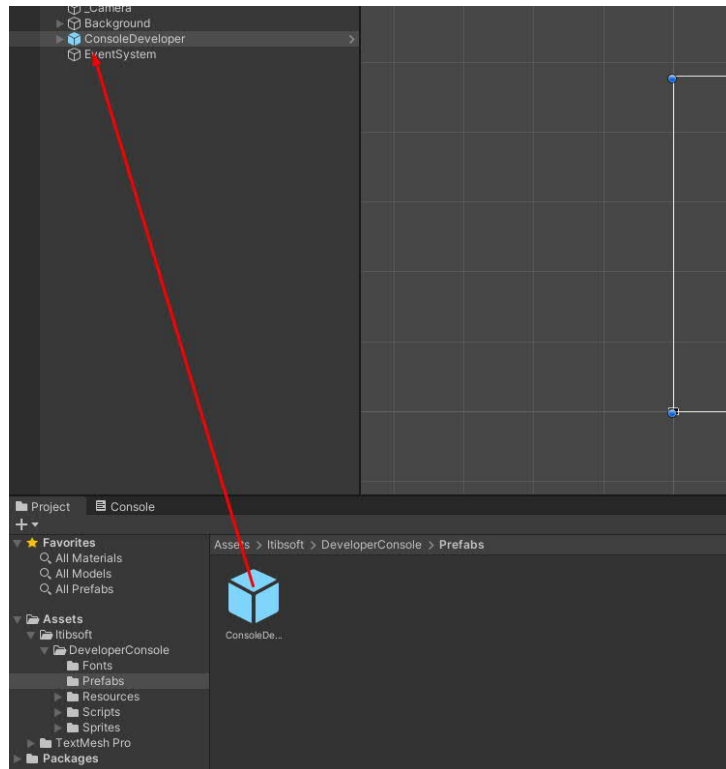


Документация: Console Developer v0.0.7 [Free]

- Урезанная версия. Скорее всего здесь не будет тех реализаций, что сейчас в разработке.

Как использовать:

1. Для начала нужно перетащить на главную сцену префаб самой консоли (ConsoleDeveloper.prefab). Он находится: [Assets\Itibsoft\DeveloperConsole\Prefabs](#)
Так же нужно проверить, чтобы на сцене у вас был "EventSystem". Без него не будет происходить нажатия на консоль. Он создается стандартно. ПКМ в пустом месте иерархии – UI – EventSystem



2. Включив режим Play Mode(Просто включить игру) нужно нажать кнопку "~" чтобы открылась сама консоль. После этого вы можете ей пользоваться.
3. Нажав на полосу ввода команды, мы можем вводить все любые имеющиеся команды в консоли. Для ознакомления с ними, вы можете прописать команду `/help`
4. Буфер истории команд работает так, что если вы уже вводили команду, то консоль добавляет ее в историю. И если в следующий вызов вам нужна та же самая команда или еще ранее введенная, то вы легко можете пройти по истории введенных команды стрелками на клавиатуре. Стрелка вверх выводит команды, которые были ранее введены. А стрелка назад выводит команды, которые более недавние. Работает по принципу LIFO (Последний зашел – первый вышел).

Принцип работы:

1. Как добавить новые команды:

а. Создание нового класса:

1. Ему нужно дать название: **ИмяCommand**. Это не принципиально, но просто это будет более понятно, для чего нужен данный класс.

2. После создания класса, его нужно поместить в namespace:

Itibsoft.ConsoleDeveloper.Commands

3. Классу нужно подписать атрибут **[Command]**. Это нужно для того, чтобы данная команда была добавлена в список активных команд консоли. Если этого атрибута не будет, консоль не будет видеть данную команду.

4. В этом классе нужно реализовать интерфейс **ICommand**. Данный интерфейс содержит в себе два свойства **Name**(Имя команды, которое нужно будет вводить в консоли) и **Description**(Описание данной команды. Оно появляется при вызове команды Help). Так же у интерфейса есть метод **void Execute()**; Он выполняет то, что должно происходить при вызове команды из консоли. Туда пишете свою реализацию, что должно произойти. Все что только вам в голову приходит, там можете реализовать.

```
5 namespace Itibsoft.ConsoleDeveloper.Commands Пространство имен, где находятся все команды
6 {
7     [Command] Атрибут для класса
8     Ссылка: 0
9     public class ClearConsoleCommand : ICommand Интерфейс
10    {
11        Ссылка: 11 Имя команды в консоле. То что нужно будет вводить
12        public string Name { get => "Clear"; set { } }
13
14        Ссылка: 7 Описание команды. Оно будет высвечиваться в буфере и при вызове команды /Help
15        public string Description { get => "Clear console"; set { } }
16
17        Ссылка: 5 Метод, которые вызывается при вызове данной команды
18        То, что должно произойти при вызове команды
19        public void Execute()
20        {
21            Logger.Instance.ClearLog();
22            HistoryCommands.Instance.ClearHistory();
23        }
24    }
25 }
```