



# PostProcess'y w Unity (LRP,URP,HDRP)

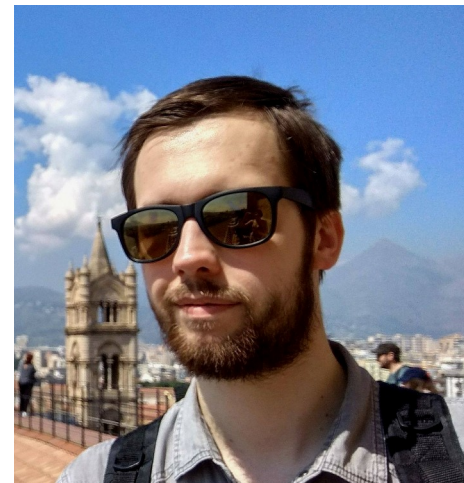
by &Jay na potrzeby Polygon'u

# O mnie

Andrzej '&Jay' Jezierski

4 lata w gamedev'ie

- ~~Unity Developer~~
- ~~Porting Developer~~
- **Technical Artist**



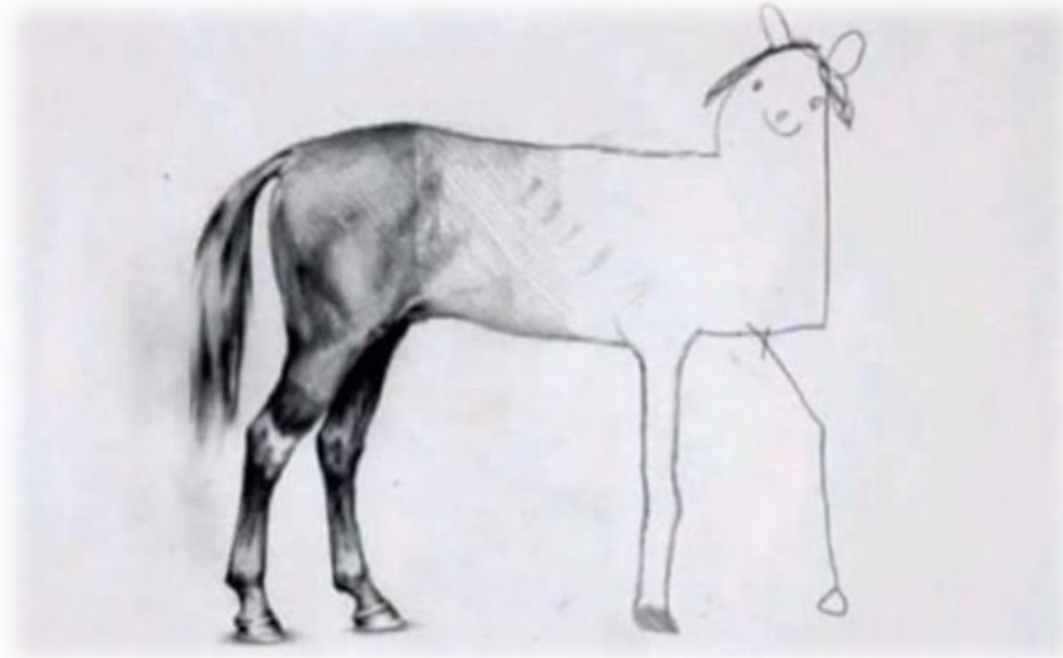


# Cel wykładu

- Przedstawienie w jaki sposób uruchomić postprocessy istniejące w środowisku Unity
- Przedstawienie w jaki sposób dodać własne postprocessy do Unity
- Przykłady obejmują następujące RenderPipeline'y Unity:  
LRP,URP/HDRP
- Przedstawione przykłady operują na wersji Unity 2021.3.11

# OSTRZEŻENIE 1!

Moja opinia o tej prezentacji



# OSTRZEŻENIE 2!

Techniczny bełkot



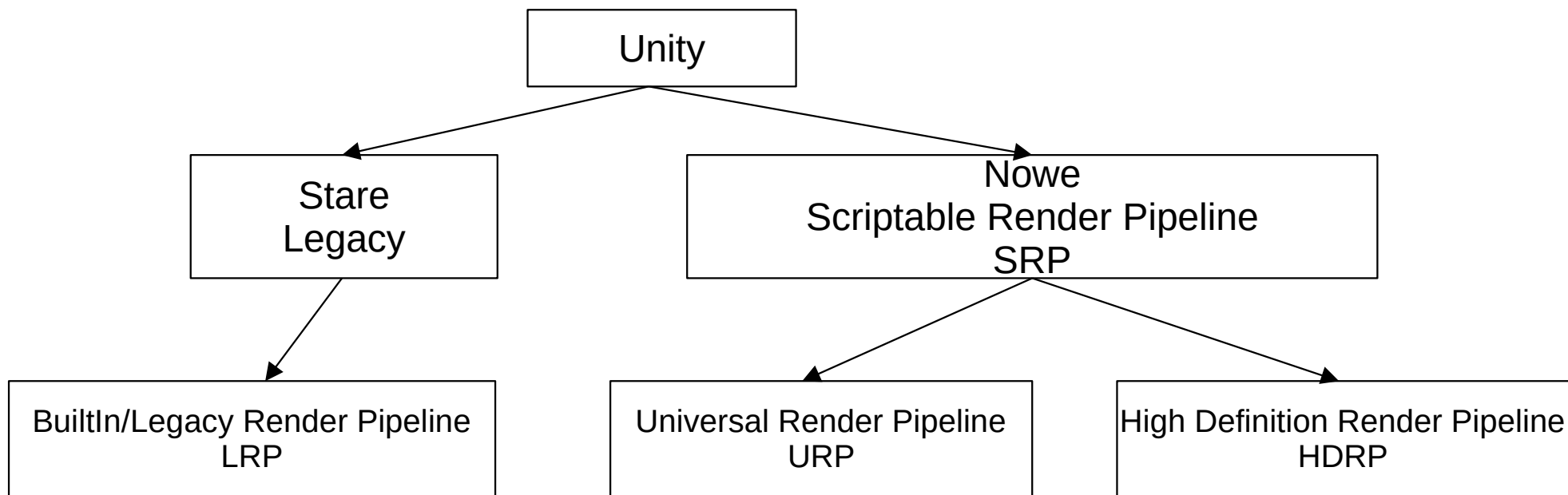


## OSTRZEŻENIE 3!

Część treści przedstawiona w prezentacji została opisana w sposób uproszczony i może różnić się w zakresie prawdziwych rozwiązań. Jednak nie powinny one mieć wpływu na poprawne korzystanie z przedstawionych narzędzi

# Co to jest RenderPipeline?

- Proces game engine'u przetwarzający obiekty 3D/2D na finalny obraz wyświetlany na ekranie





# Co to są PostProcess'y?

- **Technicznie:** Metody obróbki obrazu (“w czasie rzeczywistym”) w celu poprawy jego jakości wizualno-artystycznej
- **Normalnie:** Takie filtry z Tiktok'a





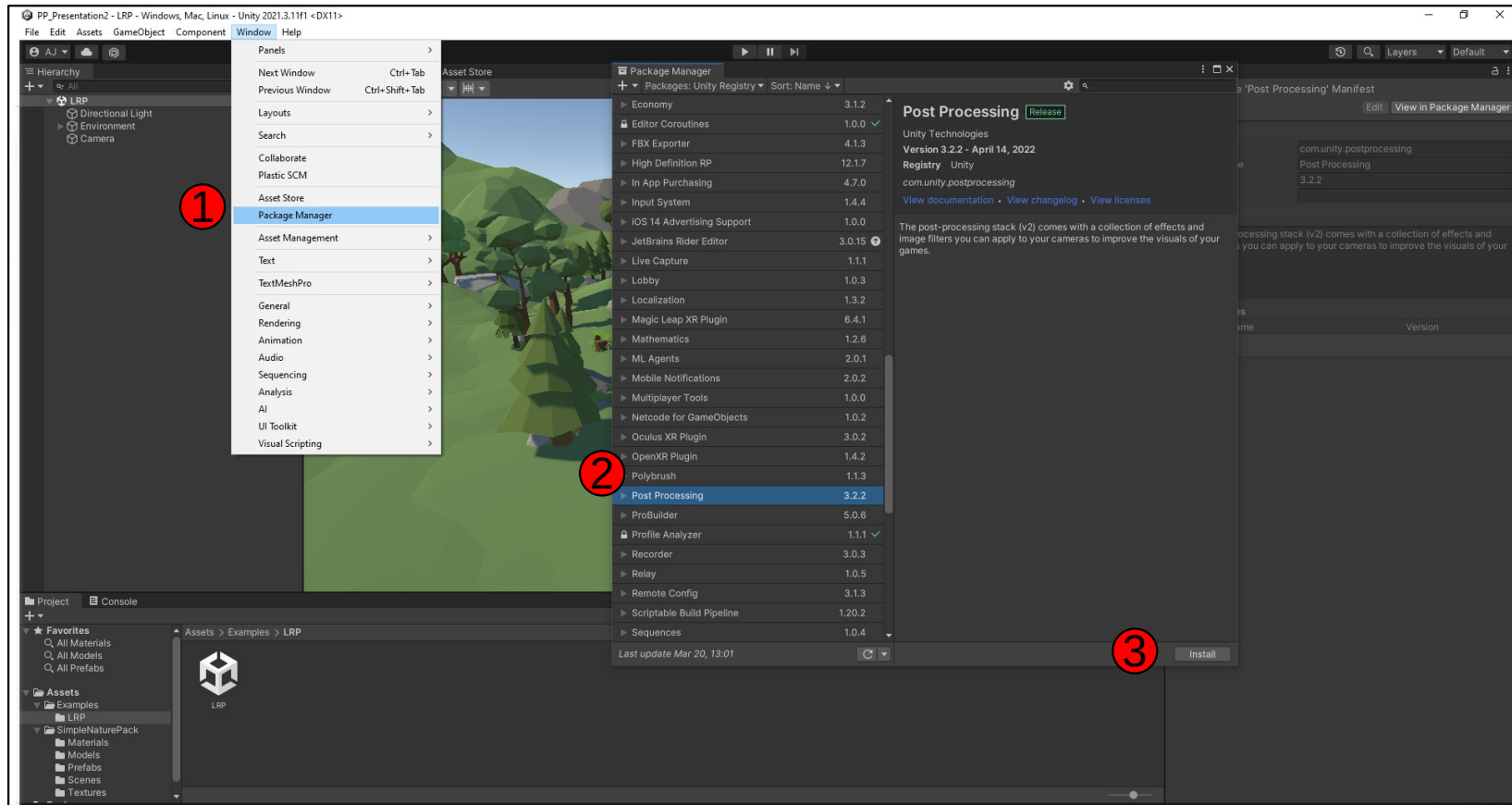
# Zalety i wady Postprocess'ów

- **Zalety:**
  - działają na wszystko na raz
- **Wady:**
  - działają na wszystko na raz



# **Uruchamianie PostProcess'ów w LRP**

# Uruchamianie PostProcess'ów w LRP – przygotowanie projektu

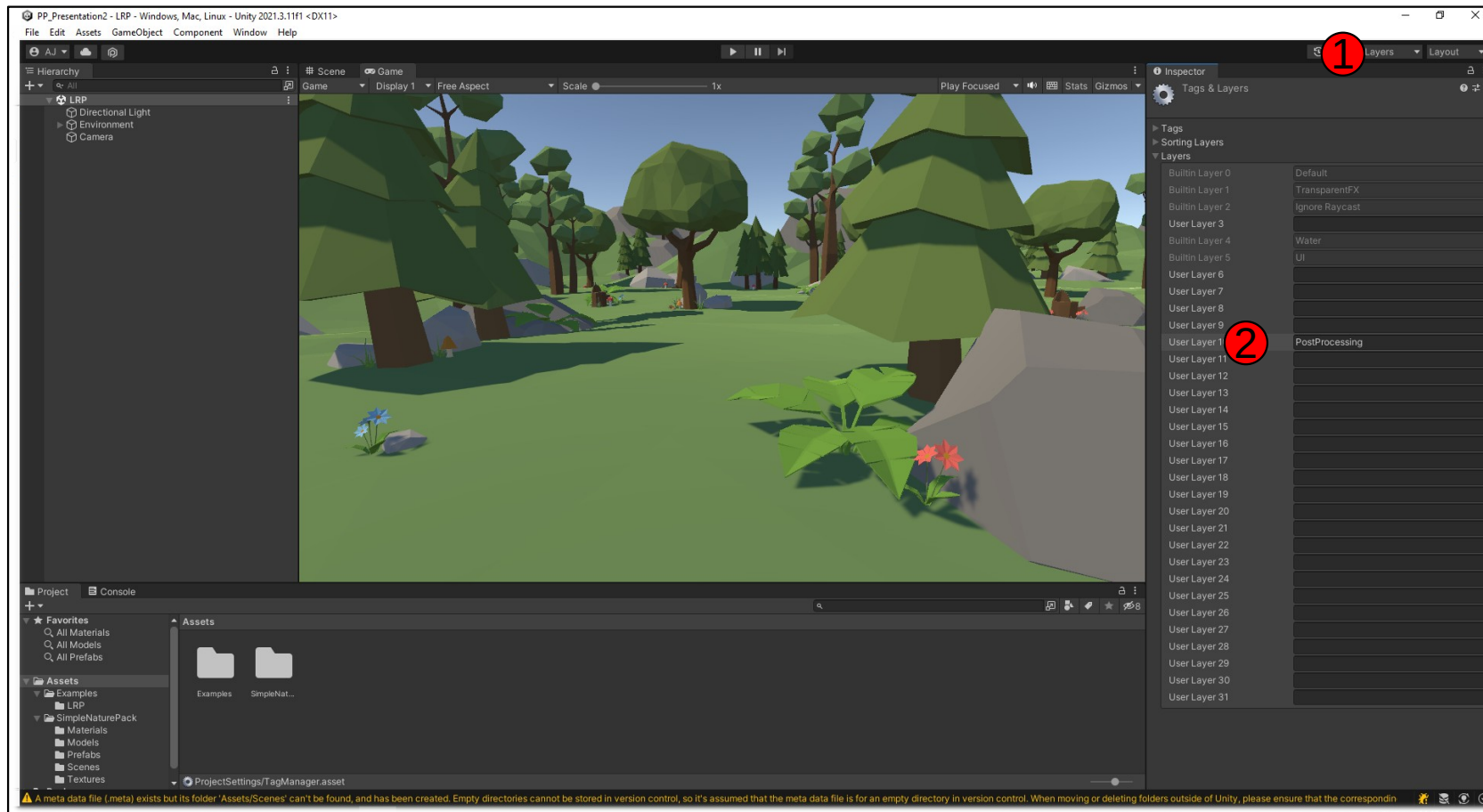




# Uruchamianie PostProcess'ów w LRP – przygotowanie projektu

- 1) Otwórz Package Manager
- 2) Wybierz package 'Post Processing'
- 3) Zainstaluj wybrany package. Dzięki temu elementy niezbędne do działania PostProcess'ów zostały dodane do projektu

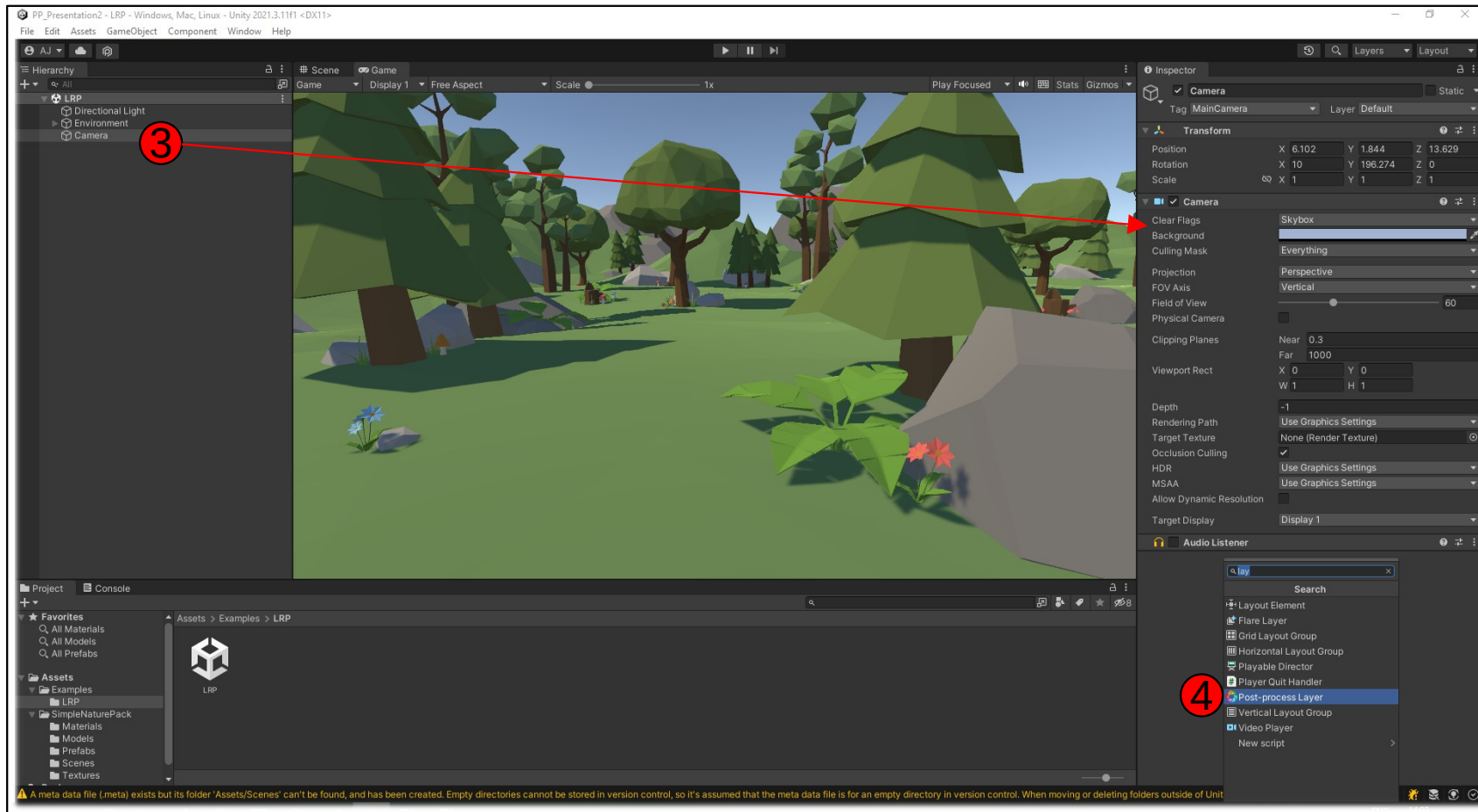
# Uruchamianie PostProcess'ów w LRP – przygotowanie projektu



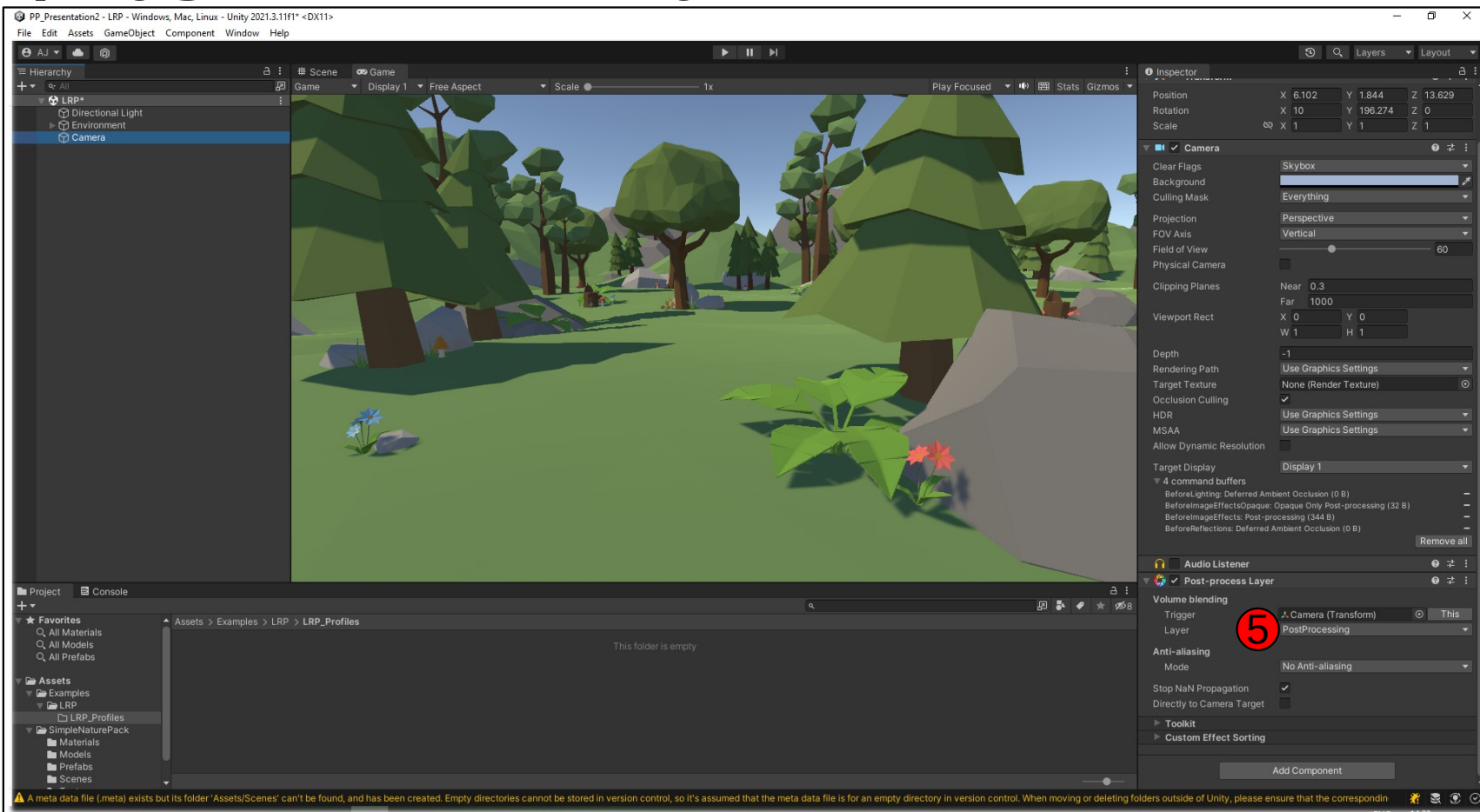
# Uruchamianie PostProcess'ów w LRP – przygotowanie projektu

- 1) Otwórz okno zarządzania Layer'ami
- 2) Dodaj nowy Layer dla PostProcess'ów (nie jest to do końca konieczne, ale pomaga w zarządzaniu i optymalizacji). W moim przykładzie mój nowy layer "PostProcessing", ale nazwa jest dowolna

# Uruchamianie PostProcess'ów w LRP – przygotowanie sceny



# Uruchamianie PostProcess'ów w LRP – przygotowanie sceny

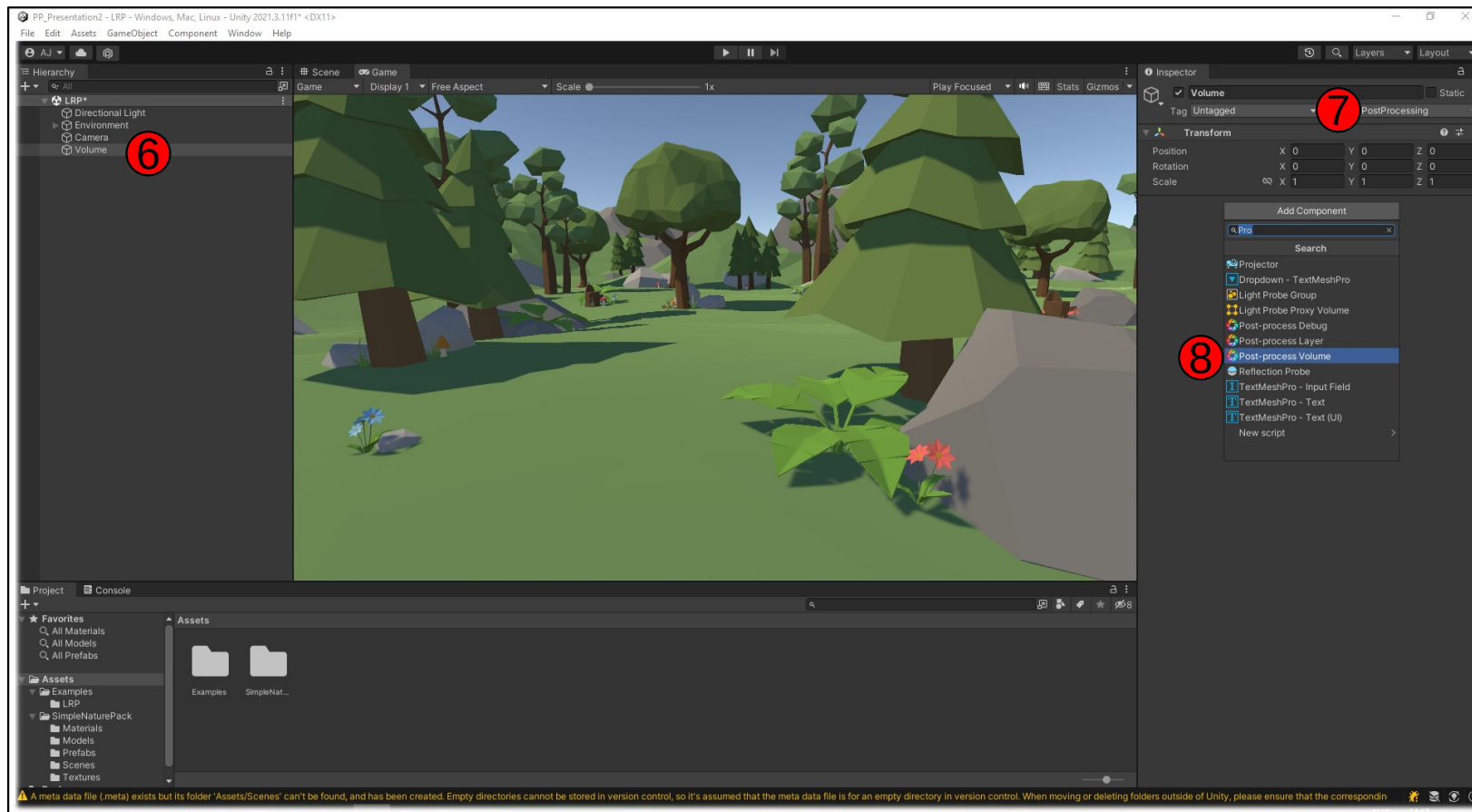




# Uruchamianie PostProcess'ów w LRP – przygotowanie sceny

- 3) Wybierz obiekt z component'em 'Camera' na scenie
- 4) Dodaj do wybranego obiektu component 'Post-process Layer'. Jest to component odpowiedzialny za realizację działania PP
- 5) W component'cie 'Post-process Layer' ustaw pole Layer, na nazwę layer'a utworzonego w kroku 2) (u mnie 'PostProcessing'). Dzięki temu system od PP będzie wiedział na których konkretnie layer'ach ma szukać obiektów z konfiguracją.

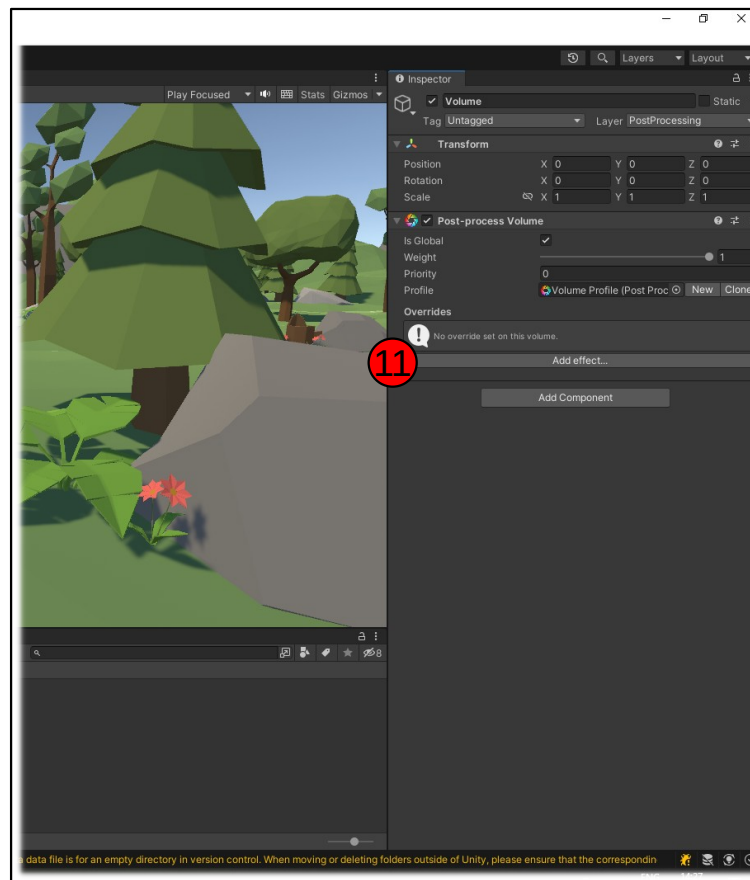
# Uruchamianie PostProcess'ów w LRP – przygotowanie sceny



# Uruchamianie PostProcess'ów w LRP – przygotowanie sceny

- 6) Stwórz pusty obiekt na scenie. Ja nazwałem go 'Volume'
- 7) Zmień Layer stworzonego obiektu na layer utworzony w kroku 2). Dzięki temu component z kroku 3) będzie w stanie go odnaleźć
- 8) Dodaj do stworzonego obiektu component 'Post-process Volume'. Jest to component odpowiedzialny za przechowywanie konfiguracji PP na scenie w LRP

# Uruchamianie PostProcess'ów w LRP – przygotowanie sceny



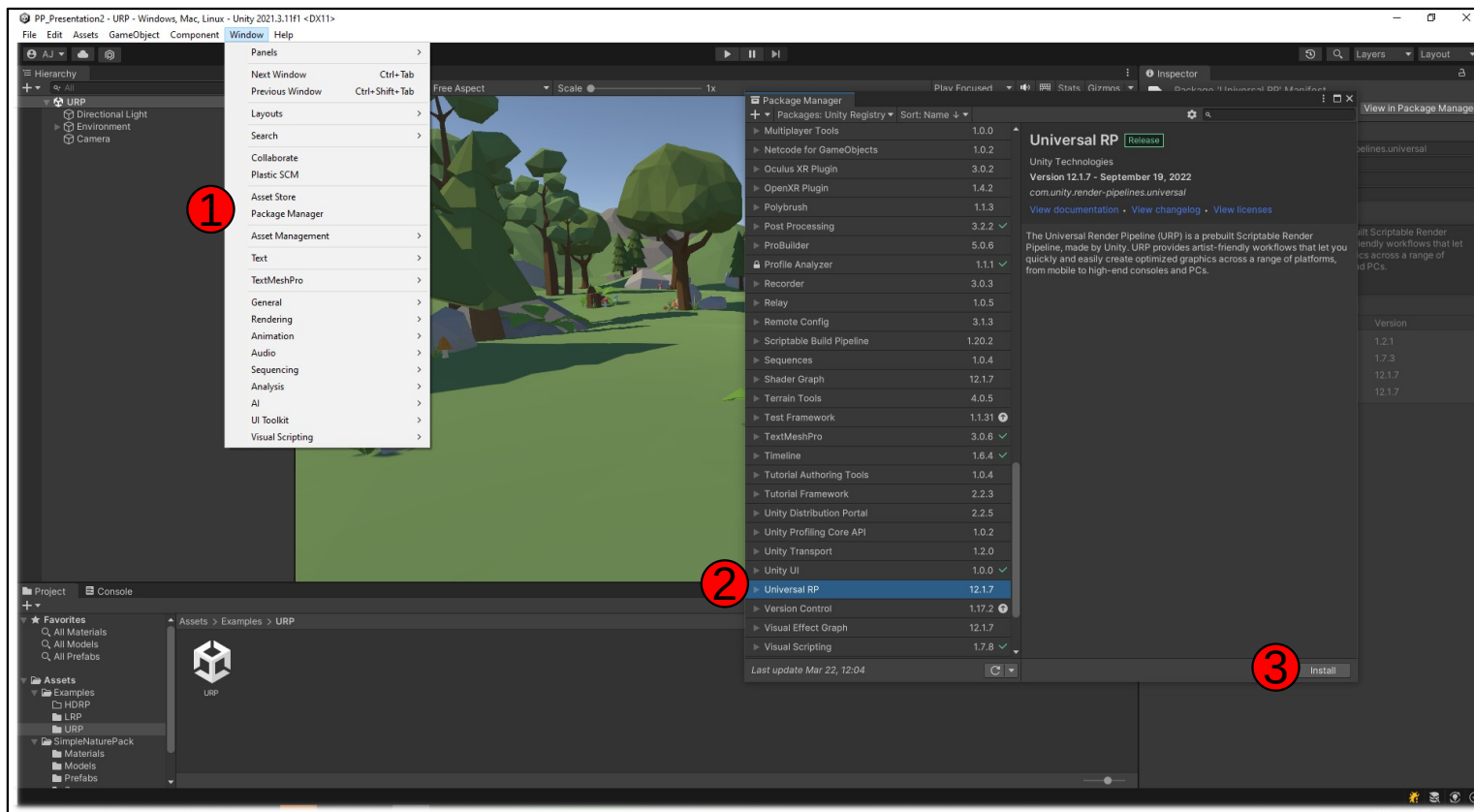
# Uruchamianie PostProcess'ów w LRP – przygotowanie sceny

- 9) Ustaw pole 'Is Global' component'u 'Post-process Volume' na aktywne. Dzięki temu aktywne w nim PP będą aktywne na całej scenie gry, niezależnie od położenia obiektu.
- 10) Kliknij przycisk 'New', aby stworzyć nowy obiekt z profilem konfiguracji PP. Obiekt pojawi się jako nowy plik w projekcie. Później można go bez problemu przenieść/zmienić nazwę dla utrzymania porządku.
- 11) Gdy profil jest podpięty do component'u 'Post-process Volume' możesz już dodawać nowe PP do profil'u i ubogacać swoją scenę :)



# **Uruchamianie PostProcess'ów w URP**

# Uruchamianie PostProcess'ów w URP – przygotowanie projektu



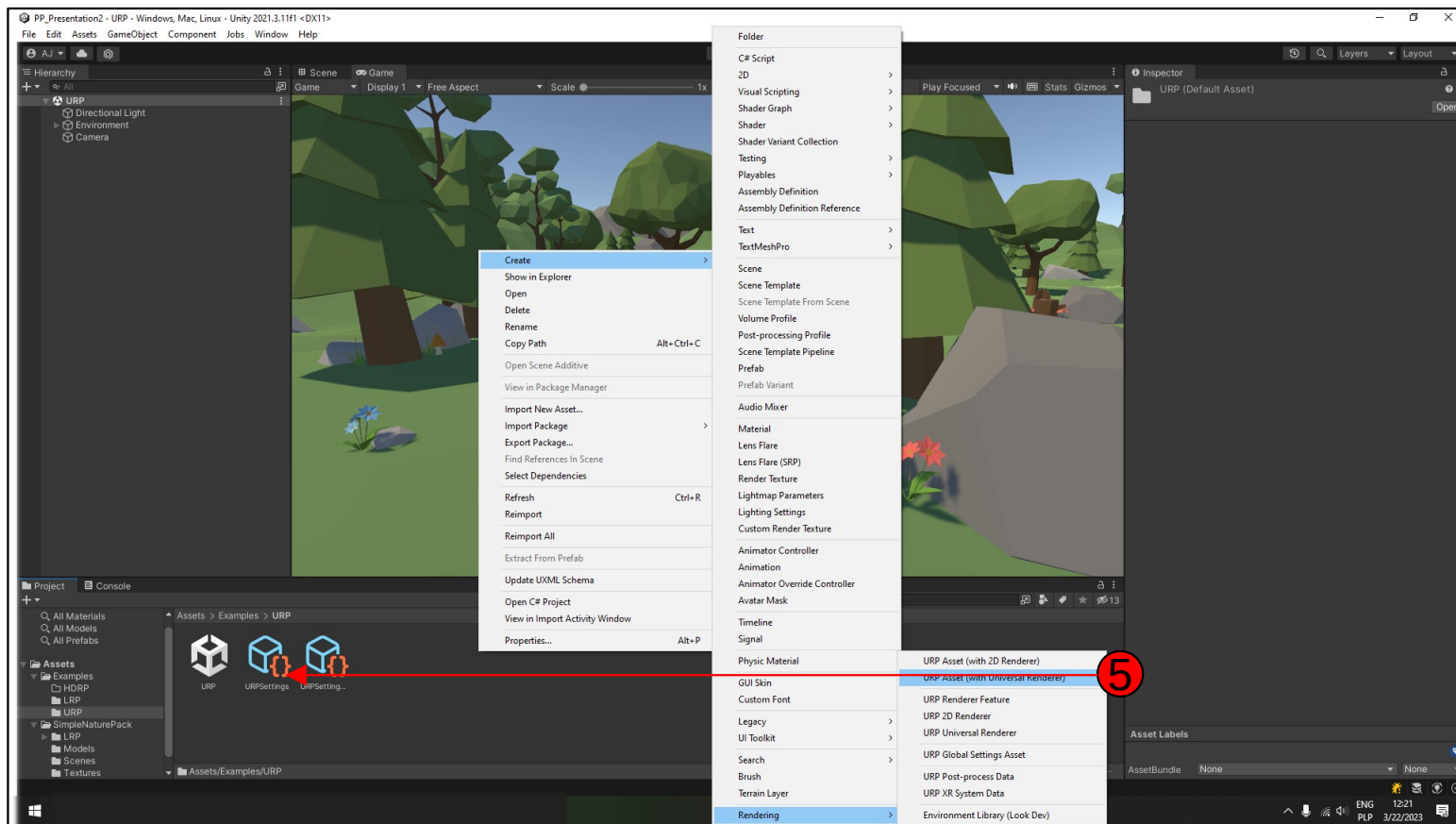


# Uruchamianie PostProcess'ów w URP – przygotowanie projektu

- 1) Otwórz Package Manager
- 2) Wybierz package 'Universal RP'
- 3) Zainstaluj wybrany package



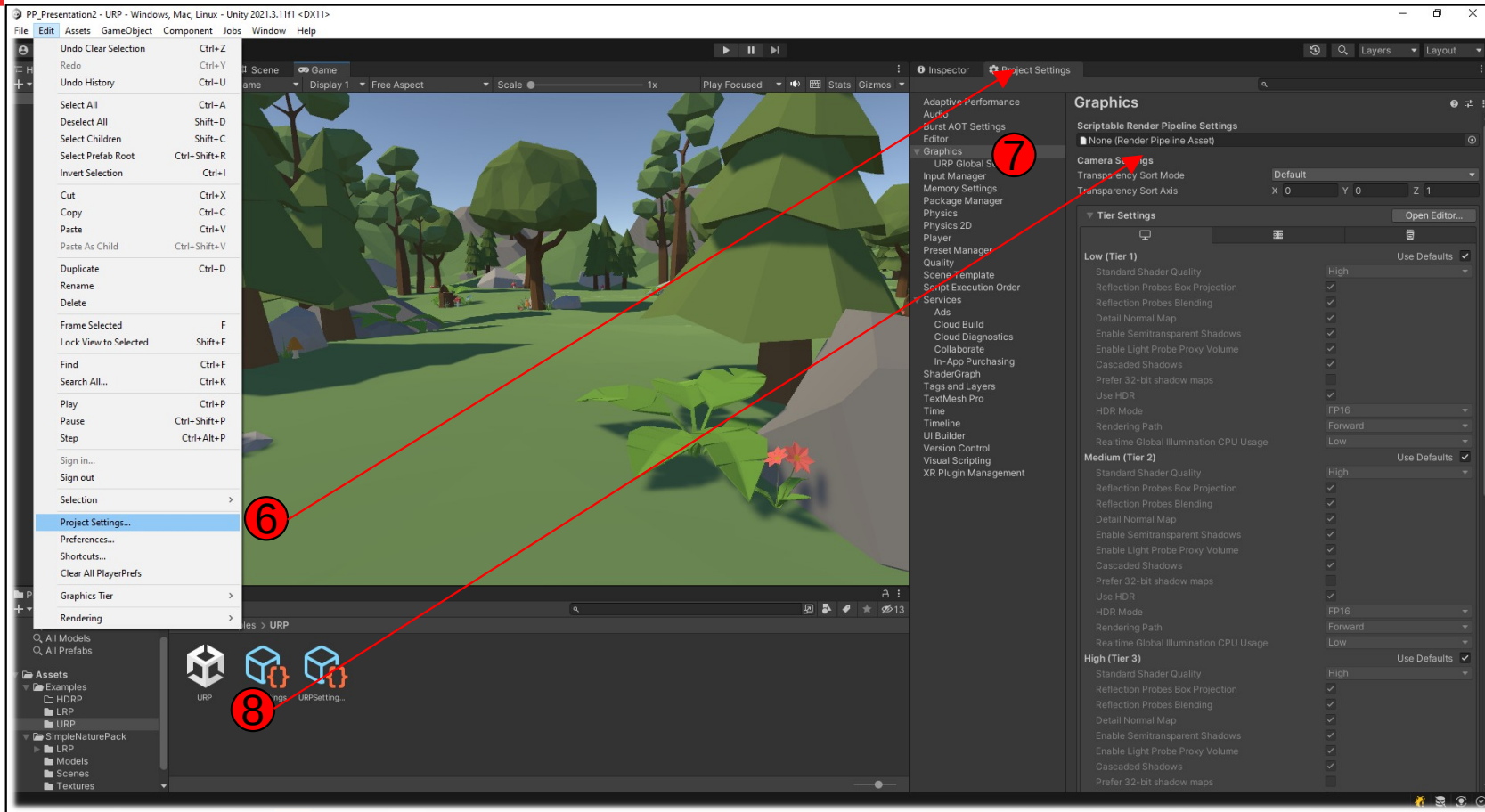
# Uruchamianie PostProcess'ów w URP – przygotowanie projektu



# Uruchamianie PostProcess'ów w URP – przygotowanie projektu

5) Klikamy PPM w polu projektu (czyli u mnie okno na dole) i wybieramy 'URP Asset (with Universal Renderer)'. Utworzy to dwa obiekty, które zawierają konfigurację URP. U siebie nazwałem je 'URPSettings' oraz 'URPSettings\_Renderer'

# Uruchamianie PostProcess'ów w URP – przygotowanie projektu



# Uruchamianie PostProcess'ów w URP – przygotowanie projektu

6) Otwórz panel 'Project Settings'

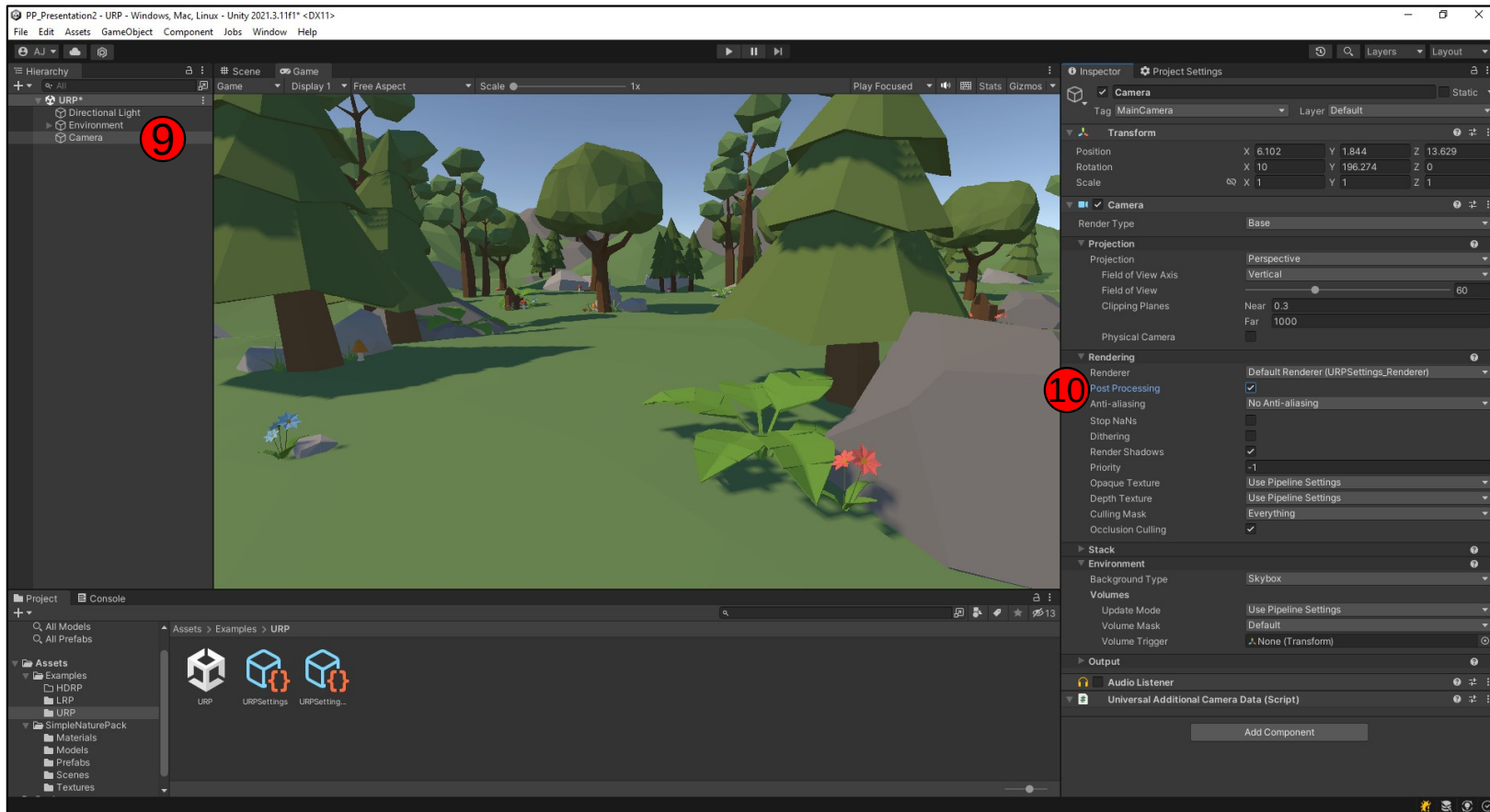
7) Otwórz zakładkę 'Graphics'

8) Podepnij utworzony w 5) plik z konfiguracją URP w polu 'Scriptable Render Pipeline Settings'. W tym momencie pojawi się ostrzeżenie, że zmiana Render Pipeline'a może chwilę zająć.

Zatwierdzamy

Info dodatkowe: Gdy pole 'Scriptable Render Pipeline Settings' jest puste Unity domyślnie działa w trybie LRP. Podpinając odpowiednie settingsy można aktywować URP lub HDRP

# Uruchamianie PostProcess'ów w URP – przygotowanie sceny

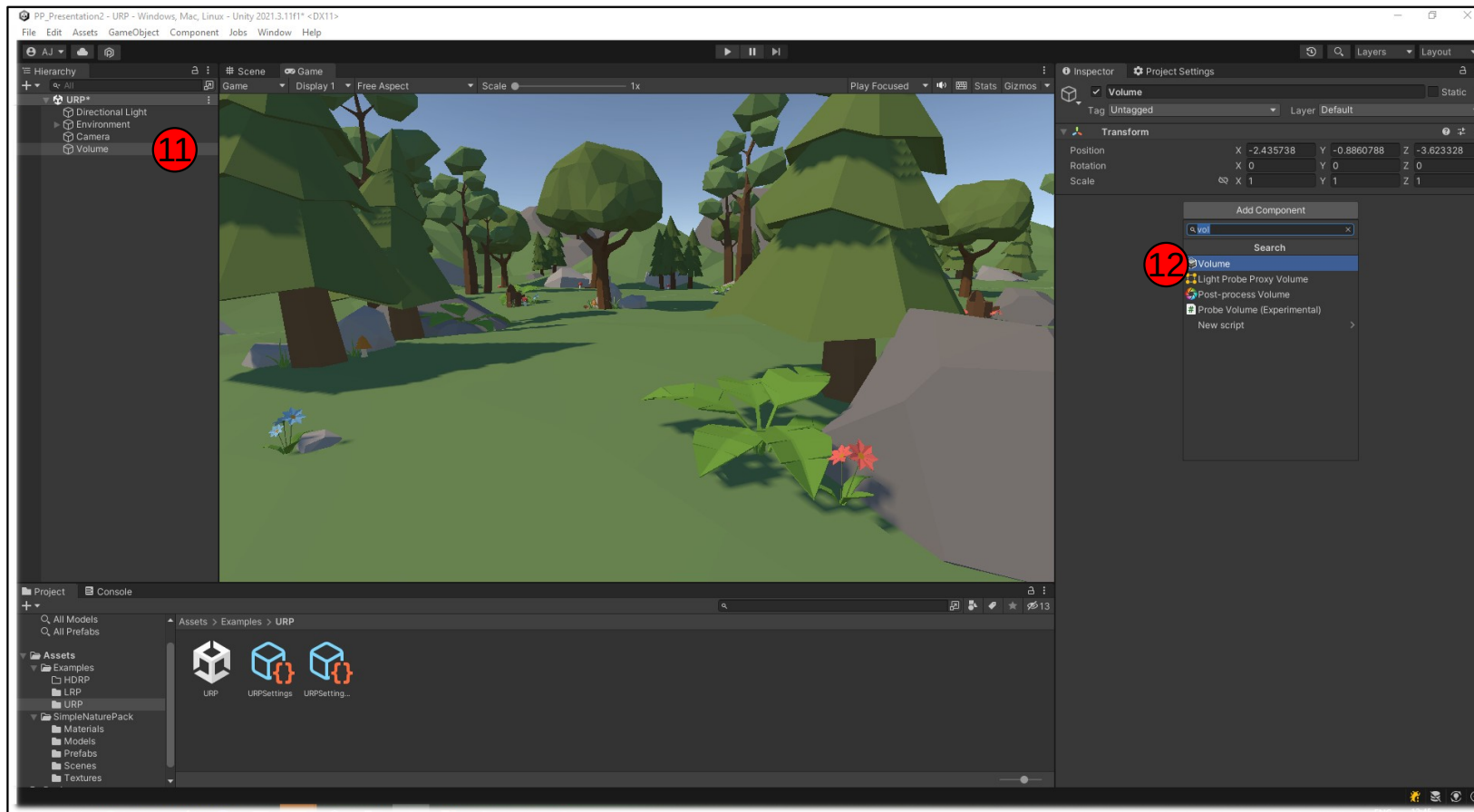




## Uruchamianie PostProcess'ów w URP – przygotowanie sceny

- 9) Wybierz obiekt z component'em 'Camera' na scenie
- 10) W component'cie 'Camera' aktywuj pole 'Post Processing'

# Uruchamianie PostProcess'ów w URP – przygotowanie sceny

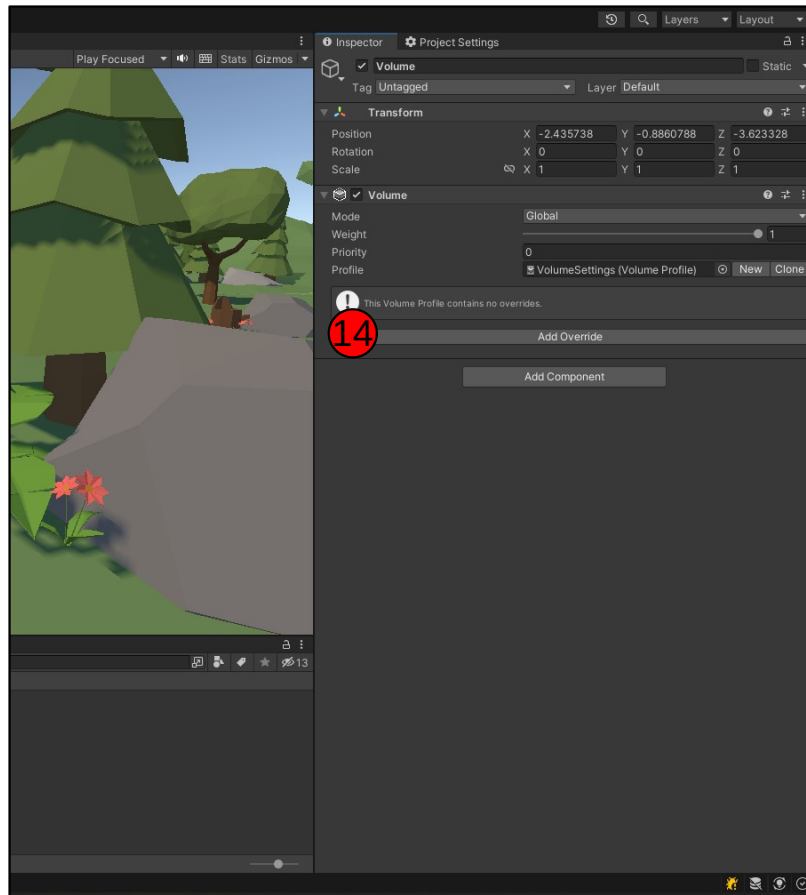
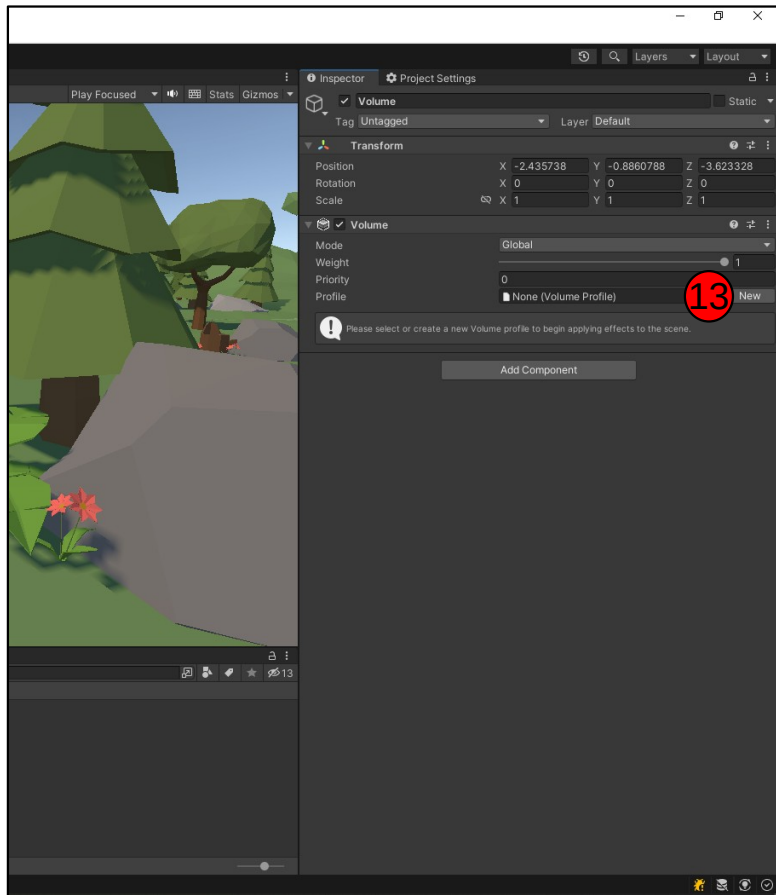


# Uruchamianie PostProcess'ów w URP – przygotowanie sceny

- 11) Stwórz pusty obiekt na scenie. Ja nazwałem go 'Volume'
- 12) Dodaj do stworzonego obiektu component 'Volume'. Jest to component odpowiedzialny za przechowywanie konfiguracji PP na scenie w URP



# Uruchamianie PostProcess'ów w URP – przygotowanie sceny



# Uruchamianie PostProcess'ów w URP – przygotowanie sceny

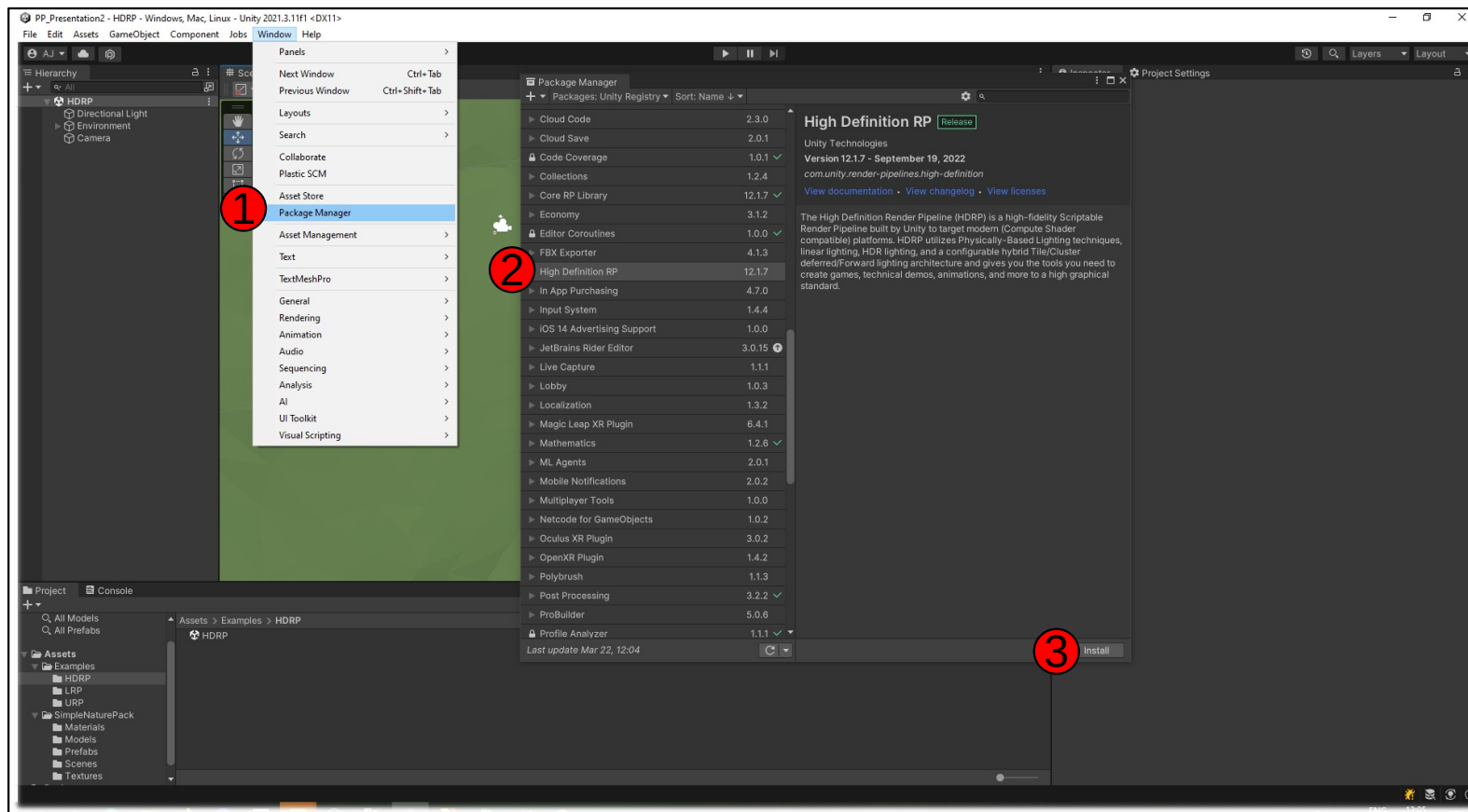
13) Kliknij przycisk 'New', aby stworzyć nowy obiekt z profilem konfiguracji PP. Obiekt pojawi się jako nowy plik w projekcie. Później można go bez problemu przenieść/zmienić nazwę dla utrzymania porządku.

14) Gdy profil jest podpięty do component'u 'Volume' możesz już dodawać nowe PP



# **Uruchamianie PostProcess'ów w HDRP**

# Uruchamianie PostProcess'ów w HDRP – przygotowanie projektu

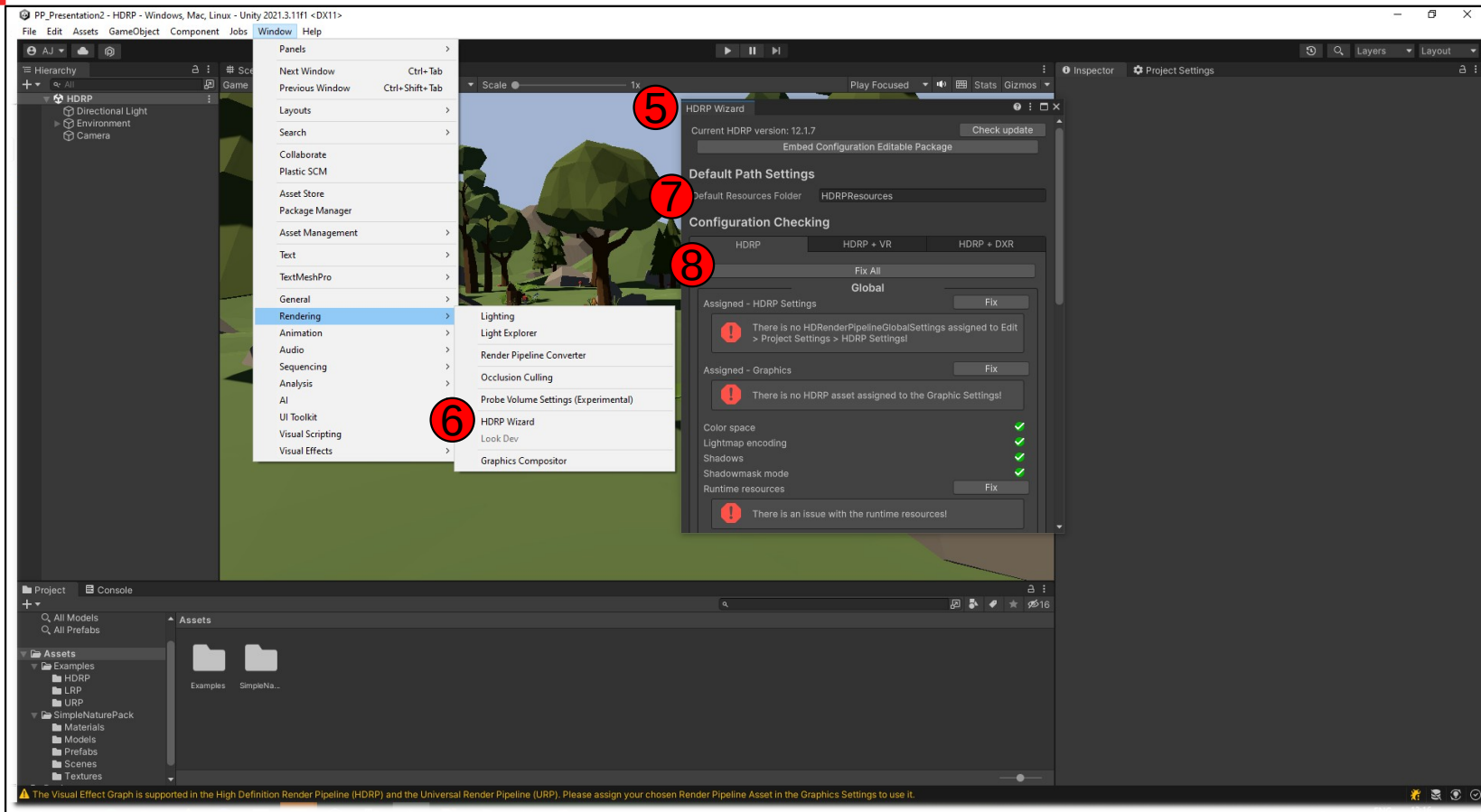




# Uruchamianie PostProcess'ów w HDRP – przygotowanie projektu

- 1) Otwórz Package Manager
- 2) Wybierz package 'High Definition RP'
- 3) Zainstaluj wybrany package

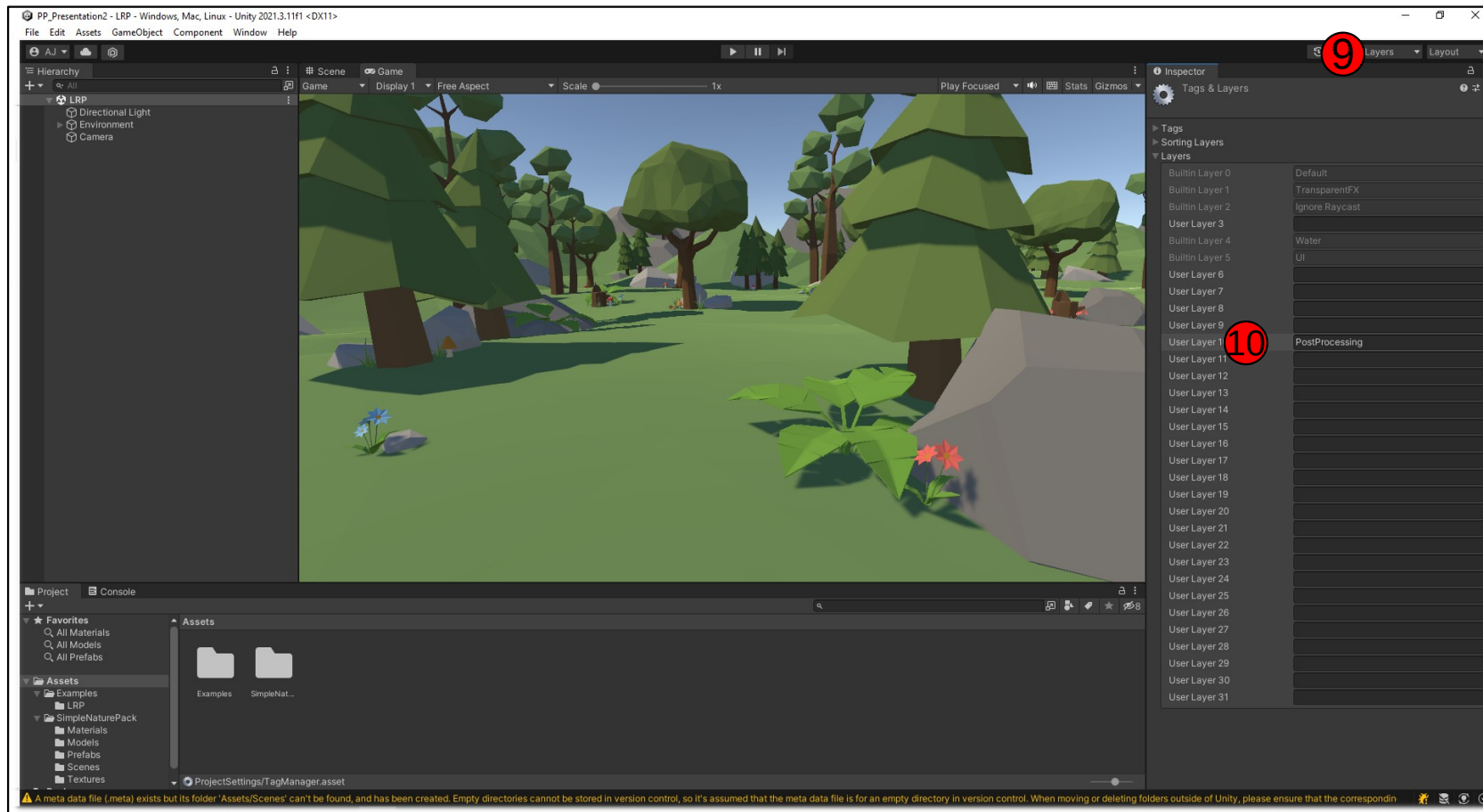
# Uruchamianie PostProcess'ów w HDRP – przygotowanie projektu



# Uruchamianie PostProcess'ów w HDRP – przygotowanie projektu

- 5) Po zainstalowaniu package'a automatycznie powinno otworzyć się okno 'HDRP Wizard'
- 6) Jeśli 'HDRP Wizard' nie otworzył się automatycznie, można go otworzyć z panelu 'Window'. Jeśli i tutaj jest wyszażony oznacza to prawdopodobnie, że masz aktualnie aktywny dowolny inny SRP i trzeba go wpierw wyłączyć. Pole ze Slajdu 26 punkt 8) musi być puste
- 7) Pole 'Default Resources Folder' wskazuje ścieżkę, gdzie zostaną utworzone wszystkie pliki z ustawieniami HDRP
- 8) W 'HDRP Wizard' klikamy 'Fix All'. Powinno to automatycznie przygotować projekt pod działanie HDRP. Pełna konfiguracja zostanie wrzucona do 7)

# Uruchamianie PostProcess'ów w HDRP – przygotowanie projektu





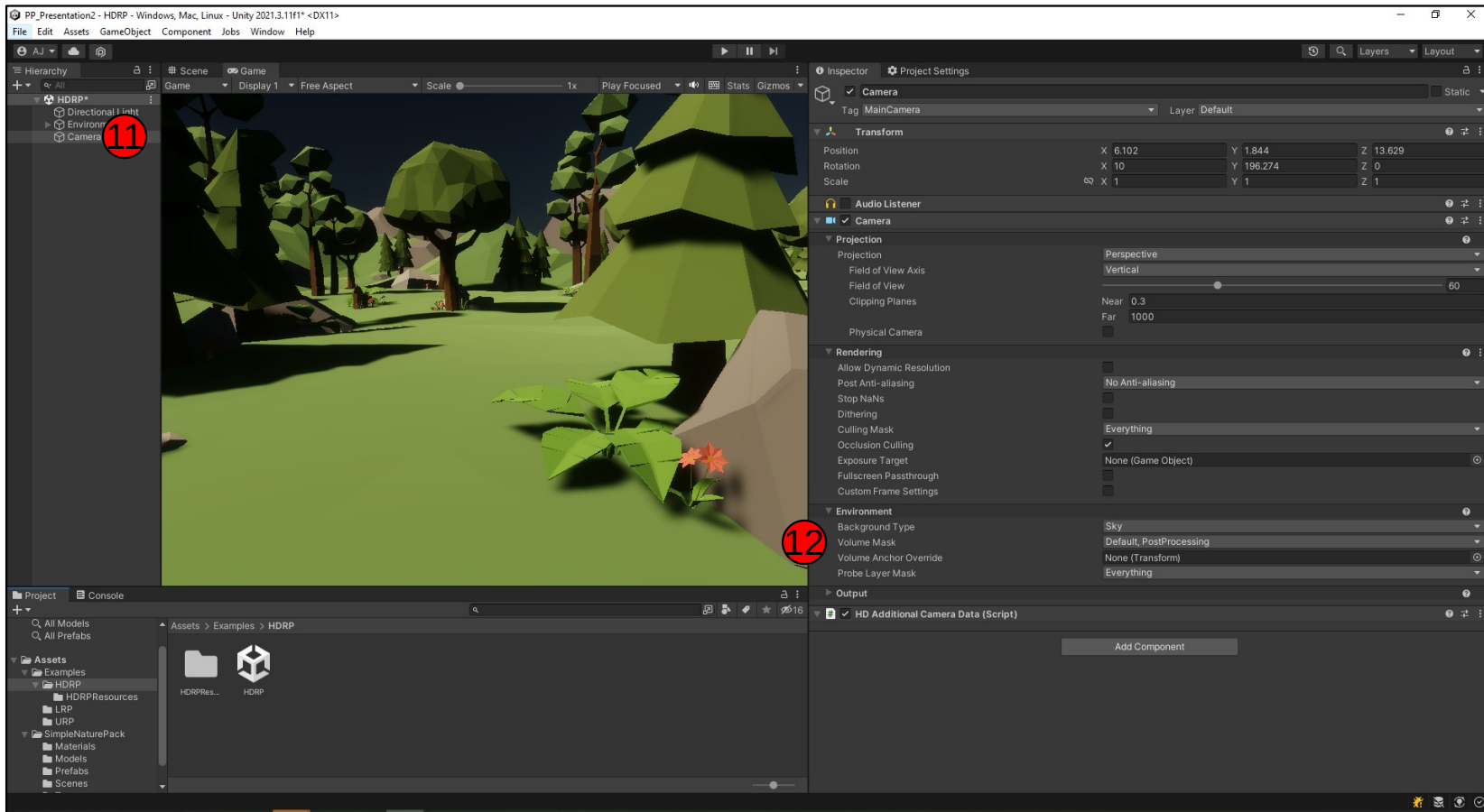
# Uruchamianie PostProcess'ów w HDRP – przygotowanie projektu

9) Otwórz okno zarządzania Layer'ami

10) Dodaj nowy Layer dla PostProcess'ów (nie jest to do końca konieczne, ale pomaga w zarządzaniu i optymalizacji). W moim przykładzie mój nowy layer "PostProcessing", ale nazwa jest dowolna

Komentarz: Tak samo jak w LRP

# Uruchamianie PostProcess'ów w HDRP – przygotowanie sceny



# Uruchamianie PostProcess'ów w HDRP – przygotowanie projektu

- 11) Wybierz obiekt z component'em 'Camera' na scenie
- 12) W component'cie 'Camera' w polu 'Volume Mask' dodaj stworzony przez siebie layer. Należy zostawić layer 'Default', ponieważ bez niego nie działa domyślny Global Volume z HDRP.

# Uruchamianie PostProcess'ów w HDRP – przygotowanie sceny

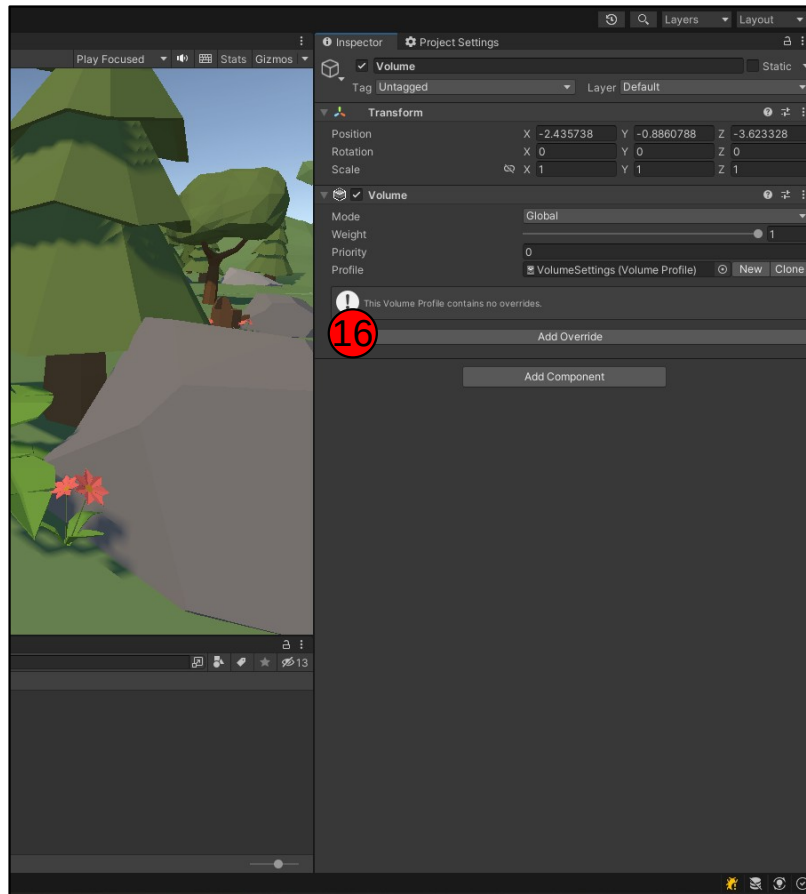
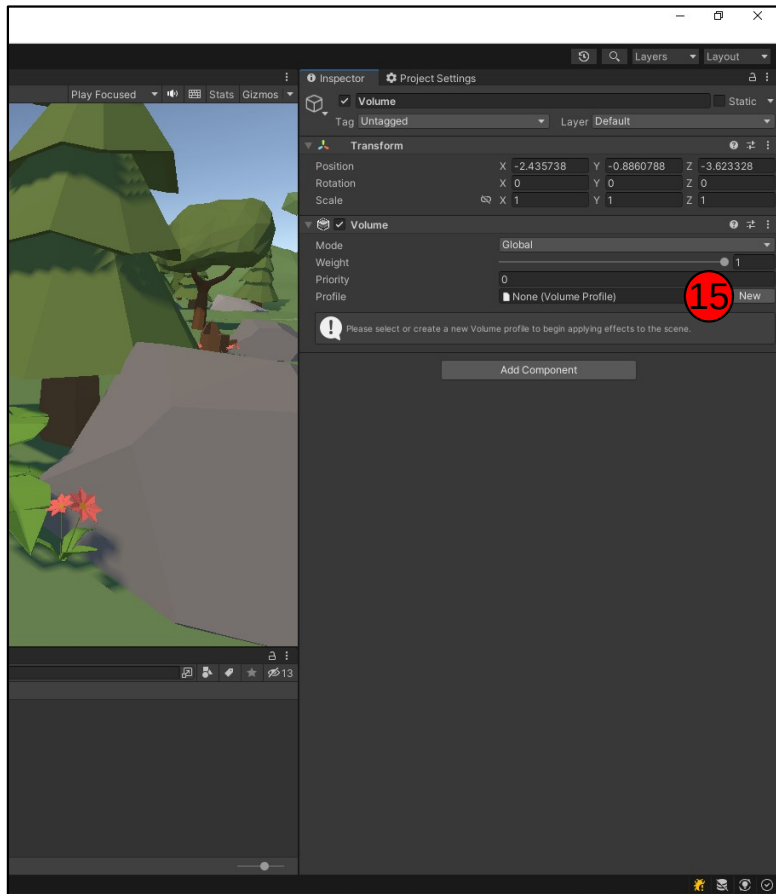


# Uruchamianie PostProcess'ów w HDRP – przygotowanie sceny

- 13) Stwórz pusty obiekt na scenie. Ja nazwałem go 'Volume'
- 14) Dodaj do stworzonego obiektu component 'Volume'. Jest to component odpowiedzialny za przechowywanie konfiguracji PP na scenie w HDRP

Komentarz: Tak samo jak w URP

# Uruchamianie PostProcess'ów w URP – przygotowanie sceny



# Uruchamianie PostProcess'ów w URP – przygotowanie sceny

15) Kliknij przycisk 'New', aby stworzyć nowy obiekt z profilem konfiguracji PP. Obiekt pojawi się jako nowy plik w projekcie. Później można go bez problemu przenieść/zmienić nazwę dla utrzymania porządku.

16) Gdy profil jest podpięty do component'u 'Volume' możesz już dodawać nowe PP

Komentarz: Tak samo jak w URP

# Dodawanie własnych PostProcess'ów

- LRP:  
<https://docs.unity3d.com/Packages/com.unity.postprocessing@2.1/manual/Writing-Custom-Effects.html>
- URP:  
wymaga ręcznego dopisania pewnych skryptów. Przykład przedstawiony w projekcie w załączonym link'u do git'a
- HDRP:  
<https://docs.unity3d.com/Packages/com.unity.render-pipelines.high-definition@12.0/manual/Custom-Post-Process.html>





# Git z projektem

Link: [https://github.com/AndJay7/PP\\_Presentation.git](https://github.com/AndJay7/PP_Presentation.git)

Uwaga: Aby zmienić RenderPipeline podepnij odpowiednie settingsy w PlayerSettings/Graphics (bądź wypnij w ogóle przy LRP)

Przy zmianie RenderPipeline'a pamiętaj o update'owaniu materiałów na scenie. Inaczej wszystko będzie w kolorze magenty



Q&A



Dziękuję za uwagę