**Day 23:**

**Adding Text In manim**

**Basic Text**

You can add simple text using the Text class. This is useful for non-mathematical text.

**Python**

from manim import \*

class TextExample(Scene):

def construct(self):

text = Text("Hello, Manim!")

self.play(Write(text))

self.wait(2)

**MarkupText**

For more advanced text formatting, you can use MarkupText, which allows you to use PangoMarkup for styling.

**Python**

from manim import \*

class MarkupTextExample(Scene):

def construct(self):

text = MarkupText("This is <span fgcolor='yellow'>highlighted</span> text.", color=RED)

self.play(Write(text))

self.wait(2)

**Adding Mathematical Equations**

**Using LaTeX**

Manim supports LaTeX for rendering mathematical equations. Use the MathTex class for this purpose.

**Python**

from manim import \*

class MathExample(Scene):

def construct(self):

equation = MathTex(r"E = mc^2")

self.play(Write(equation))

self.wait(2)

**Combining Text and Equations**

You can combine text and equations in a single scene to create more complex animations.

**Python**

from manim import \*

class CombinedExample(Scene):

def construct(self):

text = Text("The famous equation:")

equation = MathTex(r"E = mc^2")

text.next\_to(equation, UP)

self.play(Write(text), Write(equation))

self.wait(2)

**Strategies for Transformations and Animations**

**Transforming Text and Equations**

You can transform one piece of text or an equation into another using the Transform method.

**Python**

from manim import \*

class TransformExample(Scene):

def construct(self):

text1 = Text("Hello, Manim!")

text2 = Text("Goodbye, Manim!")

self.play(Write(text1))

self.wait(1)

self.play(Transform(text1, text2))

self.wait(2)

**Animating Text and Equations**

You can animate text and equations using various animation methods like FadeIn, FadeOut, MoveToTarget, etc.

**Python**

from manim import \*

class AnimationExample(Scene):

def construct(self):

text = Text("Animating Text")

equation = MathTex(r"E = mc^2")

equation.next\_to(text, DOWN)

self.play(FadeIn(text))

self.wait(1)

self.play(FadeIn(equation))

self.wait(1)

self.play(FadeOut(text), FadeOut(equation))

self.wait(2)