Terp – Easy Interpolation

Made by Dor Ben Dor; agesonera@gmail.com;
www.dorbendor.com

What is this package?

Terp is a very small asset for easy interpolation between numbers. It can be used for literally anything (i.e. position, color, size etc.) and gives a more flexible design workflow for designers and animators.

It is much recommended for beginners, which frequently misuse lerping and damping functions.

What can you find here?

Curve – Scriptable Object (Scob)

This Scob is just an Animation Curve component encapsulated. It gives you the opportunity to create different types of curves to be used anywhere in your game, with quick iteration process for changing them. Designers and animators can easily create desired curves and drag and drop them into object components. Curves can be cast into Animation Curves and vice versa. It is recommended to cast an Animation Curve only once, and not iteratively, since it is not cost effective (creates a new scob instance each time).

When you create animation curves you can click on the cog icon to set the looping options. This is supported (The Linear PP Curve, for example, uses “Ping Pong” as the looping options) Refer to the manual for more info.

Please note that you can also use animation curves from the inspector instead of making generic scobs.

Animation Curve Script Reference

Animation Curve Manual

Terp – Static Script

This is a static representation of interpolation functions. It has two basic methods which represent the whole functionality of this asset. The first shorter interpolation method is used to get a specific interpolated value based on a curve and it is very similar to the Mathf.Lerp() function, structure wise. The second one is where the magic happens. Instead of inserting a value to get an effective evaluation and incrementing that value each frame, you insert the amount of time you wish the interpolation to take, and a reference to a counter variable, which will usually start at zero – and the function will increment itself!

Note that the functions do support unclamped values, meaning that you may get unwanted results if your cross normalized values (1,1). It can be used intentionally, of course. For reference, remember that changing the x axis will extend\shorten the time and changing the y axis will extend\shorten the max value (b).

Example Scene & Curves

Open the example screen and change the curves in the "Curves" folder to watch the dynamic results you can get with a few clicks.

Terms of Use

This asset is so basic you can do whatever you like with it. If you did find it effective, give me a shout!

Contact

This asset is supposed to be simple, and you can extend on it freely. If you have feedback please contact me, it'll be my pleasure to add new needed features or optimize the asset.

This is my e-mail: agesonera@gmail.com

References

- Unity Animation Curve Script Reference
- Unity Animation Curve Manual
- A GDC Talk given by Squirrel Eiserloh. Highly recommended. Was an inspiration for this asset.
- Check out Ryan Hipple's unity blog on scriptable objects.