

PackageExporter

DOC VERSION 1.0.7 - MAY 2021

Contents

Introduction.....	3
Overview.....	3
Installing.....	4
Usage.....	4
Upgrading.....	5
Changelog.....	5

Introduction

Thank you for trying out the **PackageExporter** extension! We hope you enjoy using the product and that it makes your game development project a success.

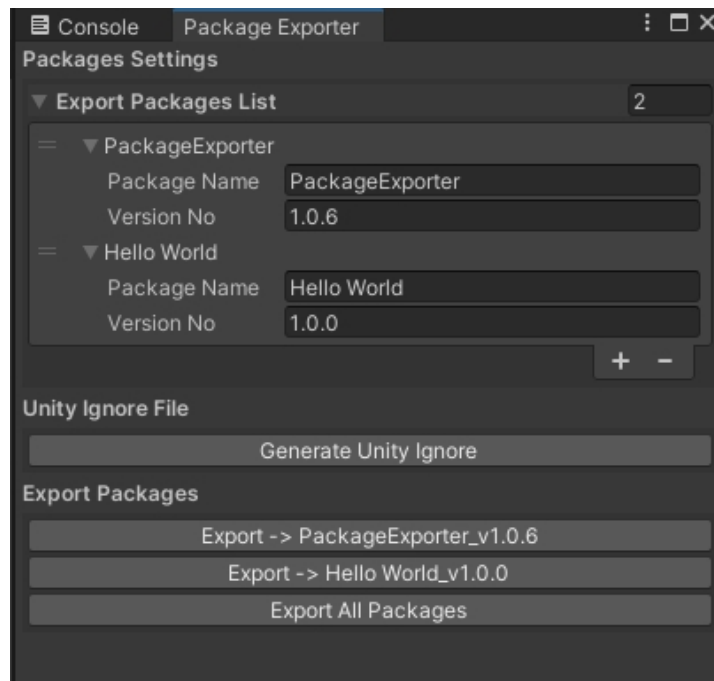
If you have any questions, please do not hesitate to contact us at jcs090218@gmail.com, we will be glad to help you out.

Also, when you have a spare moment, please leave us a review on the Asset Store.

Overview

Have you ever feel pain about exporting a package, and check/uncheck on assets repeatedly so you can export it correctly?

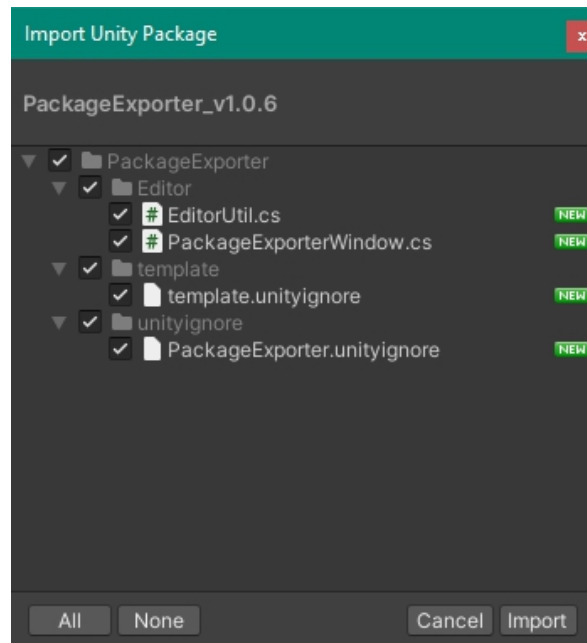
This extension simplifies the process of exporting packages by specifying rules inside a `PACKAGE_NAME.unityignore` file.



Installing

PackageExporter is a standard Unity extension and should be installed like any other Unity package. Please drag the `sHierarchy.unpackage` into your current project. Or go to your Editor, **Assets** → **Import Package** → **Custom Package** and the browse to the `PackageExporter.unpackage` file.

When downloading from the Asset Store, then the Download Manager will automate the process for you.

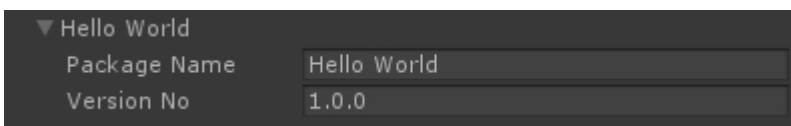


Once the Import dialog appears, click on the Import button.

Usage

Step 1. Let's create a `Hello World` package! Open the window from menu bar **PackageExporter** → **Window**.

Step 2. Then fill in the package information inside `Export Package List`!



Step 3. Then click on the button `Generate Unity Ignore` if you have not create one! The generated `.unityignore` file is under directory **Assets/PackageExporter/unityignore/**.

Step 4. Now open the generated `Hello World.unityignore` file in any text editor! Then you should be able to use that file as an ignore file like `.gitignore`.

Step 5. After editing the `.unityignore` file. Just click on button `Export -> Hello World_v1.0.0` to export the package. If you have multiple package setup, you can also click on button `Export All Packages` instead!

Upgrading

All version compatible code should be inside the package, so you would only need to overwrite the existing file for upgrading!

Changelog

The Changelog is in the main repository, you can see all changes [here](#).