

AI Assisted Coding

Name: K.Sahithya

Batch No:02

Branch : CSE AIML

Enroll no:2403A52063

DATE:16.09.2025

TASK 1

a=10

if a=10;

print("Equal")

In the given code identify the Syntax error in comditionals and give the correct code

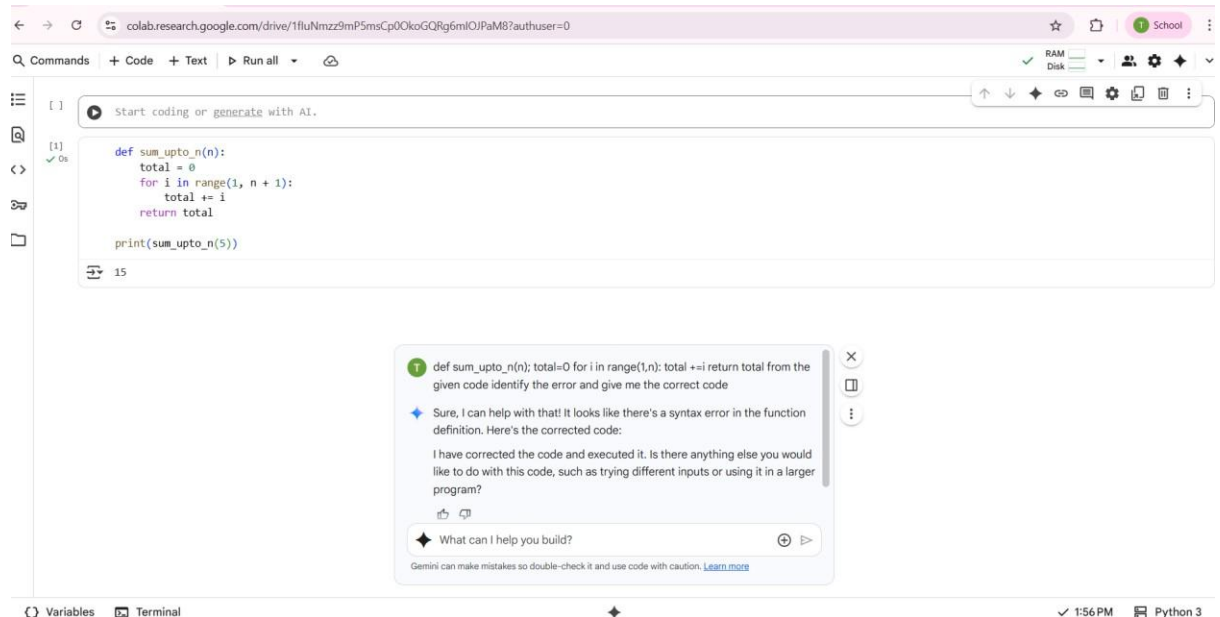
The screenshot shows a code editor interface. On the left, a code snippet is displayed:

```
[1] a = 10
    if a = 10:
        print("Equal")
```

Below the code, the output "Equal" is shown. On the right, a Gemini chat window is open. The chat history shows a user query: "a=10 if a=10; print('Equal') in the given code identify the Syntax error in comditionals and give the correct code". The Gemini response explains the syntax error: "You're right, there's a syntax error in the conditional statement. In Python, you use == for comparison, not =. Also, you need a colon after the condition and indentation for the code inside the if block. Here's the corrected code:". The corrected code is provided: "Great! The syntax error is fixed. Since you previously uploaded a PNG file, would you like to work with that image? For example, we could explore ways to display it or perform some image processing if that's what you have in mind." At the bottom of the chat window, there is a prompt: "What can I help you build?".

TASK 2

def sum_upto_n(n); total=0 for i in range(1,n): total +=i return total from the given code identify the error and give me the correct code



Start coding or generate with AI.

```
[1] ✓ 0s
def sum_upto_n(n):
    total = 0
    for i in range(1, n + 1):
        total += i
    return total

print(sum_upto_n(5))
```

15

T def sum_upto_n(n); total=0 for i in range(1,n): total +=i return total from the given code identify the error and give me the correct code

◆ Sure, I can help with that! It looks like there's a syntax error in the function definition. Here's the corrected code:

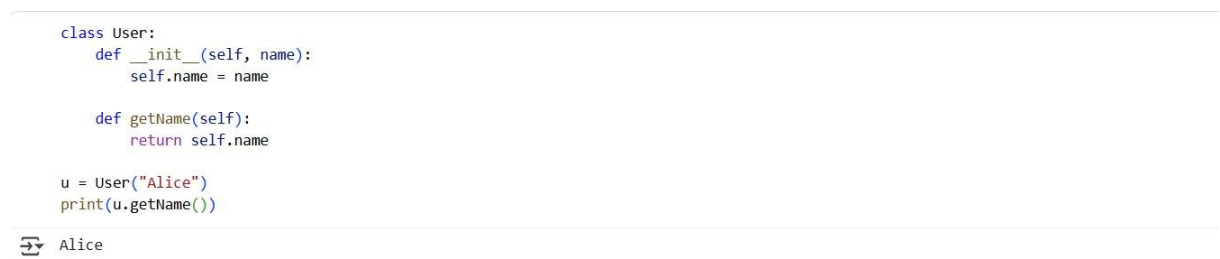
I have corrected the code and executed it. Is there anything else you would like to do with this code, such as trying different inputs or using it in a larger program?

What can I help you build?

Gemini can make mistakes so double-check it and use code with caution. [Learn more](#)

TASK 3

class User: def __init__(self,name): self.name=name u=User("Alice")
print(u.getName()) From the give code identify the error and give me the correct code



```
class User:
    def __init__(self, name):
        self.name = name

    def getName(self):
        return self.name

u = User("Alice")
print(u.getName())
```

Alice

T class User: def __init__(self,name): self.name=name u=User("Alice")
print(u.getName()) From the give code identify the error and give me the correct code

◆ It seems like you are trying to call a method `getName()` that is not defined in your `User` class. I've added the `getName()` method to the class definition. Here is the corrected code:

Great! Is there anything else you would like to do with this code, like adding more methods to the `User` class or creating more user objects?

What can I help you build?

Gemini can make mistakes so double-check it and use code with caution. [Learn more](#)

TASK 4

class Car: def start(): print("Car started") mycar=Car() mycar.start() from the given code identify the error and give me the correct code

```
class Car:
    def start(self):
        print("Car started")

mycar = Car()
mycar.start()
```

T class Car: def start(): print("Car started") mycar=Car() mycar.start() from the given code identify the error and give me the correct code

◆ I can help with that! The error is that the `start` method in the `Car` class is missing the `self` parameter. Instance methods in Python require `self` as the first argument to refer to the instance of the class.

Here is the corrected code:

The code has been corrected and executed successfully.

Is there anything else you would like to do with this `Car` class, like adding

◆ What can I help you build?

Gemini can make mistakes so double-check it and use code with caution. [Learn more](#)

TASK 5

def grade_student(score): if score<40: return "A" elif score <70: return "B" else: return"C" from the given code identify the error and give me the correct code

▶ Start coding or [generate](#) with AI.

```
def grade_student(score):
    if score < 40:
        return "C" # Corrected: Scores less than 40 should be "C"
    elif score < 70:
        return "B"
    else:
        return "A" # Corrected: Scores 70 or greater should be "A"

# Example usage:
print(grade_student(30))
print(grade_student(55))
print(grade_student(80))
```

◆ I can help with that! The error in the code is in the logic of the conditional statements. Currently, a score of 30 would result in "B" because it's less than 70. I'll generate a new cell with the corrected code that will assign grades based on the correct score ranges.

I've corrected the code and it's running as expected. Would you like to try another example with different scores, or perhaps explore how to handle invalid input (like negative scores or non-numeric input)?

👍 🗨

◆ What can I help you build?

Gemini can make mistakes so double-check it and use code with caution. [Learn more](#)

