Исследователь нашел подсказки для Chatgpt , которые помогут вам получить идеальный ответ, даже на самый каверзный запрос.

1. Универсальная промпт, который заставит Chatgptразбить любой ответ на пошаговую инструкцию для более точного поиска ошибок:

«Let's work this out in a step by step way to be sure we have the right answer.» ;

2. Включаем корректоры. ИИ можно заставить проверить точность ответа в Google, при этом ChatGPT самостоятельно перепроверит его предыдущий ответ на ошибки и слабые аргументы:

«You are a researcher tasked with investigating the [предыдущий ответ] response options provided. List the flaws and faulty logic of each answer option. Let's work this out in a step by step way to be sure we have all the errors:»;

3. Следующим промпт поможет ИИ скомпоновать все действия в один ответ:

«You are a resolver tasked with 1) finding which of the [предыдущий ответ] answer options the researcher thought was best 2) improving that answer, and 3) Printing the improved answer in full. Let's work this out in a step by step way to be sure we have the right answer:»;

# Интересные и простые промты для ChatGPT

▪️Почему нужно \*что делать\*? Аргументируй свой ответ и приведи комментарий, ссылаясь на исследования по теме.

▪️Предложи \*количество\* тем для личного блога на тему \*тематика\*.

▪️Расскажи о мифах в работе \*название профессии\*.

▪️Объясни, почему возникает страх перед \*ситуация\*,

и какие инструменты самопомощи существуют.

▪️Создай интересную историю о \*описание человека\*, который добился, чего хотел. Подробно опиши его путь к \*название цели\* и пользу для него после работы над \*название\*.

▪️Перепиши прикрепленный в файле материал таким образом, чтобы его было легче читать, он привлекал новых подписчиков. Заданный материал должен иметь тон — \*описание тона\*.

▪️Напиши полезный отзыв для сайта \*название сайта\*, отметив особенности \*название\*.

Предоставь шаблон для email-рассылки по тематике \*название темы\*, который будет включать в себя \*особенности\*.

▪️Создай бизнес план для сферы \*название\* с бюджетом \*сумма\* и возможностью дополнительного финансирования.

▪️Составь план для PowerPoint презентации на тему \*название\* с количеством слайдов \*количество\*.

Цель презентации следующая — реклама продукта.

▪️Попробуй решить задачу с помощью инструментов \*уточнение\*, условия которой \*описание условий\*.

▪️Расскажи, сколько произведений мировой

литературы существует на нашей планете?

▪️Предложи \*количество\* идей для подарка маме на 8 марта.

▪️В моей жизни возникла трудная ситуация — \*описание\*. Составь нумерованный список, в котором будут советы о том, как справиться с этим состоянием.

▪️Напиши полезную инструкцию по использованию приложения \*название\*.

▪️Создай список вопросов для социального опроса на тему \*название\*.

▪️Предложи список тем для курсовой работы по предмету \*название\*.

▪️Напиши список вопросов для интервью в виде комментария по теме \*название\* с известной личностью \*уточнение\*.

▪️Создай инструкцию по обучению бота, который будет оповещать людей о том, что их заказ пришёл в пункт выдачи.

▪️Посчитай, какое количество упоминаний моего профиля \*ссылка\* есть в социальных сетях на текущий момент, сколько человек нажало кнопку «поделиться».

▪️Изучи профиль \*ссылка\* и предложи \*количество\* вариантов для его продвижения.

Определи главную идею и цель текста.

▪️Изучи предложенный фрагмент из художественного произведения \*название\*, в \*количество\* предложениях опиши его суть.

▪️Придумай форму для внесения данных клиентов из чата, в которой должна содержаться основная информация о \*тема\* с указанием дополнительных инструментов.

▪️Создай план мероприятия \*тематика\* с учетом количества гостей — \*количество\* и среднего возраста участников \*возраст\*.

▪️Напиши план продвижения маркетингового продукта с целевой аудиторией \*уточнение\*.

▪️Найди \*количество\* источников информации, которые помогут аргументировать точку зрения на тему \*название\*.

▪️Создай код по заданным параметрам функции.

▪️Проанализируй новостные порталы и тематические сайты на тему \*название\*. Составь отчёт в виде таблицы о главных изменениях и новинках в отрасли.

▪️Изучи резюме \*количество\* кандидатов, выбери наиболее подходящее под условия вакансии — \*описание\*.

▪️Напиши ответы с дополнительными комментариями на часто задаваемые вопросы от пользователей продукта \*название\* — список вопросов.

▪️Придумай \*количество\* идей для дизайна сайта \*ссылка\* с возможностью поделиться результатами по ссылке.

▪️Найди самый простой и быстрый вариант решения задачи по \*название предмета\* — пример.

▪️Составь план сочинения на тему \*название\* по произведению \*название\*.

▪️Расскажи, какой из языков программирования самый перспективный для изучения в \*уточнение\* году.

▪️Изучи новости за день и составь аналитический отчёт по \*количество\* параметрам в виде таблицы.

# Разные роли ChatGPT

Секретные возможности ChatGPT постепенно становятся явными и быстро набирают популярность. Одна из них — умение нейросети выступать в роли помощника в разных сферах. Чтобы попробовать, как работает эта функция, достаточно объяснить ChatGPT его роль и создать запрос. Это могут быть как классические варианты, например, менеджер по туризму или юрист, так и необычные — фокусник, дантист или сомелье.

• Шеф-повар — Я ищу человека, который разработает для меня вегетарианское меню, состоящее из разнообразных блюд. Мой первый запрос:«Что-нибудь сытное на ужин с расчетом на 2 порции».

• Визажист — Я хочу, чтобы ты был визажистом, умеющим работать с разными продуктами и текстурами, специализация — вечерний и дневной макияж. Моё первое пожелание: «Придумай образ для девушки 17-ти лет на выпускной вечер с учетом того, что у неё сухой тип кожи».

• Стилист — Мне нужен стилист. Я напишу свой бюджет, а ты предложишь 10 идей образов на закрытое мероприятие. Мой первый запрос: «Бюджет 7000 рублей, нужен образ на вечеринку в стиле 90-х».

• Врач — Стань моим личным врачом. Я опишу свои симптомы, а ты поставишь диагноз и выпишешь лечение. Первая тема: «Мне тяжело вставать по утрам, сильно сокращаются мышцы при занятиях спортом, волосы и ногти легко обламываются».

• Шахматист — Я хочу, чтобы ты был моим соперником по игре в шахматы.Мы будем говорить наши ходы по очереди. Первая партия — я за белых. Не объясняй мне свои ходы — мы соперники. Не забывай обновлять состояние доски после каждого хода. Мой первый ход — d3.

• Туристический гид — Я хочу, чтобы ты на время стал моим гидом. Я напишу свое местоположение, а ты посоветуешь место, которое можно посетить. Мой первый запрос: «Я нахожусь в Санкт-Петербурге и не хочу посещать музеи».

• Психолог — Мне нужен психолог. Я расскажу о ситуации в моей жизни, а ты дашь подсказки и укажешь, как действовать. Мой запрос: «Мне не хватает мотивации выполнять все важные дела. Что мне делать?».

• Писатель эссе — Я хочу, чтобы ты был писателем эссе. Исследуй заданную тему, сформулируй тезис и напиши качественную работу — информативную и увлекательную. Мой первый запрос: «Мне нужно написать эссе о важности сортировки мусора для экологии».

• UX/UI-разработчик — Мне нужен UX/UI-разработчик. Я расскажу о деталях дизайна приложения или веб-сайта. Твоя работа будет заключаться в том, чтобы придумать креативные способы улучшить пользовательский опыт. Это могут быть прототипы, тестирование различных вариантов дизайна, отзывы о лучших вариантах. Первый запрос: «Разработай дизайн для мобильного приложения туристической компании».

• Риэлтор — Я нуждаюсь в помощи риэлтора. Я буду писать информацию о человеке, который ищет подходящее жильё, твоя роль — помочь ему найти недвижимость, исходя из его бюджета, предпочтений в образе жизни, требований к местоположению. Важно включать в работу знания местного рынка жилья, чтобы предложить подходящую недвижимость. Первый запрос: «Найди двухэтажный дом для семьи из 5 человек недалеко от центра Праги».

# find\_ok\_url

<https://chatbot.theb.ai/#/chat/1684869237763>

Ты программист разработчик. Напиши код на Python.

Программа должна проверять ссылки на страницы сайта и находить рабочие ссылки.

Схема ссылки следующая: «https://lk.neural-university.ru/learning-program-v2/320370»

Где постоянная часть это: «https://lk.neural-university.ru/learning-program-v2/», а переменная часть: «320370».

Программа должна в цикле увеличить на единицу значение переменной части, сформировать ссылку и проверить ее, например: следующая переменная часть, увеличенная на единицу это «320371», новая ссылка это «https://lk.neural-university.ru/learning-program-v2/320371», при выполнение GET запроса по этой новой ссылке получен ответ «OK», значит ссылка рабочая. Выводим на экран рабочую ссылку.

Программа каждую рабочую ссылку выводит на экран.

В конце работы программа выводит список всех найденных рабочих ссылок.

В начале работы программа спрашивает у пользователя значение переменной части для начала и для окончания.

При достижении значения для окончания, программа останавливается и выводит путь к сохраненному файлу со списках всех найденный рабочих ссылок.

Измени код программы. Вместо проверки «response.status\_code == 200» нужно проверять в теле страницы отсутствие текста «error 404»

Ты программист разработчик. Напиши код на Python.

Программа должна проверять ссылки на страницы сайта и находить рабочие ссылки.

Схема ссылки следующая: «https://lk.neural-university.ru/learning-program-v2/320370»

Где постоянная часть это: «https://lk.neural-university.ru/learning-program-v2/», а переменная часть: «320370».

Для аторизации на этом сайте программа должна использовать login\_name = “change\_name” и password «12345».

Программа должна в цикле увеличить на единицу значение переменной части, сформировать ссылку и проверить ее, например: следующая переменная часть, увеличенная на единицу это «320371», новая ссылка это «https://lk.neural-university.ru/learning-program-v2/320371», при выполнение GET запроса по этой новой ссылке получена страница в теле которой нет текста «error 404», это значит ссылка рабочая. Выводим на экран рабочую ссылку.

Программа каждую рабочую ссылку выводит на экран.

В конце работы программа выводит список всех найденных рабочих ссылок.

В начале работы программа спрашивает у пользователя значение переменной части для начала и для окончания.

При достижении значения для окончания, программа останавливается и выводит путь к сохраненному файлу со списках всех найденный рабочих ссылок.

# Olds

Напиши код на Python, код должен выполнять следующий сценарий:

Запросить у пользователя файловый каталог,

В цикле обойти все файлы в указанном файловом каталоге и подкаталогах.

Для каждого файла добавить его содержимое в выходной файл указав к комментарии имя текущего файла.

После окончания цикла вывести содержимое выходного файла.

import os

dir\_path = input("Введите путь до файлового каталога: ")

output\_file\_path = "output.txt"

with open(output\_file\_path, "w") as output\_file:

for subdir, dirs, files in os.walk(dir\_path):

for file in files:

file\_path = os.path.join(subdir, file)

with open(file\_path, "r") as input\_file:

file\_contents = input\_file.read()

output\_file.write(f"=== {file\_path} ===\n")

output\_file.write(file\_contents + "\n")

with open(output\_file\_path, "r") as output\_file:

print(output\_file.read())

Пользователю предлагается ввести путь до файлового каталога. Затем используется функция os.walk для обхода всех файлов в указанном каталоге и подкаталогах. Для каждого файла открывается файловый поток на чтение, считывается содержимое файла, и имя файла добавляется в выходной файл в качестве комментария. После завершения цикла файлов выводится содержимое выходного файла на экран.

Напиши код на Python, код должен выполнять следующий сценарий:

Запросить у пользователя файловый каталог,

В цикле обойти все файлы в указанном файловом каталоге и подкаталогах. Для каждого файла:

Открой текущий файл, запомни его имя и путь, определи кодировку файла.

Сконвертируй в koi8 кодировку содержимое текущего файла и добавить это содержимое в выходной файл указав к комментарии имя и путь текущего файла.

После окончания цикла вывести содержимое выходного файла.

import os

import codecs

def **convert\_file**(input\_file, output\_file):

*# определяем кодировку*

with open(input\_file, 'rb') as f:

raw\_data = f.read()

encoding = 'utf-8'

try:

decoded\_data = raw\_data.decode('utf-8')

except UnicodeDecodeError:

encoding = 'windows-1251'

decoded\_data = raw\_data.decode(encoding)

except:

print(f"Unable to decode {input\_file}")

return False

*# конвертируем в koi8*

koi8\_data = codecs.encode(decoded\_data, 'koi8-r')

*# добавляем в выходной файл с комментарием*

with open(output\_file, 'ab') as f:

f.write(b"# " + input\_file.encode() + b"\n")

f.write(koi8\_data)

return True

def **process\_dir**(input\_dir, output\_file):

for root, dirs, files in os.walk(input\_dir):

for file\_name in files:

input\_file\_path = os.path.join(root, file\_name)

if convert\_file(input\_file\_path, output\_file):

print(f"Processed {input\_file\_path}")

if \_\_name\_\_ == '\_\_main\_\_':

input\_dir = input("Enter input directory: ")

output\_file = "output.koi"

*# создаем пустой выходной файл*

open(output\_file, 'wb').close()

process\_dir(input\_dir, output\_file)

*# выводим содержимое выходного файла*

with open(output\_file, 'rb') as f:

print(f.read().decode('koi8-r'))

<https://ai-talks.streamlit.app/>

Напиши код на Python, код должен выполнять следующий сценарий:

Запросить у пользователя файловый каталог и имя выходного файла

В цикле обойти все файлы в указанном файловом каталоге и подкаталогах. Для каждого файла:

Если текущий файл имеет расширение из списка: .json, .proto, .sql, .xml – тогда пропусти этот файл.

Для иных файлов:

Открой текущий файл, запомни его имя и путь, определи кодировку файла.

Сконвертируй в koi8 кодировку содержимое текущего файла и добавить это содержимое в выходной файл указав к комментарии имя и путь текущего файла.

После окончания цикла закрой выходной файл и сообщи полный путь к нему.

Покрой сгенерированный код комментариями.

. import os

import codecs

# Пользователь указывает файловый каталог и имя выходного файла

directory = input("Введите файловый каталог: ")

output\_file\_name = input("Введите имя выходного файла: ")

# Список расширений файлов, которые нужно пропустить

skip\_extensions = [".json", ".proto", ".sql", ".xml"]

# Открываем выходной файл на запись

with open(output\_file\_name, "w") as output\_file:

# Обходим все файлы в указанном каталоге и подкаталогах

for root, dirs, files in os.walk(directory):

for file in files:

# Получаем полный путь к текущему файлу

file\_path = os.path.join(root, file)

# Если текущий файл имеет расширение из списка skip\_extensions, тогда пропускаем его

if os.path.splitext(file\_path)[1] in skip\_extensions:

continue

# Открываем текущий файл

with codecs.open(file\_path, "r", encoding="utf-8", errors="ignore") as input\_file:

# Получаем содержимое текущего файла

content = input\_file.read()

# Получаем имя текущего файла

file\_name = os.path.basename(file\_path)

# Конвертируем содержимое текущего файла в koi8 кодировку

converted\_content = content.encode("koi8-r", errors="ignore").decode("koi8-r", errors="ignore")

# Добавляем содержимое в выходной файл

output\_file.write(f"# Имя файла: {file\_name}, Путь: {file\_path}\n")

output\_file.write(converted\_content)

output\_file.write("\n\n")

# Выводим полный путь к выходному файлу

print(f"Выходной файл создан по пути: {os.path.abspath(output\_file\_name)}")