

Currently I'm working on a Python programming problem. The current program below contains some bugs. Can you help in fixing these bugs with as few changes as possible? Below I first provide the problem description and then the buggy program.

Given an integer N, find the number of divisors of N that are divisible by 3.

Buggy Program:

```
class P3_Python_buggy:

    def find_divisors_divisible_by_3(N):
        count = 0

        for i in range(1, N + 1):
            if N % i != 0:
                if i % 3 == 0:
                    count += 1

        return count

    def main():
        n = 36 # Beispielwert
        count = P3_Python_buggy.find_divisors_divisible_by_3(n)
        print(f"Number of divisors of {n} that are divisible by 3: {count}")

if __name__ == "__main__":
    P3_Python_buggy.main()
```

Can you fix the above buggy program? Modify only the code within the function; make sure that you make minimal possible changes needed to fix the program.

To fix the program, we need to correct the logic in the `find_divisors_divisible_by_3` function. The current program incorrectly increments the count when a number is not a divisor of `N` but is divisible by 3. Instead, we need to check that `i` is both a divisor of `N` and divisible by 3.

Here's the corrected function with minimal changes:

```
python Code kopieren

class P3_Python_buggy:

    def find_divisors_divisible_by_3(N):
        count = 0

        for i in range(1, N + 1):
            if N % i == 0: # Check if i is a divisor of N
                if i % 3 == 0: # Check if the divisor is divisible by 3
                    count += 1

        return count

    def main():
        n = 36 # Example value
        count = P3_Python_buggy.find_divisors_divisible_by_3(n)
        print(f"Number of divisors of {n} that are divisible by 3: {count}")

if __name__ == "__main__":
    P3_Python_buggy.main()
```

- Explanation of Changes:
- The condition `if N % i != 0` was changed to `if N % i == 0` so that it correctly checks whether `i` is a divisor of `N`.
 - The rest of the code remains the same. We first ensure `i` is a divisor and then check if `i` is divisible by 3.

Now, the function correctly counts the number of divisors of `N` that are divisible by 3.

⏮ ⏪ ⏩ ⏭ 🔍 ↻