Video Explanation Script: “Python Time Class with AM/PM and Ratio”

“Hi everyone! Today we’ll learn how to create a Python class that represents time in hours and minutes, display it in AM/PM format, and calculate the ratio of minutes to hours while handling errors safely.”

“First, we define a class called Time with the \_init\_ method. This method will store two pieces of data for each instance: hours and minutes.”

class Time:

def \_init\_(self, hours, minutes):

self.hours = hours

self.minutes = minutes

“Here, self.hours and self.minutes are instance variables, meaning each Time object we create will have its own values.”

“Next, we define DisplayTime, which will print the time in a readable AM/PM format.”

Check if hours > 23 → print error.

Check if minutes > 59 → print error.

Convert 24-hour format to 12-hour format:

0 hours → 12 AM

1–11 → AM

12 → PM

13–23 → subtract 12, PM

if self.hours > 23:

print("The input hours should be less than 24")

return

if self.minutes > 59:

print("The input minutes should be less than 60")

return

period = "AM"

display\_hour = self.hours

if self.hours == 0:

display\_hour = 12

elif self.hours == 12:

period = "PM"

elif self.hours > 12:

display\_hour = self.hours - 12

period = "PM"

print(f"The time is {display\_hour}:{self.minutes:02d} {period}")

“Now we create a method to display the ratio of minutes to hours. Here, hours might be zero, so we use a try-except block to avoid a ZeroDivisionError.”

def DisplayRatio(self):

try:

ratio = self.minutes / self.hours

print(f"The ratio of minutes to hours is: {ratio}")

except ZeroDivisionError:

print("Cannot divide by zero hours!")

“We create a list of sample hours and minutes, and then create a Time object for each pair. We call both methods to see the output.”

hour\_min\_list = [(23,45), (34,50), (12,34), (14,67), (19,20), (2,15), (0, 10)]

for h, m in hour\_min\_list:

t = Time(h, m)

t.DisplayTime()

t.DisplayRatio()

print("-" \* 40)

Invalid hours or minutes → error message

Valid hours → printed in AM/PM format

Ratio → calculated safely; if hours = 0 → “Cannot divide by zero hours!”

“And that’s it! We now have a fully functional Python class for handling time with proper error handling, formatted output, and calculations. You can extend this further by adding seconds, or formatting the ratio to always show an integer if you want.”