

# USB Midi Android Plugin

---

USB Midi Android Plugin is a plugin for communicate with a Midi device through USB.

## Table of Contents

---

1. [Create a new Script](#)
2. [Implements all inherited methods](#)
3. [Add it to editor](#)

## Unity Android

---

### Create a new Script

---

Create a Csharp script `MidiEventHandler` that inherits from `MonoBehaviour` and `IMidiEventHandler`. In the awake and start methods add these lines for instantiating the plugin :

```
private void Awake()
{
    gameObject.AddComponent<MidiManager>();
}

private void Start()
{
    MidiManager.Instance.RegisterEventHandler(this);
}
```

### Implements all inherited methods

---

Then implements all methods of `IMidiEventHandler` :

```
[SerializeField] private Text text;
// Called when you plug a midi note is down
public void NoteOn(int note, int velocity)
{
    Debug.Log("Note On " + note + " velocity " + velocity);
    text.text += "Note On " + note + " velocity " + velocity + Environment.NewLine;
}

// Called when you plug a midi note is released
public void NoteOff(int note)
{
    Debug.Log("Note off " + note);
    text.text += "Note off " + note + Environment.NewLine;
}

// Called when you plug a midi device
public void DeviceAttached(string deviceName)
{
    Debug.Log("Device Attached " + deviceName);
    text.text += "Device Attached " + deviceName + Environment.NewLine;
}

// Called when you unplug a midi device
public void DeviceDetached(string deviceName)
{
    Debug.Log("Device Detached " + deviceName);
    text.text += "Device Detached " + deviceName + Environment.NewLine;
}
```

## Add it to editor

---

In the editor, add a new gameObject and add it the `MidiEventHandler` . Then add a Text and link it to the `MidiEventHandler` component. You should now be able to receive midi events.