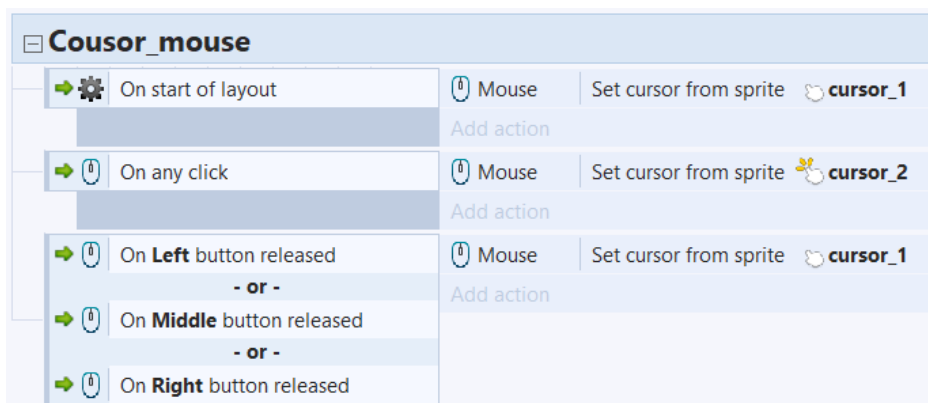
Mouse

Event

- | | |
|--------------------------|--------------------------------------|
| 1. Cursor is over object | เมื่อ Cursor ทับ object ที่เลือก |
| 2. Mouse button is down | เมื่อ คลิกข้างที่เลือก mouse ค้างไว้ |
| 3. On any click | เมื่อคลิกข้างใด ๆ |
| 4. On button released | เมื่อปล่อยข้างใด ๆ |
| 5. On click | เมื่อคลิกข้างที่เลือก(ซ้าย/กลาง/ขวา) |
| 6. On mouse wheel | เมื่อเลื่อนลูกกลิ้ง(ขึ้น/ลง) |
| 7. On object clicked | เมื่อคลิกที่ object |

Action

- | | |
|---------------------------|------------------------------------|
| 1. Set cursor from sprite | เปลี่ยน Cursor ตาม sprite ที่เลือก |
| 2. Set cursor style | เปลี่ยน style Cursor |



(รูปที่ 078 ตัวอย่าง Mouse Event)

Audio

Event Advanced

- | | |
|-----------------------------|--|
| 1. Advanced audio supported | เช็คว่ามีไฟล์เสียงที่รันไฟล์รองรับการแสดงผลของเสียงหรือไม่ |
|-----------------------------|--|

General

- | | |
|----------------------|---|
| 1. Is any playing | เช็คว่ามีไฟล์เสียงใด ๆ กำลังเล่นอยู่หรือไม่ |
| 2. Is silent | เช็คสถานะ silent ทำงานหรือไม่ |
| 3. Is tag playing | เช็ค tag ไฟล์เสียงที่กำหนดว่าทำงานหรือไม่ |
| 4. On ended | เมื่อ tag ไฟล์เสียงที่เลือกเล่นจบ |
| 5. Preloads complete | ทำงานเมื่อไฟล์เสียงพร้อมเล่น |

Action Advanced : Positioned sound

- | | |
|-------------------------------|---|
| 1. Play at object | กำหนดให้เล่น ไฟล์เสียงโดยให้ object ที่เลือกเป็นตำแหน่งต้นเสียง |
| 2. Play at object (by name) | กำหนดให้เล่น ไฟล์เสียงโดยให้ object ที่เลือกเป็นตำแหน่งต้นเสียง |
| 3. Play at position | กำหนดให้เล่น ไฟล์เสียงในตำแหน่งที่เลือก |
| 4. Play at position (by name) | กำหนดให้เล่น ไฟล์เสียงในตำแหน่งที่เลือก |

General

- | | |
|----------------------|------------------------------------|
| 1. Play | กำหนดไฟล์เสียงที่ต้องการให้เล่น |
| 2. Play (by name) | กำหนดไฟล์เสียงที่ต้องการให้เล่น |
| 3. Preload | ดาวน์โหลดไฟล์เสียงจาก sever |
| 4. Preload (by name) | ดาวน์โหลดไฟล์เสียงจาก sever |
| 5. Set looping | ทำให้ tag ที่เลือกหยุดการวนซ้ำ |
| 6. Set master volume | กำหนดระดับเสียงของไฟล์เสียงทั้งหมด |
| 7. Set muted | กำหนดให้ tag ที่เลือกไม่มีเสียง |
| 8. Set paused | กำหนดให้ tag ที่เลือกหยุด |
| 9. Set playback rate | กำหนดความเร็วของ tag ที่เลือก |
| 10. Set silent | กำหนด silent |
| 11. Set volume | กำหนดระดับเสียงให้กับ tag ที่เลือก |
| 12. Stop | หยุดการเล่นของ tag ที่เลือก |
| 13. Stop all | หยุดการเล่นไฟล์เสียงทั้งหมด |

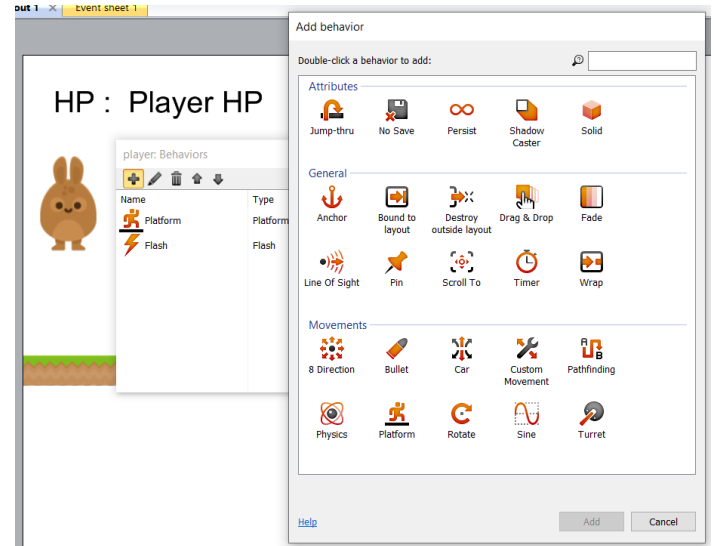
Chapter 4: Behavior

Behavior หรือ พฤติกรรม เป็น attribute ของ object ที่สามารถกำหนดค่าได้

ตัวอย่าง การกำหนด Behavior ให้กับ object
คลิกที่ object > add/edit behavior



(รูปที่ 79 หน้าต่างการเพิ่ม Behavior ให้ object (1))

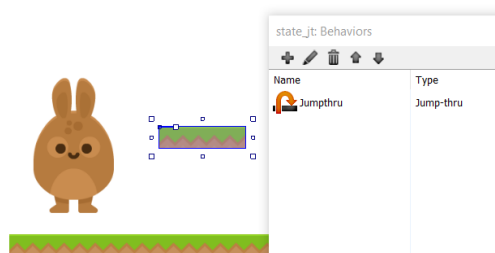


(รูปที่ 80 หน้าต่างการเพิ่ม Behavior ให้ object (2))

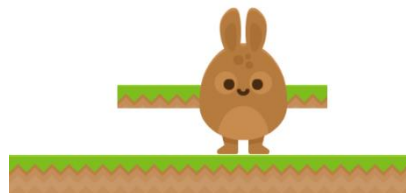
Behavior นั้นแบ่งเป็น 3 กลุ่มใหญ่ๆ คือ Attributes, General, และ Movements

Attributes

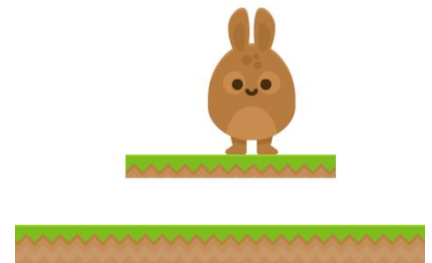
1. Jump-thru ทำให้ object นั้นเป็นพื้นที่สามารถถูกกระโดดขึ้นไปได้



(รูปที่ 81 Jump-thru (1) กำหนดค่า)



(รูปที่ 82 Jump-thru (2) เดินผ่าน)



(รูปที่ 83 Jump-thru (3) กระโดดขึ้น)

Properties

1. Initial state กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่

Event

1. Is enabled เช็คว่าทำงานหรือไม่

Action

1. set enabled กำหนดทำงาน

2. No Save ทำให้ object นั้นไม่มีผลต่อ ฟังก์ชัน save และ load
3. Persist บันทึกค่าสถานะทั้งหมดของ object ไว้ กรณีที่มีการย้าย layout
4. Shadow Caster ทำให้ object นั้นทำงานร่วมกับ shadow light object

Properties

1. Initial state กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่
2. Height กำหนดความสูงของแสง
3. Tag กำหนด tag

Event

1. Compare height เปรียบเทียบความสูงของ object
2. Is enabled เช็คการทำงานหรือไม่

Action

1. set enabled กำหนดให้ทำงาน/ไม่ทำงาน
2. set height กำหนดความสูง
3. set tag กำหนด tag

5. Solid ทำให้ object นั้นมีสถานะเป็นของแข็งไม่สามารถ ผ่าน/ ทะลุได้



Properties

1. Initial state กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่

Event

1. Is enabled เช็คว่าทำงานหรือไม่

Action

1. set enabled กำหนดการทำงาน

General

1. Anchor ทำให้ตำแหน่งของ object นั้นสัมพันธ์กับ ขนาด window size (จอแสดงผล)

Properties

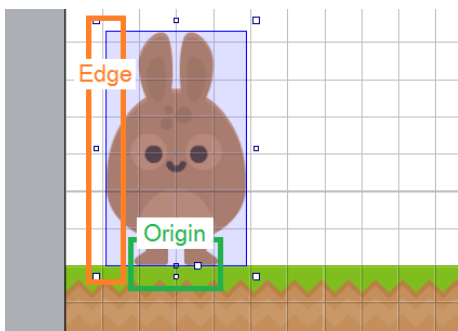
1. Left edge เลือกให้ มุมซ้ายของ object ยึดกับขอบของหน้าจอ
2. Top edge เลือกให้ ด้านบนของ object ยึดกับขอบของหน้าจอ
3. Right edge เลือกให้ มุมขวาของ object ยึดกับขอบของหน้าจอ
4. Bottom edge เลือกให้ ด้านล่างของ object ยึดกับขอบของหน้าจอ
5. Initial state กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่

2. Bound to layout ทำให้ object ไม่สามารถออกนอก layout ได้

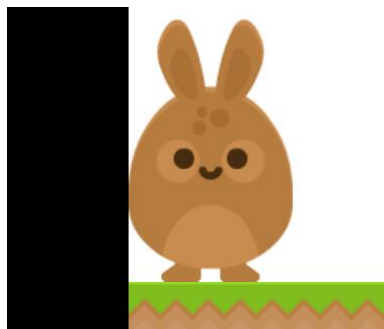
Properties

1. Bound by เลือกตัวกำหนดในการทำงาน (รูปเปรียบเทียบกับ origin / edge)

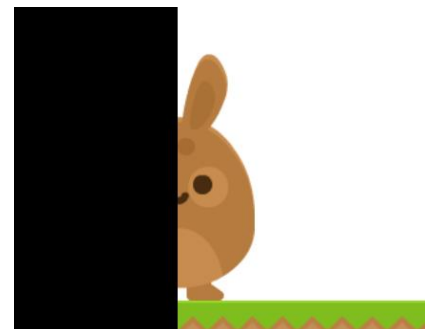
3. Destroy outside layout object จะถูกทำลายเมื่อออกนอก layout



(รูปที่ 087 Bound to layout)



(รูปที่ 088 Bound by edge)



(รูปที่ 089 Bound by origin)

4. Drag & Drop ทำให้ object นั้นสามารถลากและวางได้

Properties

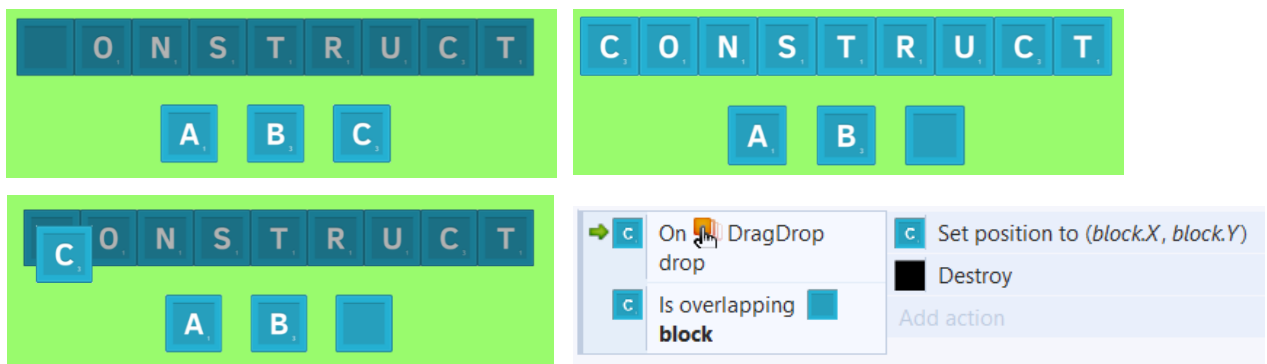
1. Axe กำหนดแนวในการเคลื่อนที่ขณะลาก object
2. Initial state กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่

Event

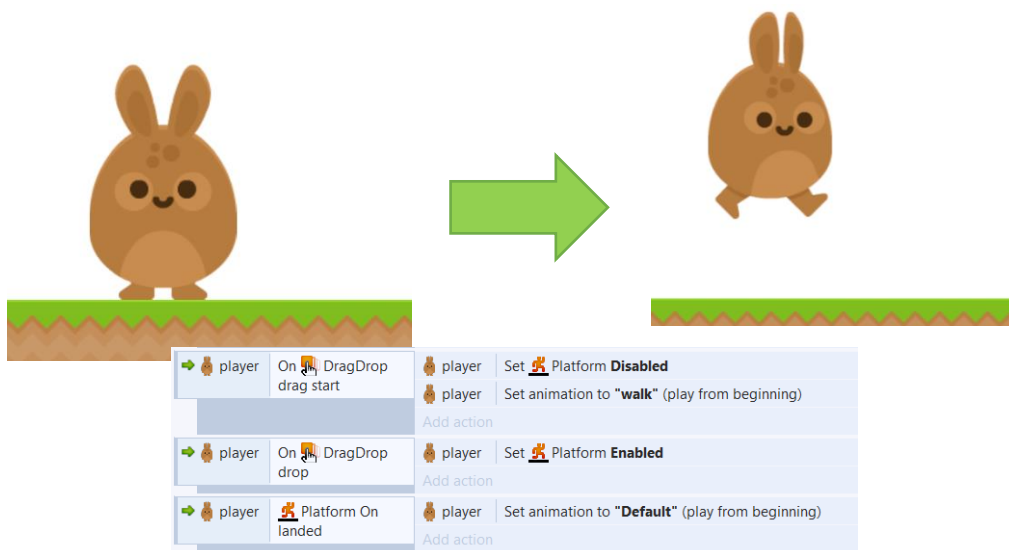
1. Is dragging เช็คว่า object กำลังถูกลากหรือไม่
2. Is enabled เช็คว่าทำงานหรือไม่
3. On drag start ทำงานเมื่อเริ่มลาก
4. On drop ทำงานเมื่อปล่อย

Action

1. Drop ปล่อย object ที่ถูกลากอยู่
2. Set enabled กำหนดให้ทำงาน / ไม่ทำงาน



(รูปที่ 090 ตัวอย่าง Drag & Drop)



(รูปที่ 091 ตัวอย่าง Drag & Drop 2)

5. Fade ทำให้ความทึบแสงของ object นั้นเปลี่ยนแปลงไป

Properties

- | | |
|--------------------|---|
| 1. Active at start | ทำงานเมื่อเริ่ม layout |
| 2. Fade in time | กำหนดเวลาในการ เฟดเข้า |
| 3. Wait time | กำหนดเวลารอการเฟดออก |
| 4. Fade out time | กำหนดเวลาในการเฟดออก |
| 5. Destroy | ทำลาย object เมื่อสิ้นสุดการเฟดออกหรือไม่ |

Event

- | | |
|-------------------------|-------------------------|
| 1. On fade-in finished | เมื่อสิ้นสุดการเฟดเข้า |
| 2. On fade-out finished | เมื่อสิ้นสุดการเฟดออก |
| 3. On wait finished | ทำงานเมื่อมีการรอการเฟด |

Action

- | | |
|----------------------|----------------------|
| 1. Restart fade | ให้เฟดทำงานอีกครั้ง |
| 2. Set fade-in time | กำหนดเวลา เฟดเข้า |
| 3. Set fade-out time | กำหนดเวลา เฟดออก |
| 4. Set wait time | กำหนดเวลารอการเฟดออก |
| 5. Start fade | สั่งเริ่มการเฟด |

6. Flash ทำให้ object กระพริบ

Event

- | | |
|-------------------|-------------------------------------|
| 1. Is flashing | เช็ค ว่า object กำลัง flash หรือไม่ |
| 2. On flash ended | ทำงานเมื่อจบการ flash |

Action

- | | |
|------------------|------------------|
| 1. Flash | ตั้งค่าการ flash |
| 2. Stop flashing | สั่งหยุด flash |

7. Line of Sight ทำให้ object สามารถตรวจหา object อื่น ๆ ในระยะที่กำหนดได้

Properties

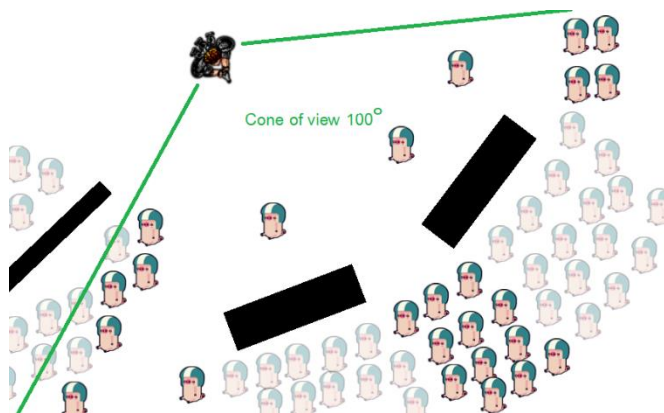
- | | |
|------------------------|--|
| 1. Obstacles | เลือกชนิดของเป้าหมาย |
| 2. Range | ระยะการมองเห็น |
| 3. Cone of view | ขอบเขตการมองเห็น |
| 4. Use collision cells | กำหนดให้ object สามารถทำงานกับการชนได้ |

Event

1. Has LOS to object เมื่อพบ object ที่กำหนด
2. Has LOS to position เมื่อได้ตำแหน่งที่กำหนด

Action

1. Add obstacle กำหนด object เป้าหมาย
2. Clear obstacle ยกเลิกการเลือกเป้าหมาย
3. Set cone of view กำหนดขอบเขตการมองเห็น
4. Set range กำหนดระยะการมองเห็น



System	Every tick	Player	Set angle toward (<i>Mouse.X</i> , <i>Mouse.Y</i>)
		Pig	Set opacity to 33
		Add action	
Player	Has LineOfSight to Pig	Pig	Set opacity to 100
		Add action	

(รูปที่ 092 Line of Sight)

8. Pin ตรึง object ไว้กับที่

Event

1. Is pinned เช็คว่า pin ทำงานหรือไม่

Action

1. Pin to object กำหนดให้ object pin ที่ object อื่น
2. Unpin ยกเลิกการ pin

9. Scroll to จอแสดงผลจะเลื่อนตามการเคลื่อนไหวของ object (object จะเป็นจุดศูนย์กลาง)

Properties

1. Initial state กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่

Action

1. Set enabled กำหนดสถานะการทำงาน
2. Shake ทำให้เกิดจอแสดงผลเกิดการเขย่า

10. Timer ทำให้ object สามารถการทำงานที่เดียวซ้ำกับเวลาได้หลายคำสั่ง

Event

1. On timer ทำงานเมื่อครบเวลาของ tag ที่ครบกำหนด

Action

1. Start timer กำหนด tag และ ช่วงเวลา
2. Stop timer หยุดการทำงานของ tag ที่กำหนด

11. Warp ทำให้ object โผล่มาจากด้านตรงกันข้ามเมื่อ object นั้นออกนอก layout

Properties

1. warp to กำหนดตำแหน่งในการโผล่ออกมา

Movements

1. 8 Direction ทำให้ object สามารถเคลื่อนที่ได้ สูงสุด 8 ทิศทาง

Properties

- | | |
|---------------------|-----------------------------------|
| 1. Max speed | กำหนดความเร็วสูงสุดของ object |
| 2. Acceleration | กำหนดความเร่งของ object |
| 3. Deceleration | กำหนดความหน่วงของ object |
| 4. Directions | กำหนดแนวในการเคลื่อนที่ |
| 5. Set angle | กำหนดมุมในการหมุน |
| 6. Default controls | กำหนดการควบคุมพื้นฐาน (ปุ่มลูกศร) |
| 7. Initial state | กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่ |

Event

- | | |
|------------------|---------------------------------------|
| 1. Compare speed | เปรียบเทียบความเร็วของ object |
| 2. Is moving | เช็ค ว่า object นั้นเคลื่อนที่หรือไม่ |

Action

- | | |
|-----------------------|-------------------------|
| 1. Reverse | กำหนดทิศทางตรงกันข้าม |
| 2. Set acceleration | กำหนดความเร่ง |
| 3. Set deceleration | กำหนดความหน่วง |
| 4. Set enable | กำหนดสถานะการทำงาน |
| 5. Set ignoring input | กำหนดให้ไม่รับ input |
| 6. Set max speed | กำหนดความเร็วสูงสุด |
| 7. Set speed | กำหนดความเร็ว |
| 8. Set vector X | กำหนดตำแหน่งในแนวแกน X |
| 9. Set vector Y | กำหนดตำแหน่งในแนวแกน Y |
| 10. Simulate control | กำหนดจำลองการเคลื่อนที่ |
| 11. Stop | สั่งหยุดการเคลื่อนที่ |

2. Bullet ทำให้ object เคลื่อนที่แบบกระสุน



(รูปที่ 093 Bullet)

Properties

- | | |
|---------------------|---|
| 1. speed | กำหนดความเร็วในการเคลื่อนที่ |
| 2. Acceleration | กำหนดความเร่ง |
| 3. Gravity | กำหนดค่าแรงโน้มถ่วง |
| 4. Bounce off solid | กำหนดให้กระเด็นเมื่อชนกับ object ที่มีสถานะ solid |
| 5. Set angle | กำหนดมุม |
| 6. Initial state | กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่ |

Event

- | | |
|-------------------------------|------------------------------|
| 1. Compare distance travelled | เปรียบเทียบระยะทางการเคลื่อน |
| 2. Compare speed | เปรียบเทียบความเร็ว |

Action

- | | |
|---------------------------|---|
| 1. Bounce off object | กำหนดให้กระสุนกระเด็นเมื่อชนกับ object ที่เลือก |
| 2. Set acceleration | กำหนดความเร่ง |
| 3. Set angle of motion | กำหนดมุมในการเคลื่อนที่ |
| 4. Set distance travelled | กำหนดระยะทาง |
| 5. Set enabled | กำหนดสถานะการทำงาน |
| 6. Set gravity | กำหนดค่าแรงโน้มถ่วง |
| 7. Set speed | กำหนดความเร็ว |

3. Car ทำให้ object มีการบังคับแบบรถ

Properties

1. Max speed	กำหนดความเร็วสูงสุดของ object
2. Acceleration	กำหนดความเร่งของ object
3. Deceleration	กำหนดความหน่วงของ object
4. Steer speed	กำหนดความเร็วในการหมุน
5. Drift recover	กำหนดแรงเหวี่ยงขณะหมุน
6. Friction	กำหนดการชน(0 ชนแบบยืดหยุ่น ถึง 1 ชนแบบไม่ยืดหยุ่น)
7. Set angle	กำหนดมุมในการหมุน
8. Default controls	กำหนดการควบคุมพื้นฐาน (ปุ่มลูกศร)
9. Initial state	กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่

Event

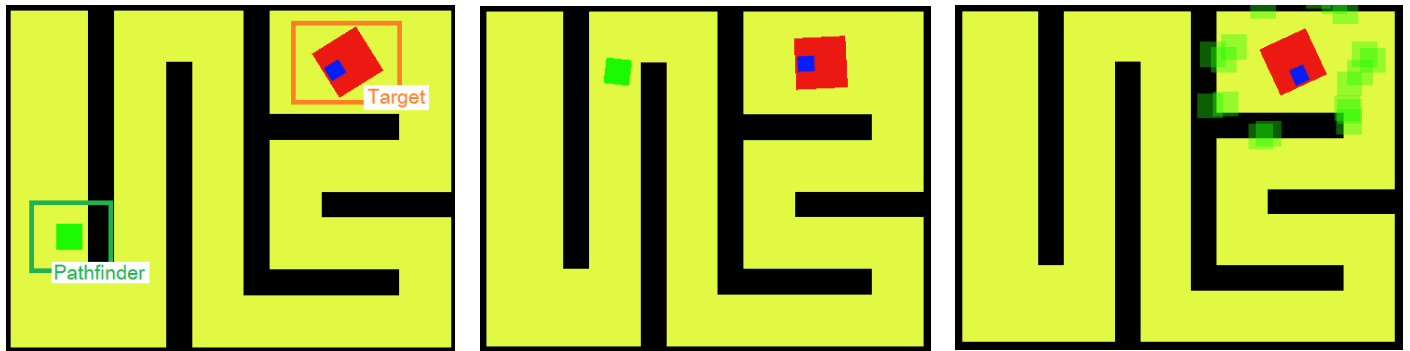
1. Compare speed	เปรียบเทียบความเร็ว
2. Is moving	เช็คว่ากำลังเคลื่อนที่หรือไม่

Action

1. Set acceleration	กำหนดความเร่ง
2. Set deceleration	กำหนดความหน่วง
3. Set drift recover	กำหนดความหน่วง
4. Set enable	กำหนดสถานะการทำงาน
5. Set friction	กำหนดการชน
6. Set ignoring input	กำหนดให้ไม่รับ input
7. Set max speed	กำหนดความเร็วสูงสุด
8. Set speed	กำหนดความเร็ว
9. Set steer speed	กำหนดความเร็วในการหมุน
10. Simulate control	กำหนดจำลองการเคลื่อนที่
11. Stop	สั่งหยุดการเคลื่อนที่

4. Custom Movement ใช้กำหนดการเคลื่อนที่ของ object

5. Pathfinding ทำให้ object เคลื่อนที่ไปหาเป้าหมาย



(รูปที่ 094 Pathfinding)

Properties

- | | |
|-------------------|-------------------------------------|
| 1. Cell size | กำหนดขนาดของ cell (ระยะการมองเห็น) |
| 2. Cell border | กำหนดขอบเขตการค้นหา |
| 3. Obstacles | กำหนดอุปสรรคเพื่อหลีกเลี่ยง |
| 4. Max speed | กำหนดความเร็วสูงสุด |
| 5. Acceleration | กำหนดความเร่งของ object |
| 6. Deceleration | กำหนดความหน่วงของ object |
| 7. Rotate speed | กำหนดความเร็วในการหมุน |
| 8. Rotate object | กำหนดให้ object สามารถหมุนได้ |
| 9. Diagonals | กำหนดให้เคลื่อนที่แบบเส้นทแยงมุมได้ |
| 10. Initial state | กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่ |

Event

- | | |
|---------------------------|-----------------------------------|
| 1. Compare speed | เปรียบเทียบความเร็ว |
| 2. Diagonals are enabled | เช็คสถานะของ Diagonals |
| 3. Is calculating path | เช็คที่กำลังคำนวณเส้นทางหรือไม่ |
| 4. Is moving along path | เช็คกำลังเดินทางตามเส้นทางหรือไม่ |
| 5. On arrived | เมื่อถึงจุดหมาย |
| 6. On failed to find path | เมื่อหาเส้นทางไม่เจอ |
| 7. On path found | เมื่อพบเส้นทาง |

Action

Normal

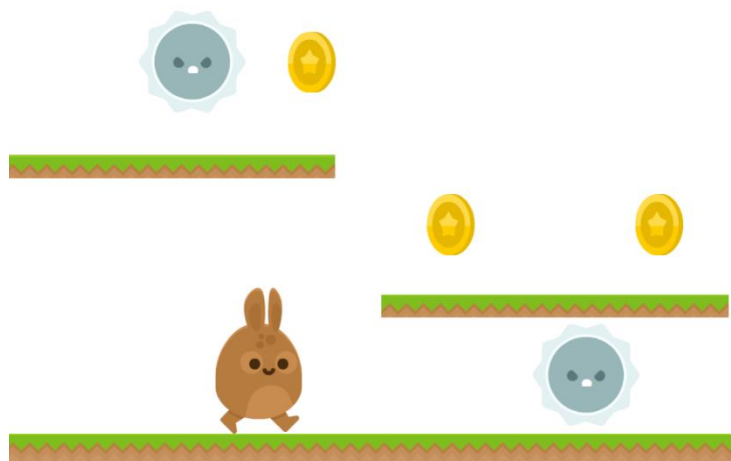
- | | |
|--------------------|-------------------------------|
| 1. Add obstacle | เพิ่มสิ่งกีดขวาง |
| 2. Clear obstacles | ลบรายการสิ่งกีดขวางที่เพิ่มไป |
| 3. Find path | ค้นหาเส้นทาง |
| 4. Set enabled | กำหนดสถานะการทำงาน |

Movement

- | | |
|--------------------------|---------------------------|
| 1. Move along path | เดินทางตามเส้นทาง |
| 2. Set acceleration | กำหนดความเร่ง |
| 3. Set deceleration | กำหนดความหน่วง |
| 4. Set diagonals enabled | กำหนดการเดินทางแบบทแยงมุม |
| 5. Set max speed | กำหนดความเร็วสูงสุด |
| 6. Set rotate speed | กำหนดความเร็วในการหมุน |
| 7. Set speed | กำหนดความเร็ว |
| 8. Stop | สั่งให้หยุด |

6. Physics ทำให้ object มีการเคลื่อนที่เป็นไปตามกฎฟิสิกส์ (แรงโน้มถ่วง)

7. Platform ทำให้ object สามารถเคลื่อนที่, กระโดดบน platform (object ที่เป็น solid) ได้



(รูปที่ 094 Platform)

Properties

- | | |
|---------------------|---|
| 1. Max speed | กำหนดความเร็วสูงสุด |
| 2. Acceleration | กำหนดความเร่ง |
| 3. Deceleration | กำหนดความหน่วง |
| 4. Jump strength | กำหนดกำลังในการกระโดด |
| 5. Gravity | กำหนดค่าแรงโน้มถ่วง |
| 6. Max fall speed | กำหนดความเร็วสูงสุดในการตก |
| 7. Double jump | กำหนดสถานะการกระโดดสองครั้ง |
| 8. Jump sustain | กำหนดแรงกระโดดที่จะเพิ่มขึ้นเมื่อมีการกดปุ่มกระโดดค้างไว้ |
| 9. Default controls | กำหนดการควบคุมพื้นฐาน(แป้นลูกศร) |
| 10. Initial state | กำหนดสถานะเริ่มต้นว่าทำงานหรือไม่ |

Event

Normal

- | | |
|---------------------------|---------------------------------------|
| 1. Compare speed | เปรียบเทียบความเร็ว |
| 2. Is by wall | เช็คว่ายชนกับกำแพงซ้าย/ขวาหรือไม่ |
| 3. Is double-jump enabled | เช็คว่ายสามารถกระโดดสองชั้นได้หรือไม่ |
| 4. Is falling | เช็คว่ายกำลังตกหรือไม่ |
| 5. Is jumping | เช็คว่ายกำลังกระโดดหรือไม่ |
| 6. Is moving | เช็คว่ายกำลังเคลื่อนที่หรือไม่ |
| 7. Is on floor | เช็คว่ายอยู่บนพื้นหรือไม่ |

Animation triggers

- | | |
|---------------|-----------------------------|
| 1. On fall | ทำงานเมื่อเริ่มตกลงด้านล่าง |
| 2. On jump | ทำงานเมื่อกระโดด |
| 3. On landed | ทำงานเมื่ออยู่บนพื้น |
| 4. On moved | ทำงานเมื่อเริ่มเคลื่อนไหวน |
| 5. On stopped | ทำงานเมื่อหยุด |

8. Rotate ทำให้ object หมุนตามที่กำหนด

Properties

1. Speed กำหนดความเร็ว ค่า + หมุนตามเข็มนาฬิกา, ค่า - หมุนทวนเข็มนาฬิกา
2. Acceleration กำหนดความเร่ง

Action

1. Set acceleration กำหนดความเร่ง
2. Set speed กำหนดความเร็ว

9. Sine ทำให้ object สามารถเคลื่อนที่แบบคลื่น ทั้งแนวนอนและแนวตั้ง

Properties

1. Active on start ทำงานเมื่อเริ่ม layout
2. Movement รูปแบบการเคลื่อนไหว
3. Wave รูปแบบคลื่น
4. Period เวลาในการเคลื่อนที่ 1 รอบ
5. Magnitude กำหนดระยะทางในการเคลื่อนที่ 1 รอบ

Event

1. Compare magnitude เปรียบเทียบระยะทาง
2. Compare movement เปรียบเทียบรูปแบบการเคลื่อนที่
3. Compare period เปรียบเทียบเวลาในการเคลื่อนที่ 1 รอบ
4. Compare wave เปรียบเทียบรูปแบบของคลื่น
5. Is active เช็คว่าทำงานหรือไม่

Action

1. Set active กำหนดการทำงาน
3. Set magnitude กำหนดระยะทาง
4. Set movement กำหนดรูปแบบการเคลื่อนที่
5. Set period กำหนดระยะเวลาในการเคลื่อนที่ 1 รอบ
6. Set wave กำหนดรูปแบบคลื่น
7. Update initial state กำหนดให้สถานะทั้งหมดกลับไปเป็นพื้นฐานตามที่ตั้งค่าไว้

10. Turret ทำให้ object มีคุณสมบัติของป้อมปืน สามารถ ตรวจจับ, หันไปหาเป้าหมาย, และยิงได้

Properties

- | | |
|---------------------|---------------------------------------|
| 1. Range | ระยะในการมองเห็นเป้าหมาย |
| 2. Rate of fire | อัตราการยิง |
| 3. Rotate | การหมุน |
| 4. Rotate speed | ความเร็วในการหมุน |
| 5. Target mode | รูปแบบในการเลือกเป้าหมาย |
| 6. Predictive aim | หันตามเป้าหมายล่วงหน้า |
| 7. Projectile speed | ความเร็วของ การหันตามเป้าหมายล่วงหน้า |
| 8. Initial state | สถานะการทำงาน |

Event

- | | |
|-----------------------|----------------------|
| 1. Has target | เมื่อพบเป้าหมาย |
| 2. On shoot | ทำงานเมื่อเริ่มยิง |
| 3. On target acquired | ทำงานเมื่อพบเป้าหมาย |

Action

Normal

- | | |
|-------------------------|----------------|
| 1. Acquire target | กำหนดเป้าหมาย |
| 2. Add object to target | เพิ่มเป้าหมาย |
| 3. Clear targets | ยกเลิกเป้าหมาย |
| 4. Unacquire target | ยกเลิกเป้าหมาย |

Settings

- | | |
|-------------------------|---------------------------------|
| 1. Set enabled | กำหนดสถานะการทำงาน |
| 2. Set predictive aim | กำหนดการ หันตามเป้าหมายล่วงหน้า |
| 3. Set projectile speed | กำหนดความเร็วของ predictive aim |
| 4. Set range | กำหนดระยะตรวจจับ |
| 5. Set rate of fire | กำหนดอัตราการยิง |
| 6. Set rotate | กำหนดการหมุน |
| 7. Set rotate speed | กำหนดความเร็วในการหมุน |
| 8. Set target mode | กำหนดรูปแบบในการเลือกเป้าหมาย |

Chapter: 5 Let's make a game

ในบทนี้เราจะเริ่มสร้างเกมด้วยโปรแกรม Construct2 โดยจะสร้างเกม 2 ประเภทคือ top down และ platform

Top down game

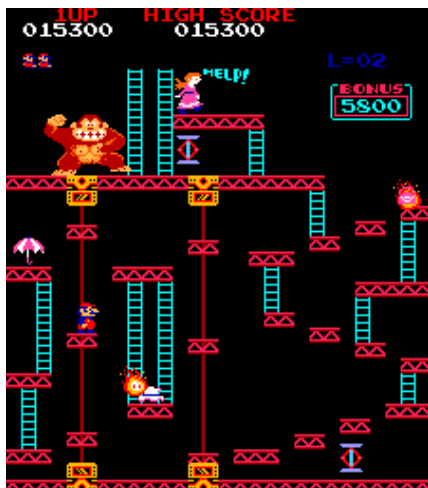
เป็นเกมที่นำเสนอผ่านมุมมองจากด้านบนของผู้เล่น โดยส่วนมากจะเป็นเกม แนว Shoot 'Em Up หรือ เกมยิงในมุมมอง 2 มิติ โดยมีเกมต้นแบบคือ เกมตู้จากประเทศญี่ปุ่นในชื่อ Space Invaders ปี ค.ศ. 1978



(รูปที่ 095 ตัวอย่างเกมมุมมอง top down)

game

เป็นหนึ่งในแขนงเกมแอ็คชั่น ที่ผู้เล่นจะต้องควบคุมตัวละครมีปฏิสัมพันธ์กับสิ่งของต่าง ๆ รอบตัว เช่น การกระโดด การปีน การเดินทางผ่านอุปสรรคต่าง ๆ เพื่อไปถึงเป้าหมายของเกม



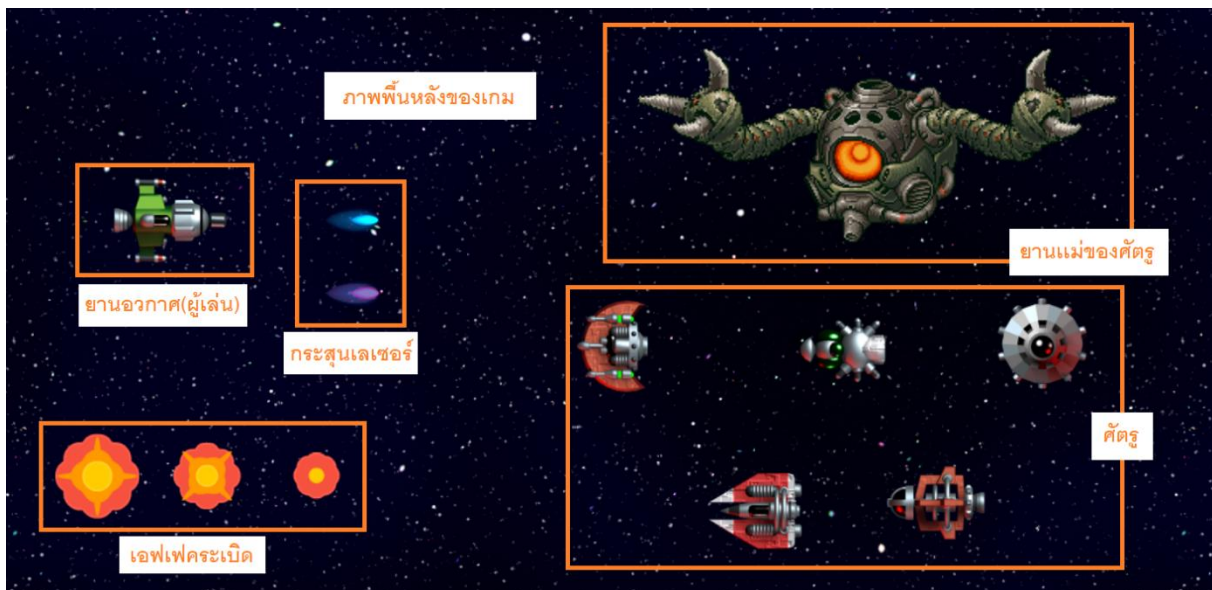
(รูปที่ 096 ตัวอย่างเกม platform)

5 step to make a game

1. choose genre เลือกประเภทของเกมที่ต้องการ
2. story เนื้อเรื่อง (หากมี)
3. theme, sprite, sound, and effect คี๊ม + ตัวละคร
4. gameplay ระบบเกม
5. level design ออกแบบระดับความยากง่ายในการเล่น
6. test and optimize ทดสอบและพัฒนา

Top down game

1. ประเภทของเกม side scrolling 2 D shoot 'em up
2. เนื้อเรื่อง ในอวกาศแห่งหนึ่งได้มีมนุษย์อาศัยอยู่ได้ถูกรุกรานจากผู้ล่าอวกาศต่างแดน คุณจำต้องขับยานอวกาศฝ่าลงล้อมข้าศึกเพื่อไปจัดการยานแม่ของศัตรูเพื่อตัดขาดการเชื่อมต่อของศัตรู และปกป้องสถานที่แห่งนี้เอาไว้
3. ในส่วนของ คี๊มและตัวละคร จะเลือกฉากแหล่งที่เป็นภาพอวกาศ ดังนี้



(รูปที่ 097 คี๊มและตัวละครที่เลือกใช้)

4. ในระบบการเล่น ยานของเราจะยิงอัตโนมัติ ซึ่งความสามารถในการยิงนั้นสามารถอัพเกรดได้ 3 ระดับโดยจะเพิ่มเมื่อกำจัดศัตรูได้ตามที่กำหนด โดยผู้เล่นจะสามารถบังคับได้ 8 ทิศทาง และผู้เล่นจะมีพลังชีวิตทั้งหมด 5 หน่วย และไม่มีระบบพื้นพลัง

5. การออกแบบระดับความยาก เราจะอิงจากเลเวลของผู้เล่น คือ 1 – 3 โดยจะแบ่งเกม เป็น 4 ช่วงดังนี้

1. ช่วงเริ่มเกม ผู้เล่นจะมีเลเวล 1 และจำนวนศัตรูที่ออกมาจะมีปริมาณน้อย แต่ก็ยังสามารถทำให้ผู้เล่นแพ้ได้หากมีสมาธิไม่มากพอ

2. ช่วง 2 หลังจากเลเวลอัพ ความสามารถในการยิงของผู้เล่นจะเพิ่มขึ้น และจำนวนศัตรูก็นั่นขึ้น ในช่วงนี้จะมีการเพิ่มศัตรูที่มีพลังชีวิตสูงและสามารถทำให้กระสุนของผู้เล่นกระเด็นออกไปได้เมื่อ กระสุนชนตัวมัน ซึ่งหากผู้เล่นชนกับกระสุนที่กระเด็นกลับมาก็จะได้รับคามเสียหายด้วย

3. ช่วงที่ 3 จะมีการปล่อยฝูงของศัตรูที่เพิ่มขึ้น และหากฝ่ายช่วงนี้ไปได้จะเข้าสู่การต่อสู้กับบอสในช่วงที่ 4

4. ช่วงที่ 4 หากทำการชนะบอสได้จะเป็นการจบเกม

ขั้นตอนการสร้างเกม

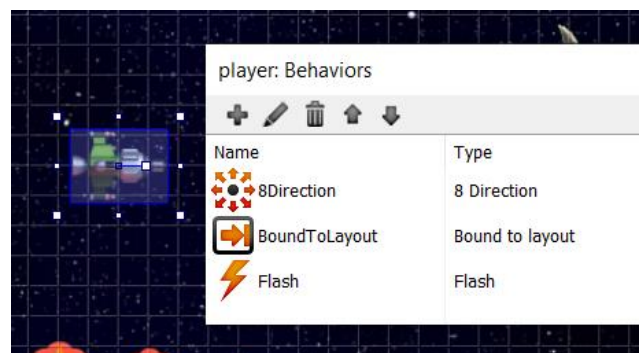
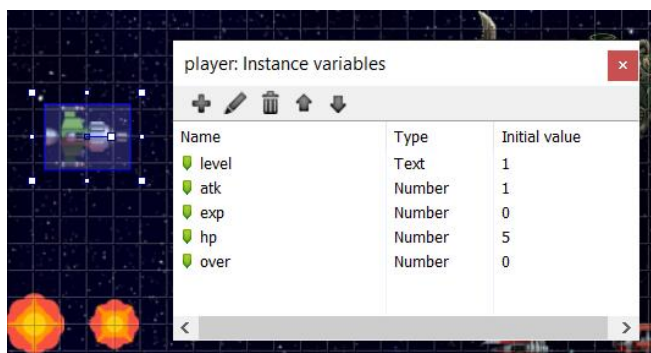
1. Insert object โดยด้วยของเกม

- ภาพพื้นหลัง

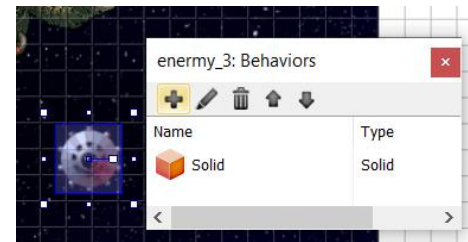
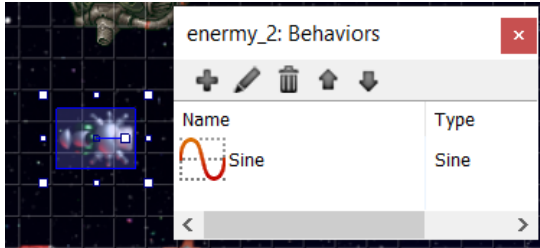
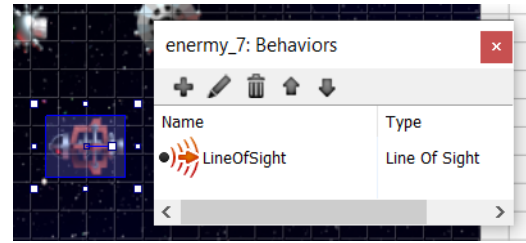
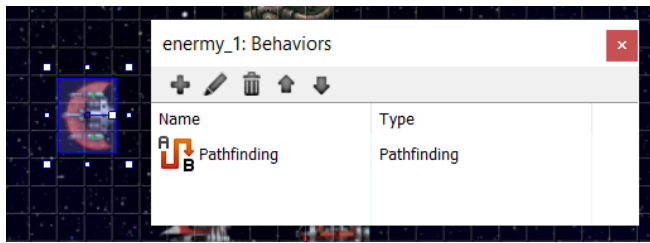
- sprite ตัวละคร, ศัตรู, กระสุน, เอฟเฟค

2. เพิ่มค่าตัวแปร (instance variable) ที่จำเป็น เช่น HP(พลังชีวิต) เป็นต้น

3. คิดความสามารถเพื่อกำหนด behavior ให้กับ object



(รูปที่ 098 Instance variable และ behavior ของ player)



(รูปที่ 099 behavior ของ enemy)

4. เขียน Event sheet ให้ตรงกับที่คิดไว้ เช่น ยานของเราจะยิงอัตโนมัติ ซึ่งความสามารถในการยิงนั้นสามารถอัพเกรดได้ 3 ระดับโดยจะเพิ่มเมื่อกำจัดศัตรูได้ตามที่กำหนด โดยผู้เล่นจะสามารถบังคับได้ 8 ทิศทาง และผู้เล่นจะมีพลังชีวิตทั้งหมด 5 หน่วย และไม่มีระบบฟื้นฟูพลัง เป็นต้น



System	Every 0.5 seconds	Add action
player	level = "1"	Spawn laser on layer 1 (image point 1)
player	level = "2"	Spawn laser on layer 1 (image point 2)
player	level = "2"	Spawn laser on layer 1 (image point 3)
player	level = "3"	Spawn laser on layer 1 (image point 4)
player	level = "3"	Spawn laser on layer 1 (image point 5)
player	level = "3"	Spawn laser on layer 1 (image point 6)

(รูปที่ 100 event ของ player(1))



System	Every tick	player_hp	Set width to 64 * player.hp
player	level = "1"	exp_bar	Set width to 8 * player.exp
player	level = "2"	exp_bar	Set width to 8 * player.exp
player	level = "3"	exp_bar	Set width to 8 * player.exp

(รูปที่ 101 event ของ player(2))

5. กำหนด Event ให้กับ enemy



enemy_1	On created	enemy_1	Set state to "ready"
Add action			
enemy_1	state = "ready"	enemy_1	Spawn laser2 on layer 0 (image point 1)
Add action			
e1_path		e1_path	Set position to (random(900, 1200) , random(20, 700))
Add action			
enemy_1		enemy_1	Find path to (e1_path.X, e1_path.Y)
Add action			
enemy_1		enemy_1	Set state to "reC"
Add action			
enemy_1	On Pathfinding path found	enemy_1	Move along path
Add action			
enemy_1	On Pathfinding arrived	enemy_1	Set state to "ready"
Add action			
laser2	Is on-screen	laser2	Set angle to 180 degrees
Add action			

(รูปที่ 102 event ของ enemy(1))

enemy_2	On created	System	Add 1 to e2_show
Add action			
System	e2_show = 10	System	Wait 5 seconds
Add action			
System	Trigger once	System	Set e2_show to 0
Add action			
enemy_2	On created	enemy_2	Set Sine magnitude to -200
Add action			
enemy_2		enemy_2	Set Sine period to 6
Add action			
enemy_2		enemy_2	Set Sine movement to Vertical
Add action			
enemy_2_alt...	On created	enemy_2_alt...	Set Sine magnitude to 200
Add action			
enemy_2_alt...		enemy_2_alt...	Set Sine period to 6
Add action			
enemy_2_alt...		enemy_2_alt...	Set Sine movement to Vertical
Add action			
enemy_2	Is Sine active	enemy_2	Move forward -2 pixels
Add action			
enemy_2...	Is Sine active	enemy_2...	Move forward -2 pixels
Add action			

(รูปที่ 103 event ของ enemy(2))

คำอธิบาย

1. กำหนดเมื่อสร้าง enemy ให้มีสถานะ ready
2. เมื่อ enemy มีสถานะ ready ทำสร้าง object ที่มี behavior เป็นกระสุนจากนั้น ทำการสุ่ม path และให้ enemy หา path ที่สุ่มตำแหน่งแล้ว เปลี่ยน สถานะเป็น reC
3. เมื่อ enemy พอ path ให้เดินทางไปหา path
4. เมื่อ enemy ไปถึงตำแหน่งของ path เปลี่ยน สถานะเป็น ready วนซ้ำไปที่ข้อ 1 จนกว่าจะถูกทำลาย
5. กำหนดให้ กระสุนของ enemy มีองศา 180

คำอธิบาย

1. เมื่อ enemy และ enemy_alter ถูกสร้าง กำหนดคุณสมบัติของ behavior sine ดังนี้
1. กำหนดระยะทาง -200 และ 200
2. กำหนดเวลาในเดินทางครบ 1 รอบ
3. กำหนดรูปแบบในการเคลื่อนที่ เป็นแนวตั้ง
4. เมื่อ behavior sine ของ enemy และ enemy_alter ทำงานกำหนดให้ เคลื่อนที่เข้าหาผู้เล่น



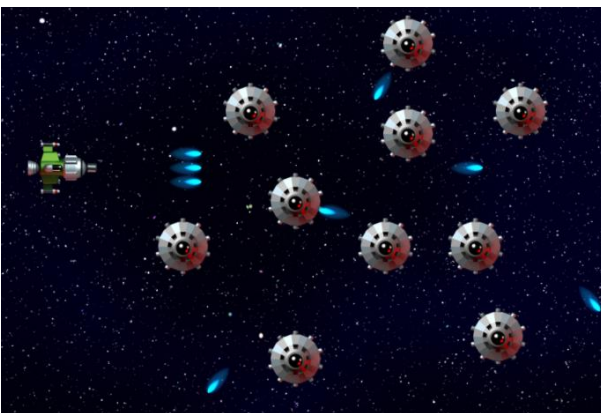
energy_7	On created	energy_7	Set LineOfSight range to 100
Add action			
energy_7	On destroyed	energy_7	Spawn Particles on layer 0 (image point 0)
Add action			
energy_7	Has LineOfSight to player	energy_7	Destroy
Add action			
energy_7	Has LineOfSight to player	energy_7	Move forward -3 pixels
Add action			

คำอธิบาย

1. เมื่อ enemy ถูกสร้างกำหนดให้ มีค่าการมองเห็นของ behavior los
2. เมื่อ enemy ถูกทำลาย ให้สร้าง Particles ขึ้น
3. เมื่อ enemy พบ player ให้ทำลายตัวเอง
4. เมื่อ enemy ไม่พบ player ให้เคลื่อนที่ไปทางซ้าย

(รูปที่ 104 event ของ enemy(3))

player	level = "2"	Add action	
System	Every 5 seconds	System	Create object enemy_1 on layer 1 at (random(900, 1200) , random(20, 700))
Add action			
System	Every 5 seconds	System	Create object enemy_7 on layer 1 at (random(900, 1200) , random(20, 700))
Add action			
System	Every 15 seconds	System	Create object enemy_3 on layer 1 at (random(900, 1200) , random(20, 700))
		System	Create object enemy_3 on layer 1 at (random(900, 1200) , random(20, 700))
		System	Create object enemy_3 on layer 1 at (random(900, 1200) , random(20, 700))
		System	Create object enemy_3 on layer 1 at (random(900, 1200) , random(20, 700))
Add action			



คำอธิบาย

1. enemy มี behavior เป็น solid
2. กระสุนของ player ถูกกำหนดให้กระเด็นออกเมื่อชนกับ solid

(รูปที่ 105 event ของ enemy(4))

6. กำหนด level design



state management			
player	level = "1"	Add action	
System	Every 10 seconds	System	Create object enemy_1 on layer 1 at (random(900, 1200) , random(20, 700))
		Add action	
System	Every 5 seconds	System	Create object enemy_7 on layer 1 at (random(900, 1200) , random(20, 700))
		Add action	
player	level = "2"	Add action	
System	Every 5 seconds	System	Create object enemy_1 on layer 1 at (random(900, 1200) , random(20, 700))
		Add action	
System	Every 5 seconds	System	Create object enemy_7 on layer 1 at (random(900, 1200) , random(20, 700))
		Add action	
System	Every 15 seconds	System	Create object enemy_3 on layer 1 at (random(900, 1200) , random(20, 700))
		System	Create object enemy_3 on layer 1 at (random(900, 1200) , random(20, 700))
		System	Create object enemy_3 on layer 1 at (random(900, 1200) , random(20, 700))
		System	Create object enemy_3 on layer 1 at (random(900, 1200) , random(20, 700))
		Add action	
player	level = "3"	Add action	
player	over = 0	Add action	
System	Every 0.6 seconds	System	Create object enemy_2 on layer 1 at (1200, 300)
System	e2_show < 10	System	Create object enemy_2_alter on layer 1 at (1200, 600)
		Add action	
player	over > 0	System	Create object boss on layer 1 at (1100, 350)
System	Trigger once	Add action	

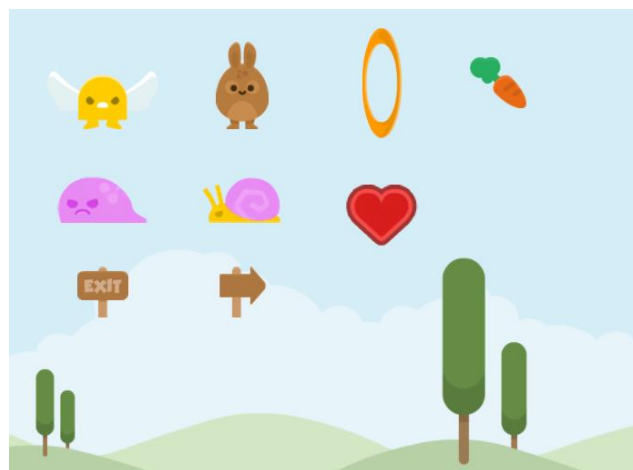
(รูปที่ 106 event level design)

Platform

1. ประเภทของเกม Platform 2D side scrolling
2. เนื้อเรื่อง การเดินทางของกระต่ายน้อยตัวหนึ่งที่หลงเข้าไปในป่า และในที่แห่งนั้นได้มีอันตรายจากสิ่งต่าง ๆ รอบตัว เขาจะสามารถเดินทางออกจากป่าแห่งนี้ได้หรือไม่ นั่นก็เป็นหน้าที่ของคนที่ต้องไปเล่นเอง
3. ในส่วนของ ฮีโร่และตัวละคร
4. ในระบบการเล่น การควบคุมเป็นแบบพื้นฐานของ platform โดยจะมีระบบโปรแกรมกระต่ายที่ใช้สำหรับการเดินทางข้ามฉากได้
5. การออกแบบระดับความยาก จะเป็นการได้ระดับ โดยจะเพิ่มศัตรูและอุปสรรคตามลำดับ

ขั้นตอนการสร้างเกม

1. Insert object โดยด้วยของเกม
 - ภาพพื้นหลัง
 - sprite ตัวละคร, ศัตรู
2. เพิ่มค่าตัวแปร (instance variable) ที่จำเป็น เช่น HP(พลังชีวิต) เป็นต้น
3. คิดความสามารถเพื่อกำหนด behavior ให้กับ object



(รูปที่ 106 ฮีโร่และตัวละครที่เลือกใช้)

4. เขียน Event sheet และ Behavior ให้ตรงกับที่คิดไว้ ทั้งส่วนของ player และ enemy

Player

Keyb...	On Left arrow pressed	player	Set Mirrored
Keyb...	On Right arrow pressed	player	Set Not mirrored
player	Platform On moved	player	Set animation to "walk" (play from beginning)
player	Platform On stopped	player	Set animation to "Default" (play from beginning)
player	Platform On jump	player	Set animation to "jump" (play from beginning)
player	Platform On landed	player	Set animation to "walk" (play from beginning)
player	Platform is moving	player	Set animation to "walk" (play from beginning)
player	Platform is moving	player	Set animation to "Default" (play from beginning)
player	On collision with enemy	player	Subtract 1 from HP
player	HP > 0	player	Flash: Flash 0.1 on 0.1 off for 0.4 seconds
player	Platform is falling	player	
player	Is flashing	player	
player	On collision with enemy	enemy	Fade: start fade
player	Platform is falling	player	Set Platform vector Y to -700
player	On collision with carrot	System	Add 1 to carrot
carrot		carrot	Destroy

Behaviors

ScrollTo	Initial state	Enabled
Platform	Max speed	330
	Acceleration	1500
	Deceleration	1500
	Jump strength	650
	Gravity	1500
	Max fall spe...	1000
	Double jump	Enabled
	Jump sustain	0
	Default contr...	Yes
	Initial state	Enabled
Flash		(no properties)

การเล่นอนิเมชัน

การชนกับ enemy

การเก็บ carrot

คำอธิบาย

1. กำหนดการเล่นอนิเมชัน
2. กำหนดการชนกับ enemy
3. กำหนดการเก็บ carrot

(รูปที่ 107 event และ behavior ของ player)

Enemy

slime	move = 0	slime	Simulate Platform pressing Left
slime		slime	Set Not mirrored
slime	move = 1	slime	Simulate Platform pressing Right
slime		slime	Set Mirrored
slime	Platform is moving	System	Wait 2 seconds
slime	move = 0	slime	Set move to 1
slime	move = 1	System	Wait 2 seconds
slime		slime	Set move to 0

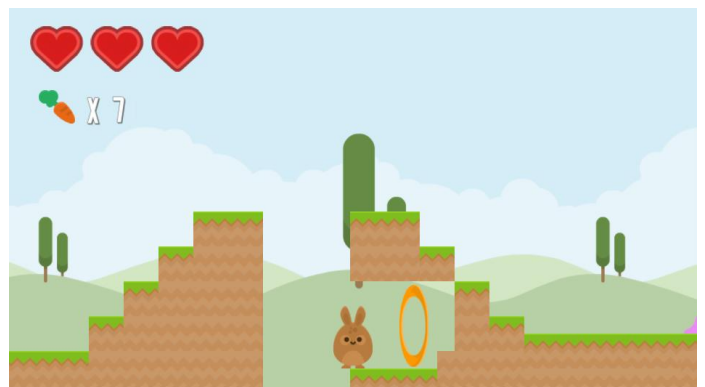
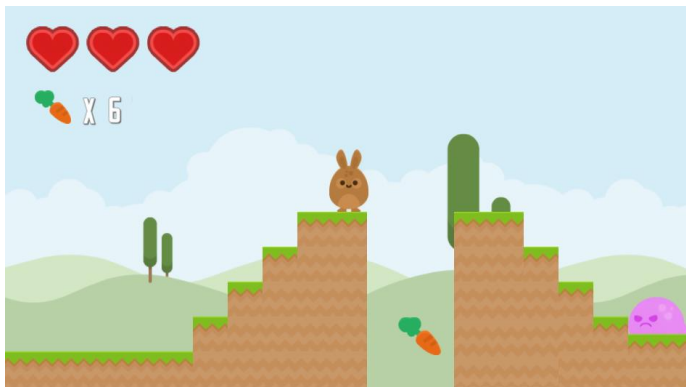
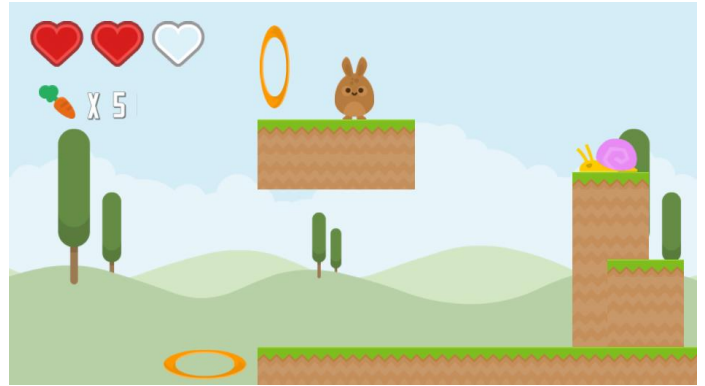
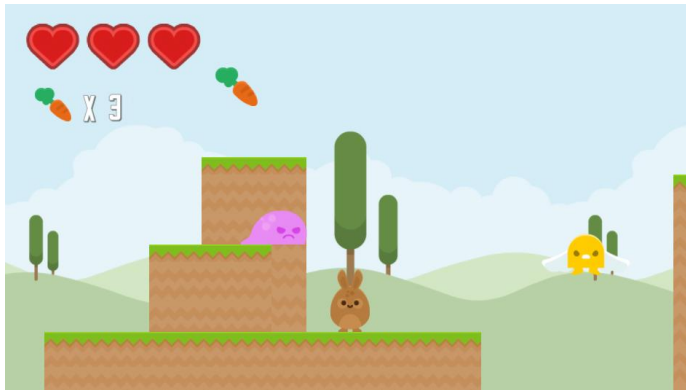
Behaviors

Fade [enemy]	Active at start	No
	Fade in time	0
	Wait time	0
	Fade out time	0.5
	Destroy	After fade out
Platform	Max speed	110
	Acceleration	1500
	Deceleration	1500
	Jump strength	650
	Gravity	1500
	Max fall spe...	1000
	Double jump	Disabled
	Jump sustain	0
	Default contr...	No
Effects	Blend mode	Normal

ปิด Default control

(รูปที่ 108 event และ behavior ของ enemy)

5. กำหนด level design ของเกมประเภท Platform นั้นจะเน้นไปที่สร้างแผนที่, ด้าน และการจัดวาง ตำแหน่งและความสามารถของ ศัตรู



(รูปที่ 109 ตัวอย่าง level design)