Part 17 - Timeline Tracks

We could set up a spawner to create a load of monsters for us, but we'll use a timeline instead. A timeline can issue commands.

Our timeline will simply issue the same create object command over and over in a loop.

Before we do this, remove the single monster from the Scene. We don't need it any more:

```
[Scene]
ChildList = PlatformObject # MiddlePlatformObject #
TopLeftPlatformObject # TopPlatformObject #
TopRightPlatformObject #
StarObject
```

Now to create a simple track:

```
[MonsterMakerTrack]
1    = Object.Create MonsterObject
Loop = True
```

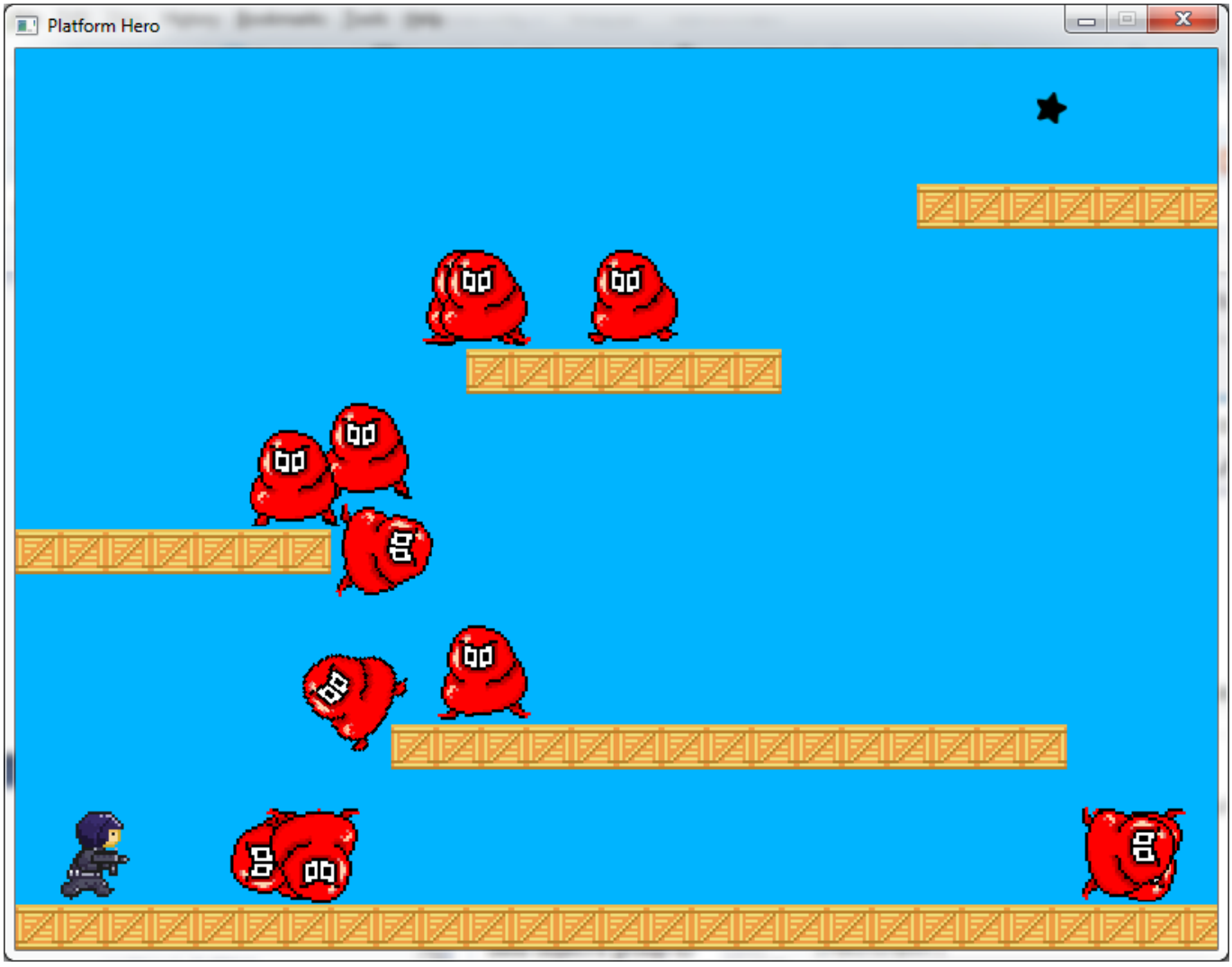
Attach the track to the Scene object so that Monster objects are created over and over:

```
[Scene]
ChildList = PlatformObject # MiddlePlatformObject #
TopLeftPlatformObject # TopPlatformObject #
TopRightPlatformObject #
StarObject
TrackList = MonsterMakerTrack
```

So that the monster objects start at a random position each time. Change the monster object to have a range of starting x positions:

```
[MonsterObject]
Graphic      = MonsterGraphic
AnimationSet = MonsterAnimationSet
Position     = (-380, -300, 0) ~ (200, -200, 0)
Scale        = 2.0
Body         = MonsterBody
```

Looking great! Monsters should be dropping in all over the place:



Some tweaks can be added to the monster and the body to improve things a little:

```
[MonsterObject]
Graphic      = MonsterGraphic
AnimationSet = MonsterAnimationSet
Position     = (-380, -300, 0) ~ (200, -200, 0)
Speed        = (-20, 0, 0) ~ (20, 0, 0)
Scale        = 2.0
Body         = MonsterBody

[MonsterBody]
Dynamic           = true
PartList          = MonsterBodyPart
AngularDamping    = 50
LinearDamping     = 0.2

[MonsterBodyPart]
Type        = box
Solid       = true
SelfFlags   = monster
CheckMask   = hero # platforms # bullet
Friction    = 0
```

```
Restitution = 0.2
Density     = 20
```

The Speed on the object will give the monsters a little random left/right movement. The `Friction` on the bodypart will make the monster less slippery on the ground.

The `LinearDamping` on the body will slow him down a little over time if he's too fast. The high `AngularDamping` will ensure the monster tips over the edge but not rotate and tumble wildly.

Finally, a touch of `Restitution` on the body will allow it to bounce just a touch when landing from a height.

That should work a little better. Additionally, if you prefer the Jelly Monsters not too rotate when tipping over the edges of the platforms you can add a `FixedRotation` to the body:

```
[MonsterBody]
Dynamic             = true
PartList            = MonsterBodyPart
AngularDamping      = 50
LinearDamping       = 0.2
FixedRotation       = true
```

Next: Part 18 – Exploding Monsters.

- Part 1 – Downloading Orx
- Part 2 – How Orx works
- Part 3 – Setting up a new game project
- Part 4 – A tour of an Orx project
- Part 5 – Viewport and the camera
- Part 6 – Objects
- Part 7 – Spritesheets and Animation
- Part 8 – Platforms and Texture Repeating
- Part 9 – Physics
- Part 10 – Input Controls
- Part 11 – Running and Standing
- Part 12 – Changing Direction
- Part 13 – Getting our hero to shoot
- Part 14 – FX
- Part 15 – Collision Events.
- Part 16 – Jelly Monsters
- Part 17 – Timeline Tracks
- Part 18 – Exploding Monsters
- Part 19 – The Hero's survival.
- Part 20 – Text and Game Over

From:
https://www.orx-project.org/wiki/ - **Orx Learning**

Permanent link:
https://www.orx-project.org/wiki/en/guides/beginners/timeline_tracks

Last update: **2025/09/30 17:26 (14 months ago)**