**Day 22:**

**Creating Simple Shapes and Animations**

**Basic Shapes**

Manim provides a variety of basic shapes that you can easily create and manipulate. Here are some examples:

1. **Circle**:

**Python**

from manim import \*

class CircleExample(Scene):

def construct(self):

circle = Circle()

self.play(Create(circle))

self.wait(1)

AI-generated code. Review and use carefully.

1. **Square**:

**Python**

from manim import \*

class SquareExample(Scene):

def construct(self):

square = Square()

self.play(Create(square))

self.wait(1)

AI-generated code. Review and use carefully.

1. **Triangle**:

**Python**

from manim import \*

class TriangleExample(Scene):

def construct(self):

triangle = Triangle()

self.play(Create(triangle))

self.wait(1)

AI-generated code. Review and use carefully.

**Combining Shapes**

You can combine multiple shapes in a single scene:

**Python**

from manim import \*

class CombinedShapes(Scene):

def construct(self):

circle = Circle().shift(LEFT)

square = Square().shift(UP)

triangle = Triangle().shift(RIGHT)

self.play(Create(circle), Create(square), Create(triangle))

self.wait(1)

**Strategies for Transformations and Animations**

**Transformations**

Transformations allow you to morph one shape into another. Here are some common transformations:

1. **Transform**:

**Python**

from manim import \*

class TransformExample(Scene):

def construct(self):

circle = Circle()

square = Square()

self.play(Create(circle))

self.wait(1)

self.play(Transform(circle, square))

self.wait(1)

AI-generated code. Review and use carefully.

1. **ReplacementTransform**:

**Python**

from manim import \*

class ReplacementTransformExample(Scene):

def construct(self):

circle = Circle()

square = Square()

self.play(Create(circle))

self.wait(1)

self.play(ReplacementTransform(circle, square))

self.wait(1)

**Animations**

Animations bring your scenes to life. Here are some basic animation techniques:

1. **MoveToTarget**:

**Python**

from manim import \*

class MoveToTargetExample(Scene):

def construct(self):

circle = Circle()

circle.generate\_target()

circle.target.shift(RIGHT \* 2)

self.play(Create(circle))

self.wait(1)

self.play(MoveToTarget(circle))

self.wait(1)

AI-generated code. Review and use carefully.

1. **FadeIn and FadeOut**:

**Python**

from manim import \*

class FadeExample(Scene):

def construct(self):

circle = Circle()

self.play(FadeIn(circle))

self.wait(1)

self.play(FadeOut(circle))

self.wait(1)

AI-generated code. Review and use carefully.

**Rotate**:

**Python**

from manim import \*

class RotateExample(Scene):

def construct(self):

square = Square()

self.play(Create(square))

self.wait(1)

self.play(Rotate(square, angle=PI/4))

self.wait(1)

**Combining Animations**

You can combine multiple animations to create more complex scenes:

**Python**

from manim import \*

class CombinedAnimations(Scene):

def construct(self):

circle = Circle()

square = Square()

self.play(Create(circle))

self.wait(1)

self.play(Transform(circle, square))

self.wait(1)

self.play(Rotate(square, angle=PI/4))

self.wait(1)

self.play(FadeOut(square))

self.wait(1)