2023-07-03_Interactive_04_Manim

July 3, 2023

```
[1]: from manim import *
```

Manim Community v0.17.3

```
[2]: %%manim -qm -v WARNING SimpleExample

class SimpleExample(Scene):
    def construct(self):
        dot = Dot(point=LEFT,radius=0.5,color="#ff0000")
        self.add(dot)
        self.wait(1)

        l1 = Line(LEFT,RIGHT)
        self.play(MoveAlongPath(dot, l1, run_time=1, rate_func=linear))

        l2 = Line(RIGHT,LEFT)
        self.play(MoveAlongPath(dot, l2, run_time=1, rate_func=linear))
```

<IPython.core.display.Video object>

```
[3]: %%manim -qm -v WARNING SimpleExample2

def getPtList(n):
    ptList=np.zeros(shape=(n,3))
    ptList[:,:2]=5*(np.random.random(size=(n,2))-0.5)
    return ptList

class SimpleExample2(Scene):
    def construct(self):

        self.camera.background_color = "#ece6e2"
        Text.set_default(color=BLACK,font_size=30)
        MathTex.set_default(color=BLACK,font_size=30)
        Tex.set_default(color=BLACK,font_size=30)
```

```
        nPts=10
        ptListA=getPtList(nPts)
        ptListB=getPtList(nPts)

        dotList=[Dot(point=x,radius=0.1,color="#ff0000") for x in ptListA]
        for dot in dotList:
            self.add(dot)
        self.wait(1)

        lineList=[Line(x,y) for x,y in zip(ptListA,ptListB)]
        self.play(*[MoveAlongPath(dot, l, run_time=1, rate_func=linear) for␣
  ↪dot,l in zip(dotList,lineList)])
        self.wait(1)

        lineList2=[Line(y,x) for x,y in zip(ptListA,ptListB)]
        self.play(*[MoveAlongPath(dot, l, run_time=1, rate_func=linear) for␣
  ↪dot,l in zip(dotList,lineList2)])
```

<IPython.core.display.Video object>

```
[4]: %%manim -qm -v WARNING SquareToCircle

class SquareToCircle(Scene):
    def construct(self):
        circle = Circle()  # create a circle
        circle.set_fill(PINK, opacity=0.5)  # set color and transparency

        square = Square()  # create a square
        square.rotate(PI / 4)  # rotate a certain amount

        self.play(Create(square))  # animate the creation of the square
        self.play(Transform(square, circle))  # interpolate the square into the␣
  ↪circle
        self.play(FadeOut(square))  # fade out animation
```

<IPython.core.display.Video object>

[ ]: