


# Mouse Party Offline Documentation

[www.chocolatedinosaur.com](http://www.chocolatedinosaur.com)

1 February 2025



# Table of contents:

- [About Mouse Party](#)
  - [Introduction](#)
  - [Features](#)
  - [Requirements](#)
  - [Trial Version](#) 
- [Download](#)
  - [Download From](#)
  - [Customer Reviews](#)
  - [Release Notes](#)
- [Scripting](#)
  - [Code Snippets](#)
  - [Examples](#)
- [Support](#)
  - [Product Support](#)



# About Mouse Party

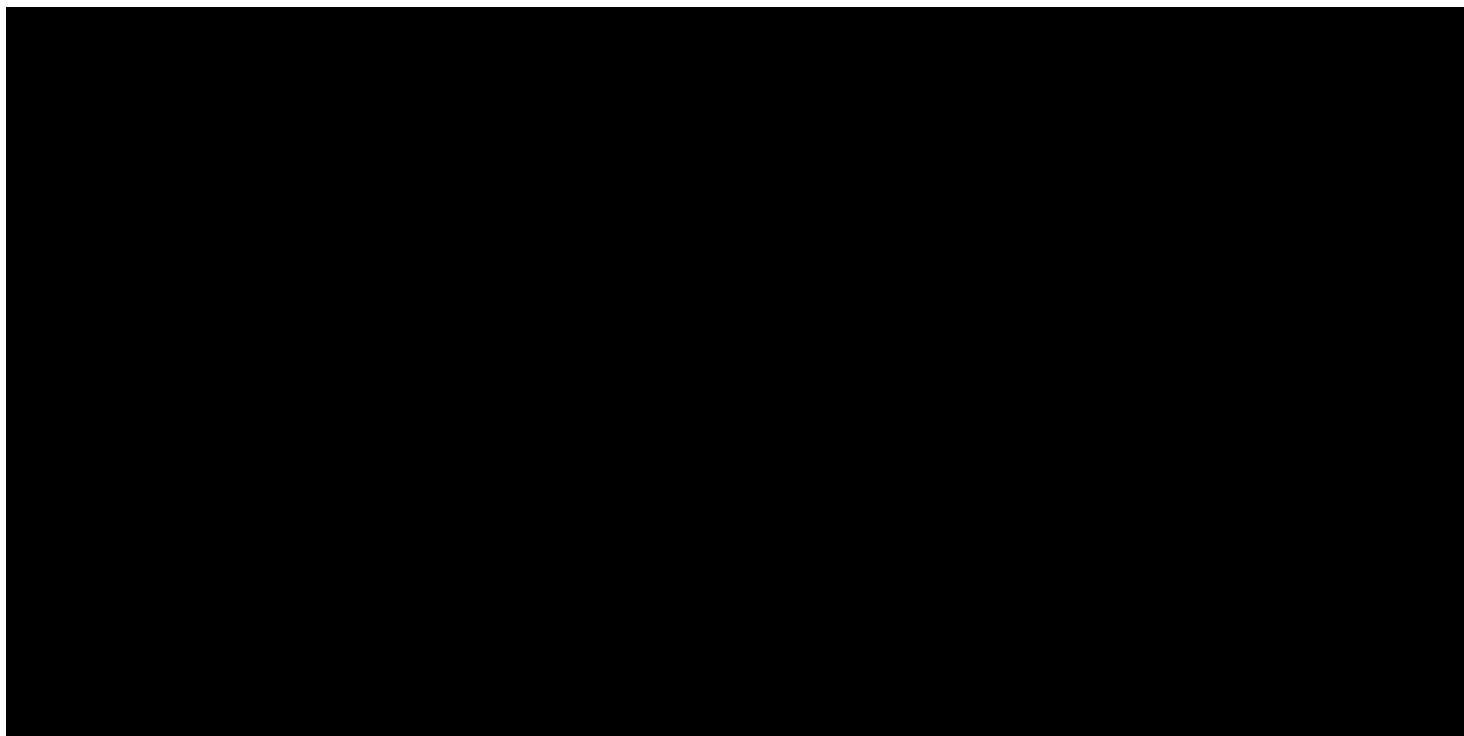
## Introduction

`Mouse Party` is a Unity plugin that enables games (and apps) to support multi-player input from multiple mice connected to the same PC.

Each player can have their own mouse cursor allowing for novel game play experiences!

We developed this plugin for our own internal projects and thought others would also find it useful. Initially we tried to use some existing solutions, but we decided to create our own as the others were either deprecated, no longer offered support and weren't actively developed.

Have fun making interesting multi-player games and invite your friends over for a `Mouse Party`!



## Features

- Create fun and unique local multi-player games!
- Supports over 200 mice connected to the same PC
- Handles dynamic mouse connection/disconnection
- Full C# source code included
- Works in editor as well as builds
- Very easy to use
- Well documented & supported
- Actively maintained

## Requirements

- Windows only (32-bit & 64-bit - Windows 8, 10 & 11)
- Supports all Unity versions from 2019.4.0 up to Unity 6.x
- Supports all render pipelines: Built-in, URP and HDRP

## Trial Version

Available soon!



# Download

Latest version: Mouse Party v1.0.0 (3 February 2025) >> [Release Notes](#).

## Download From

- [The Chocolate DinoStore](#) ✨
- [The Unity Asset Store](#)

### ▼ Why use the DinoStore? ✨

We built the DinoStore to give our customers the best purchasing and after-sales experience. Benefits include:

- Support the developer directly.
- Lowest price!
- History - access to all previously released versions.
- Earliest access to new version releases (no need to wait for Unity Asset Store review process).
- Don't worry, it's legit - our store is built using [LemonSqueezy](#) and Chocolate Dinosaur Ltd is a registered UK company.

## Customer Reviews

Once you have tried this asset, we would really appreciate your rating / review on the [Unity Asset Store](#).

## Release Notes

- ▶ **3 February 2025 - Version 1.0.0**



# Scripting

## Code Snippets

The namespace

```
1 using ChocDino.PartyIO;
```

## Examples

Example of low-level enumeration of all mice

```
1 using System.Collections.Generic;
2 using UnityEngine;
3 using ChocDino.PartyIO;
4
5 public class Example : MonoBehaviour
6 {
7     private MouseManager _manager;
8
9     void Awake()
10    {
11        MouseManager.ChangedConnectionState += OnMouseConnectionChanged;
12        _manager = new MouseManager();
13    }
14
15    void Update()
16    {
17        _manager.Update();
18        LogMiceState();
19    }
20
21    void LogMiceState()
```

```
22     {
23         List<Mouse> mice = _manager.All;
24         foreach (var mouse in mice)
25         {
26             string text = string.Format("Id: {0} Connection: {1} Position: {2}
Scroll; {3}", mouse.DeviceId, mouse.ConnectionState, mouse.PositionDelta,
mouse.ScrollDelta);
27             Debug.Log(text);
28
29             if (mouse.IsPressed(MouseButton.Left))
30             {
31                 Debug.Log("Left button is pressed");
32             }
33             if (mouse.WasPressedThisFrame(MouseButton.Left))
34             {
35                 Debug.Log("Left button was pressed this frame");
36             }
37             if (mouse.WasReleasedThisFrame(MouseButton.Left))
38             {
39                 Debug.Log("Left button was released this frame");
40             }
41         }
42     }
43
44     void OnDestroy()
45     {
46         _manager.Dispose();
47         _manager = null;
48     }
49
50     void OnMouseConnectionChanged(Mouse mouse)
51     {
52         if (mouse.ConnectionState == MouseConnectionState.Connected)
53         {
54             Debug.Log("Mouse with id " + mouse.DeviceId + " just connected.");
55         }
56         else if (mouse.ConnectionState == MouseConnectionState.Disconnected)
57         {
58             Debug.Log("Mouse with id " + mouse.DeviceId + " just
disconnected.");
59         }
60     }
61 }
```

# Support

We believe that `software is only as good as level of support and maintenance provided`.

If you're having a problem with one of our products or have a question, please get in touch via one of the methods below:

## Product Support

 [Discord Community](#)

 [GitHub Issues/Bugs](#)

 [Email support@chocdino.com](mailto:support@chocdino.com)