

## Study Manager: Line-by-Line Explanation

Python Code	Explanation (What is happening?)
<code>from datetime import datetime</code>	We are grabbing a special clock tool from the big Python toolbox, even though we aren't using it yet!
<code>study = {}</code>	This is our empty backpack. We will put subject names and times inside it later.
<code>day_count = 0</code>	This is a scoreboard that starts at zero. It counts how many times we check our progress.
<code>def add_Data(name, time):</code>	This is a recipe (function) called 'add_Data'. It needs a 'name' and 'time' to work.
<code>if name in study:</code>	We look inside the backpack. Is this subject name already in there?
<code># If subject exists...</code>	Note: We are handling the case where we already have the subject.
<code>study[name] += time</code>	If it is there, we take the old time and add the new time to it.
<code>print(f"Subject '{name}' already exists!")</code>	We tell the user: "Hey, we found this subject!"
<code>print(f"Added {time} minutes...")</code>	We tell them we added more minutes to the pile.
<code>print(f"New total time for '{name}'...")</code>	We show them the big new total number.
<code>else:</code>	But, if the subject was NOT in the backpack...
<code># If subject doesn't exist...</code>	Note: This is a brand new subject.
<code>study[name] = time</code>	We put the new subject and its time into the backpack for the first time.
<code>print(f"Subject name '{name}'...")</code>	We cheer because we added a new thing!

```
def update_Data(name, time):
```

Here is a new recipe called 'update\_Data'. It changes things.

```
if name in study:
```

First, we check if the subject is actually in our backpack.

```
study[name] = time
```

We erase the old time and write down the brand new time.

```
print(f"Updated! Subject  
'{name}'...")
```

We tell the user the change is done.

```
else:
```

If we couldn't find that subject in the backpack...

```
print(f"This Subject Name  
'{name}' is Not Found!")
```

We say: "Sorry, I don't know that subject!"

```
def delete_Data(name):
```

This recipe is for throwing things away.

```
if name in study:
```

We check if the subject is in the backpack first.

```
del study[name]
```

We take the subject out of the backpack and toss it in the trash!

```
print(f"Subject name '{name}' is  
Deleted")
```

We confirm it is gone forever.

```
else:
```

If it wasn't there to begin with...

```
print(f"This Subject Name  
'{name}' is Not Found!")
```

We say: "I can't delete what isn't there!"

```
def view_data():
```

This recipe lets us look at everything we have.

```
if study:
```

We check if the backpack has anything inside it.

```
print("\n--- Current Study List -  
---")
```

We print a fancy title for our list.

```
for name, time in study.items():
```

We pull out every single item (name and time) from the backpack, one by one.

```
print(f"{name}: {time} minutes")
```

We print the name and the time on the screen.

<code>else:</code>	If the backpack is completely empty...
<code>print("No Study Plan Found")</code>	We tell the user there is nothing to see here.
<code>def show_statistics():</code>	This is a big recipe to do some math for us.
<code>global day_count</code>	We are shouting to get the 'day_count' scoreboard from outside this room so we can change it.
<code>if study:</code>	We only do math if there is data in the backpack.
<code>day_count += 1</code>	We add 1 to our scoreboard because we are checking stats again.
<code>total = 0</code>	We start a counter at zero to add up all the minutes.
<code>for time in study.values():</code>	We look at only the 'time' numbers in our backpack.
<code>total += time</code>	We add each time to our total pile.
<code>sub_count = len(study)</code>	We count how many different subjects are in the backpack.
<code>average = total // sub_count...</code>	We share the total time equally among subjects. The '//' means no decimals allowed!
<code>print(f"\n📊 Study Day Count : {day_count}")</code>	We show the scoreboard number.
<code>print(f"Total Subjects: {sub_count}")</code>	We show how many subjects we have.
<code>print(f"Total Study Time: {total} minutes")</code>	We show the total time for everything.
<code>print(f"Average Study Time per Subject...")</code>	We show the average time.
<code>if average &lt;= 80:</code>	We check: Is the average time small (less than or equal to 80)?
<code>print("\n You Need To Study More")</code>	If it is small, we give a gentle nudge to work harder.

<code>else:</code>	If the number is big...
<code>print("Keep Going")</code>	We give a high-five!
<code>else:</code>	If the backpack was empty...
<code>print("No Data")</code>	We say there is nothing to calculate.
<code>def main():</code>	This is the Main Control Room where the program starts.
<code>while True:</code>	This is a 'forever loop'. It keeps spinning around and around until we tell it to stop.
<code>print("\n=== Study Management System ===")</code>	We print the big welcome sign.
<code>print("1. ADD")</code>	Option 1 is to add.
<code>print("2. Update")</code>	Option 2 is to change.
<code>print("3. Delete")</code>	Option 3 is to remove.
<code>print("4. View")</code>	Option 4 is to look.
<code>print("5. Show Statistics")</code>	Option 5 is for math.
<code>print("6. Exit")</code>	Option 6 is to leave.
<code>choice = int(input('Enter Your choice...'))</code>	We ask the human to type a number, and we turn it into a real number (integer).
<code>if choice == 1:</code>	If the human picked door number 1...
<code>name = input("Enter Subject Name:")</code>	We ask: "What is the subject name?"
<code>time = int(input("Enter Study Time..."))</code>	We ask: "How many minutes?" and make sure it's a number.
<code>add_Data(name, time)</code>	We run the 'add_Data' recipe we wrote earlier.
<code>elif choice == 2:</code>	If the human picked door number 2...

```
name = input("Enter Subject  
Name...")
```

We ask which subject to fix.

```
time = int(input("Enter New Study  
Time..."))
```

We ask for the new number.

```
update_Data(name, time)
```

We run the 'update\_Data' recipe.

```
elif choice == 3:
```

If the human picked door number 3...

```
name = input("Enter Subject  
Name...")
```

We ask which subject to throw away.

```
delete_Data(name)
```

We run the 'delete\_Data' recipe.

```
elif choice == 4:
```

If the human picked door number 4...

```
view_data()
```

We run the 'view\_data' recipe to show the list.

```
elif choice == 5:
```

If the human picked door number 5...

```
show_statistics()
```

We run the 'show\_statistics' recipe to do the math.

```
elif choice == 6:
```

If the human picked door number 6...

```
print('Closing the Program\n')
```

We say goodbye!

```
break
```

This is the emergency brake! It stops the 'forever loop' and ends the program.

```
else:
```

If the human typed a weird number like 99...

```
print("Invalid Choice")
```

We say: "That wasn't one of the options!"