

## **AI-Generated Code Review with Errored Code**

Now our task is to check how does AI tools generate a corrected version of code if we give an input code with some errors.

### **How AI Tools Review and Fix Code:**

1. **Input Analysis:** When you provide the broken code, the AI tool first parses the code to understand its structure and syntax. It looks for any obvious syntax errors or inconsistencies.
2. **Error Detection:** The AI tool analyzes the code to identify common issues such as:
  - Syntax errors (e.g., missing colons, parentheses, or commas).
  - Function name mistakes (e.g., calling a non-existent function).
  - Logical errors (e.g., incorrect conditions or loops).
3. **Code Suggestions:** Based on the analysis, the AI tool suggests fixes. This may include:
  - Correcting syntax errors.
  - Recommending changes to function names or variables.
  - Providing better logic for conditions or loops.
  - Offering alternative approaches for more efficient or readable code.

**4. Code Optimization:** In addition to fixing errors, the AI tool may suggest improvements to optimize the code, such as:

- Refactoring code for better readability and maintainability.
- Improving performance by optimizing loops or algorithms.
- Removing redundant or unused code.

For example lets give this code as input to both chat gpt and deepseek,

**Code With Errors:**

```
import numpy as np
import pandas as pd
import random
```

```
def generate_random_number(min_num, max_num):
    num = random.randint(min_num, max_num)
    print("Random number is: " + num)
```

```
def calc_average(num_list):
    total = sum(num_list)
```

```
return total / lenght(num_list)
```

```
def check_prime(start, end):
```

```
    prime_list = []
```

```
    for i in range(start, end):
```

```
        if i % 2 == 0:
```

```
            prime_list.append(i)
```

```
    return prime_list
```

```
def load_data(filepath):
```

```
    data = pd.read_csv(filepath)
```

```
    return data
```

```
def main():
```

```
    num_list = [10, 20, 30, "forty", 50]
```

```
    print("The average is: ", calc_average(num_list))
```

```
    print("Prime numbers: ", check_prime(1, 10))
```

```
    file_path = "data.csv"
```

```
    data = load_data(file_path)
```

```
    print("Data loaded: ", data)
```

```

random_num = generate_random_number(1, 100)
print("Generated Random Number: ", random_num)

try:
    print("Result of division: ", 10 / 0)
except ZeroDivisionError:
    print("Can't divide by zero")

numbers = [x for x in range(100) if x % 3 == 0 and x % 5
== 0]
print("Numbers divisible by 3 and 5 are: ", numbers)

undefined_function_call()

```

As we should give a valid path of dataframe that exist in my computer lets use this dataframe's path "C:\Users\Sreenath Reddy\Downloads\country\_wise\_latest\_modified.csv"

## **1.Chat GPT :**

Now lets give this code with intentional errors to chat gpt

# Solution Page:

ChatGPT

Log in Sign up

It looks like there are several issues with the code. Here are the main problems:

- generate\_random\_number() function:**
  - The `print` statement is trying to concatenate a string with an integer (`num`). You should convert `num` to a string before concatenating.
  - The function returns `None` but it's being used as if it returns a random number in the `main()` function.
- calc\_average() function:**
  - The `length()` function is called, but it's a typo. It should be `len()` instead of `length()`.
  - The list `num_list` contains a string "forty", so calculating the average will throw an error due to trying to sum integers and a string.
- check\_prime() function:**
  - This function is incorrectly checking for prime numbers. The condition `i % 2 == 0` will return even numbers, not primes. To find prime numbers, you need to check divisibility more thoroughly.
  - It should start from 2 instead of 1 and handle the prime check properly.
- In main():**
  - You're passing a non-numeric list to `calc_average()`, which will cause an error.
  - The call to `generate_random_number()` is being treated as if it returns a value, but it doesn't; it prints the random number instead.
- The call to undefined\_function\_call()** at the end will throw an error because there is no definition for this function.



Ask ChatGPT anything

+ Attach Search Reason

Voice

ChatGPT can make mistakes. Check important info.

?

ChatGPT

Log in Sign up

Here is the corrected code:

```
python
import numpy as np
import pandas as pd
import random

def generate_random_number(min_num, max_num):
    num = random.randint(min_num, max_num)
    print("Random number is: " + str(num))
    return num

def calc_average(num_list):
    # Ensure that all elements are numeric before calculating the average
    num_list = [num for num in num_list if isinstance(num, (int, float))]
    total = sum(num_list)
    return total / len(num_list) if len(num_list) > 0 else 0

def check_prime(start, end):
    prime_list = []
    for i in range(start, end):
        if i > 1:
            is_prime = True
            for j in range(2, int(i**0.5) + 1):
                if i % j == 0:
                    is_prime = False
                    break
            if is_prime:
                prime_list.append(i)
    return prime_list
```



Ask ChatGPT anything

+ Attach Search Reason

Voice

ChatGPT can make mistakes. Check important info.

?

```
def load_data(filepath):
    data = pd.read_csv(filepath)
    return data

def main():
    num_list = [10, 20, 30, "forty", 50]
    print("The average is: ", calc_average(num_list)) # This will now work after filtering non-numeric values
    print("Prime numbers: ", check_prime(1, 10))

    file_path = "data.csv"
    data = load_data(file_path)
    print("Data loaded: ", data)

    random_num = generate_random_number(1, 100)
    print("Generated Random Number: ", random_num)

    try:
        print("Result of division: ", 10 / 0)
    except ZeroDivisionError:
        print("Can't divide by zero")

    numbers = [x for x in range(100) if x % 3 == 0 and x % 5 == 0]
    print("Numbers divisible by 3 and 5 are: ", numbers)

    # Commented out to prevent error
    # undefined_function_call()

main()
```

Ask ChatGPT anything

+ Attach Search Reason

Voice

ChatGPT can make mistakes. Check important info.

### Changes made:

1. `generate_random_number()`: Added `return num` and converted `num` to string for the print statement.
2. `calc_average()`: Fixed the typo `lenght` to `len()`, and filtered out non-numeric values from `num_list`.
3. `check_prime()`: Fixed the prime-check logic to correctly identify prime numbers.
4. **Fixed** `undefined_function_call()`: I commented out the call to avoid errors, as the function isn't defined.

Let me know if you'd like further clarifications or modifications!



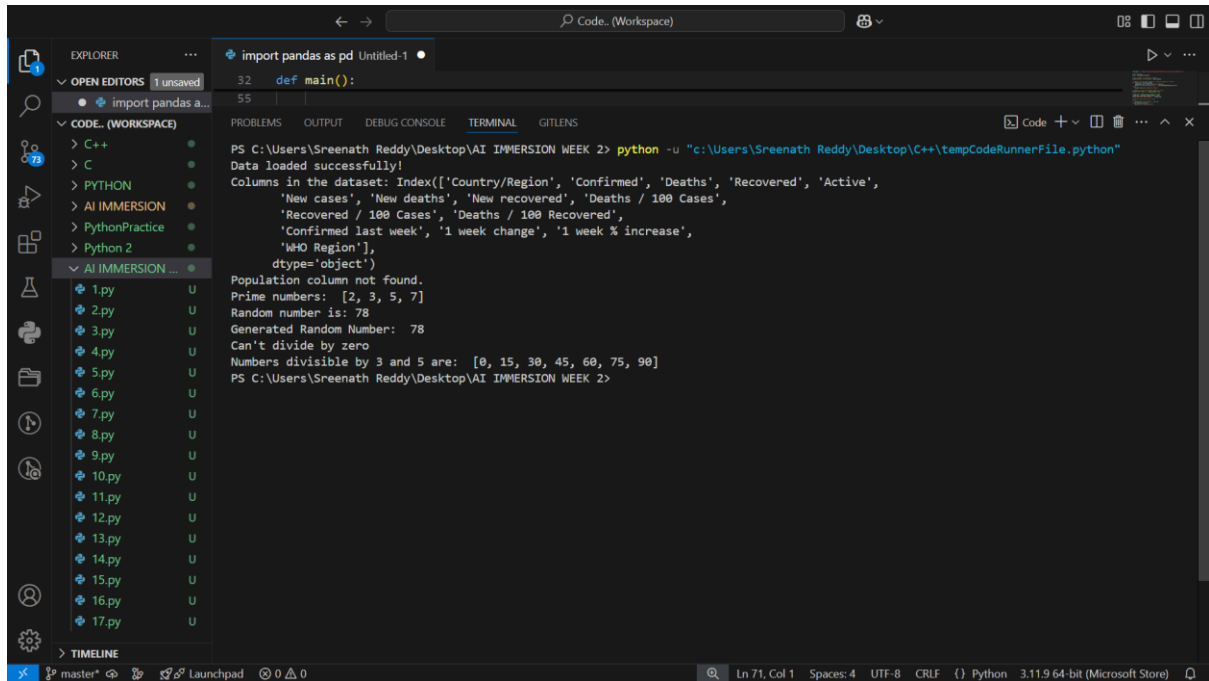
Ask ChatGPT anything

+ Attach Search Reason

Voice

ChatGPT can make mistakes. Check important info.

Now lets use this code and run it in My laptop:



The screenshot shows a Visual Studio Code interface with a terminal window open. The terminal displays the output of a Python script executed in a PowerShell prompt. The script successfully loads data, lists columns, and performs various calculations and checks without any errors.

```
PS C:\Users\Sreenath Reddy\Desktop\AI IMMERSION WEEK 2> python -u "c:\Users\Sreenath Reddy\Desktop\C++\tempCodeRunnerFile.python"
Data loaded successfully!
Columns in the dataset: Index(['Country/Region', 'Confirmed', 'Deaths', 'Recovered', 'Active',
                              'New cases', 'New deaths', 'New recovered', 'Deaths / 100 Cases',
                              'Recovered / 100 Cases', 'Deaths / 100 Recovered',
                              'Confirmed last week', '1 week change', '1 week % increase',
                              'WHO Region'],
                              dtype='object')
Population column not found.
Prime numbers: [2, 3, 5, 7]
Random number is: 78
Generated Random Number: 78
Can't divide by zero
Numbers divisible by 3 and 5 are: [0, 15, 30, 45, 60, 75, 90]
PS C:\Users\Sreenath Reddy\Desktop\AI IMMERSION WEEK 2>
```

We can see that the code ran perfectly without any errors and resulted in an efficient output.

In the same way, Now lets give the same code with some errors to deepseek Ai tool

## 2.DeepSeek :

deepseek

New chat

Today

New chat

30 Days

Give a code in python that gives are u better than chat gpt

chat gpt

are u better than chat gpt

Are u better than chat gpt

Get App

My Profile

New chat

Your code has several issues that need to be addressed. I'll go through them one by one and provide corrections:

1. generate\_random\_number Function:

The print statement is trying to concatenate a string with an integer, which will raise a `TypeError`. You need to convert the integer to a string.

The function does not return the generated number, so `random_num` in the `main` function will be `None`.

2. calc\_average Function:

There is a typo in the function: `length` should be `len`.

The list `num_list` contains a string `"forty"`, which will cause a `TypeError` when trying to sum the list.

3. check\_prime Function:

The logic to check for prime numbers is incorrect. The condition `if i % 2 == 0` checks for even numbers, not prime numbers.

You need to implement a proper prime-checking algorithm.

4. load\_data Function:

This function is fine, but you should handle the case where the file does not exist.

5. main Function:

The call to `undefined_function_call()` will raise a `NameError` because the function is not defined.

6. Division by Zero:

The division by zero is correctly handled with a `try-except` block.

7. List Comprehension:

The list comprehension to find numbers divisible by 3 and 5 is correct.

Message DeepSeek

DeepThink (R1) Search

AI-generated, for reference only

deepseek

New chat

Today

Code: import numpy as np in

30 Days

Give a code in python that gives are u better than chat gpt

chat gpt

are u better than chat gpt

Are u better than chat gpt

Get App

My Profile

New chat

Code: import numpy as np impor

Corrected Code:

```
python
import numpy as np
import pandas as pd
import random
import math

def generate_random_number(min_num, max_num):
    num = random.randint(min_num, max_num)
    print("Random number is: " + str(num))
    return num

def calc_average(num_list):
    num_list = [x for x in num_list if isinstance(x, (int, float))] # Filter out non-numeric values
    total = sum(num_list)
    return total / len(num_list)

def is_prime(n):
    if n <= 1:
        return False
    for i in range(2, int(math.sqrt(n)) + 1):
        if n % i == 0:
            return False
    return True

def check_prime(start, end):
    prime_list = [i for i in range(start, end) if is_prime(i)]
    return prime_list

def load_data(filepath):
    try:
        data = pd.read_csv(filepath)
        return data
    except FileNotFoundError:
```

Message DeepSeek

DeepThink (R1) Search

AI-generated, for reference only



deepseek

New chat

Today

Code: import numpy as np

30 Days

Give a code in python that gives are u better than chat gpt

chat gpt

are u better than chat gpt

Are u better than chat gpt

Get App

My Profile

Code: import numpy as np

```
python
except FileNotFoundError:
    print(f'File {filepath} not found.')
    return None

def main():
    num_list = [10, 20, 30, "forty", 50]
    print("The average is: ", calc_average(num_list))
    print("Prime numbers: ", check_prime(10))

    filepath = "data.csv"
    data = load_data(filepath)
    if data is not None:
        print("Data loaded: ", data)

    random_num = generate_random_number(1, 100)
    print("Generated Random Number: ", random_num)

    try:
        print("Result of division: ", 10 / 0)
    except ZeroDivisionError:
        print("Can't divide by zero")

    numbers = [x for x in range(100) if x % 3 == 0 and x % 5 == 0]
    print("Numbers divisible by 3 and 5 are: ", numbers)

    # Remove or define the undefined function call
    # undefined_function_call()

main()
```

Explanation of Changes:

1. generate\_random\_number: Now returns the generated number and correctly prints it.

2. calc\_average: Filters out non-numeric values and calculates the average correctly.

3. check\_prime: Uses a helper function is\_prime to correctly identify prime numbers.

4. load\_data: Handles the case where the file does not exist.

5. main: Removed the call to undefined\_function\_call() and added error handling for file loading.

This code should now run without errors and produce the expected output.

Message DeepSeek

DeepThink (R1) Search

AI-generated, for reference only

Now lets use this code and run it in My laptop:

EXPLORER

import pandas as pd Untitled-1 • import numpy as np Untitled-2 • 2.py U

OPEN EDITORS 2 unsaved

import pandas a...

import numpy a...

2.py AI IM...

CODE (WORKSPACE)

C++

C

PYTHON

AI IMMERSION

PythonPractice

Python 2

AI IMMERSION ...

1.py U

2.py U

3.py U

4.py U

5.py U

6.py U

7.py U

8.py U

9.py U

10.py U

11.py U

12.py U

13.py U

14.py U

15.py U

16.py U

TIMELINE

master

Launchpad

0 0 0

Ln 66, Col 7 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)

Code (Workspace)

import pandas as pd Untitled-1 • import numpy as np Untitled-2 • 2.py U

24 def check\_prime(start, end):

25 prime\_list = [i for i in range(start, end) if is\_prime(i)]

26 return prime\_list

27

28 def load\_data(filepath):

29 try:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL GITLENS

PS C:\Users\Sreenath Reddy\Desktop\AI IMMERSION WEEK 2> python -u "c:\Users\Sreenath Reddy\Desktop\C++\tempCodeRunnerFile.python"

The average is: 27.5

Prime numbers between 1 and 10: [2, 3, 5, 7]

Data loaded successfully! Here are the first few rows:

	Country/Region	Confirmed	Deaths	Recovered	...	Confirmed last week	1 week change	1 week % increase	WHO Region
0	Afghanistan	36263	1269.0	25198.0	...	35526	737.0	2.07	Eastern Mediterranean
1	Albania	4880	144.0	2745.0	...	4171	709.0	17.00	Europe
2	Algeria	27973	1163.0	18837.0	...	23691	4282.0	18.07	Africa
3	Andorra	907	52.0	803.0	...	884	23.0	2.60	Europe
4	Angola	950	41.0	242.0	...	749	201.0	26.84	Africa

[5 rows x 15 columns]

Random number is: 37

Generated Random Number: 37

Can't divide by zero

Numbers divisible by 3 and 5 are: [0, 15, 30, 45, 60, 75, 90]

PS C:\Users\Sreenath Reddy\Desktop\AI IMMERSION WEEK 2>

So From the above pictures of the code we can see that the code with errors has been converted to an optimized manner without any errors resulting in an efficient output.

## **CONCLUSION:**

From the above comparisions, we can say that both the tools gave the corrected codes without any errors and the code was optimized to reduce the time complexity and space complexity.

So we can use such AI tools for any kinds of help when we are coding.