100+ ChatGPT Programming Prompts

Explore the power of Chat GPT faster with these prompts.

▼ Implement an Algorithm from Scratch

Templates

Implement a [type of] algorithm in [Language]

Implement a [type of] algorithm in [Language] to [do something specific]

Examples

▼ Beginner

Implement a linear search algorithm in Python to find a specific value in an array.

Write a bubble sort algorithm in Java to sort an array of integers in ascending order.

Develop a binary search algorithm in C++ to search a sorted array for a specific value.

Implement a selection sort algorithm in JavaScript to sort an array of numbers.

Write a quick sort algorithm in Go to sort an array of strings in alphabetical order.

Develop a insertion sort algorithm in Python to sort an array of floats in descending order.

Implement a counting sort algorithm in R to sort an array of positive integers.

Write a merge sort algorithm in Java to sort an array of dates in chronological order.

Develop a shell sort algorithm in C++ to sort an array of characters in lexicographical order.

Implement a radix sort algorithm in JavaScript to sort an array of large integers.

Write a heap sort algorithm in Go to sort an array of arbitrary objects.

Develop a bucket sort algorithm in Python to sort an array of real numbers into uniform buckets.

▼ IDE Juggernaut

Implement a breadth-first search algorithm in Python to traverse a graph data structure.

Write a dynamic programming algorithm in Java to solve the knapsack problem.

Develop a depth-first search algorithm in C++ to search a binary tree data structure.

Implement an A* search algorithm in JavaScript to find the shortest path in a navigation map.

Write a greedy algorithm in Go to schedule tasks with conflicting requirements.

Develop a simulated annealing algorithm in Python to solve optimization problems.

Implement a decision tree algorithm in R for predictive modeling and classification.

Write a genetic algorithm in Java to find the global optimum in a complex search space.

Develop a neural network algorithm in TensorFlow for image classification.

Implement a clustering algorithm in MATLAB to group similar data points together.

Write a support vector machine algorithm in scikit-learn for binary classification problems.

Develop a reinforcement learning algorithm in PyTorch to train an agent in a simulation environment.

▼ Build a Simple Tool

Templates

Build a [tool concept] in [language]

Develop a [simple app idea] in [language] that [specify functionality]

Create a [tool concept] in [language] that [specify functionality]

Build a [tool concept] in [language] that [specify functionality]

Examples

Develop a code formatter in Python that formats and indents the given code according to a set of rules.

Build a regular expression evaluator in Java that takes a regular expression pattern and a string as inputs, and returns a boolean indicating whether the string matches the pattern.

Build a URL shortener in JavaScript that takes a long URL as input and returns a shortened URL.

Create a weather app in Ruby that displays the current weather conditions for a specified city.

Develop a to-do list application in Swift that allows users to add, edit, and delete tasks.

Build a calculator in Go that takes mathematical expressions as input and returns the calculated result.

Create a file compression tool in C# that compresses and decompresses files.

Build a unit converter in R that converts between different units of measurement.

Develop a color picker tool in Typescript that displays a color wheel and allows users to select a color and see its RGB value.

Build a spell checker in Python that takes a string as input and returns a list of spelling errors, if any.

Create a password generator in Java that generates random, secure passwords of a specified length.

Develop a currency converter in JavaScript that converts between different currencies based on real-time exchange rates.

▼ Expand your Coding Horizons

Templates

Explore a new [area or speciality] [library/package/language] that [functionality]

Build a [feature/tool] that demonstrates how to to work with [new library/package/language]

Examples

Write code snippets in Rust that demonstrate basic syntax, control structures, and data types.

Build a simple web application in Flask that displays a list of items and allows users to add, edit, and delete items.

Write a short program in a new functional programming language that implements a Fibonacci sequence.

Explore a new game development framework by creating a simple 2D game with graphics and user input.

Write code snippets in a new system programming language that demonstrate how to work with file I/O and process management.

Build a simple data analysis tool in a new data science framework that reads a CSV file, performs some basic computations, and generates a visual report.

Write a short program in a new web development framework that implements a basic RESTful API.

Explore a new machine learning library by creating a simple model that predicts the likelihood of a binary outcome based on some input data.

Write code snippets in a new database query language that demonstrate how to insert, update, and retrieve data from a database.

Build a simple chat application in a new real-time communication framework that allows multiple users to send and receive messages in real-time.

Write a short program in a new blockchain platform that implements a simple smart contract.

Explore a new virtual reality framework by creating a simple VR experience with graphics and user interaction.

▼ Write Basic + Advanced Functions

Templates

Write a function that takes [parameter type and description] as input and returns [type + functional explanation of output]

Write a function that takes [parameter type and description] as input and returns [type + functional explanation of output] in [language]

Write a function that takes [parameter type and description] as input and [transformation] and returns [type] in [language + library]

Examples

▼ For Beginners

Write a function that takes two integers as input and returns their sum.

Build a function that takes a list of numbers as input and returns the average.

Write a function that takes a string as input and returns the number of vowels in the string.

Write a function that takes a list of strings as input and returns the longest string.
Write a function that takes a positive integer as input and returns its factorial.

Write a function that takes a string and a substring as input and returns the number of occurrences of the substring in the string.

Write a function that takes two strings as input and returns a string that is the concatenation of the input strings.

Write a function that takes a list of integers as input and returns the second largest integer.

Write a function that takes a positive integer as input and returns a list of its divisors.

Write a function that takes a list of numbers as input and returns a list of the squares of the numbers.

Write a function that takes a string as input and returns a string with all the letters in reverse order.

Implement a function that takes a list of numbers as input and returns the median.

▼ For Seasoned Key-slappers

Write a function that takes a list of numbers as input and returns the mode.

Implement a function that takes a string as input and returns the longest palindromic substring.

Write a function that takes a list of strings as input and returns the list of anagrams.

Implement a function that takes a list of numbers as input and returns the largest contiguous sum.

Write a function that takes a list of integers as input and returns a list of all the prime numbers.

Implement a function that takes a string as input and returns a boolean indicating whether the string is a valid IP address.

Write a function that takes two strings as input and returns the edit distance between them.

Implement a function that takes a list of numbers as input and returns the longest increasing subsequence.

Write a function that takes a string as input and returns the longest common prefix among all the words in the string.

Implement a function that takes a graph as input and returns the shortest path between two nodes.

Write a function that takes a list of numbers as input and returns a list of all the combinations of the numbers that add up to a target sum.

Implement a function that takes a list of strings as input and returns the longest common substring among all the strings.

▼ Dive into Cutting-Edge programming

Templates

Write a program that [does something impressive]

Write a program that [does something impressive] in [language]

Write a program that [does something impressive] in [language] using [technique/library]

Examples

Write a program that performs image recognition using computer vision techniques.

Develop a natural language processing tool that can extract named entities from text.

Write a program that can generate new music using machine learning techniques.

Develop a recommendation system that suggests movies or TV shows based on a user's viewing history.

Write a program that implements a deep reinforcement learning algorithm to play a game.

Develop a web application that allows users to perform sentiment analysis on social media posts.

Write a program that generates new text using a GPT-3-like model.

Develop a tool that can automatically grade code submissions based on specific criteria.

Write a program that implements a generative adversarial network (GAN) to generate new images.

Develop a chatbot that can answer questions about a specific domain using a knowledge graph.

Write a program that implements a variational autoencoder (VAE) to generate new data points.

Develop a platform for hosting and sharing interactive programming tutorials.

▼ Build an App with any Tech Stack

If you want to ACTUALLY get ChatGPT to build you an REAL app you're proud of, documented and tested, I recommend viewing **Prompt Engineering for Programmers**.

Templates

Develop a [app description] using [tech stack]

Build a [app description] using [tech stack]

Write a [app description] that [does] with [libraries/packages]

Examples

▼ Javascript

Develop a real-time chat application using Node.js, Express, and Socket.io.

Implement a weather app that displays real-time weather information for a user's location using the OpenWeatherMap API and Typescript.

Develop a platform for hosting and sharing code snippets that allows users to view and comment on each other's code using Meteor and MongoDB.

Implement a recommendation system that suggests items based on a user's previous purchases or interests using the TensorFlow.js library.

Develop a scheduling app that allows users to create and manage appointments and events using Vue.js and firebase.

Implement a chatbot that can respond to user inquiries in a conversational manner using Dialogflow and Node.js.

Develop a recipe database and search engine that allows users to search for recipes based on ingredients or dietary restrictions using Vue.js and a MongoDB database.

Implement a task management app that allows users to create, manage, and track their to-do lists using jQuery and a MySQL database.

Develop a job board that allows users to search and apply for job postings using Node.js, Express, and Elasticsearch.

Implement a budgeting app that tracks a user's expenses and income to help them stay within their budget using Next.js and a NoSQL database.

Develop a music player app that allows users to search, play, and save songs using Angular and a NoSQL database.

Implement a real-time voting system that allows users to vote on various topics using Node.js, Express, and an EOSIO contract to store the data in a table using EOSJS.

▼ React (Popular Javascript UI Library)

Develop a dashboard that displays real-time stock market data using React components and Chart.js.

Write a program that implements a to-do list application using React components and Material-UI.

Develop a weather app that displays real-time weather information for a user's location using React components and Ant Design.

Write a program that implements a recipe database and search engine using React components and Semantic UI.

Develop a scheduling app that allows users to create and manage appointments and events using React components and Blueprint.

Write a program that implements a job board using React components and PrimeNG.

Develop a budgeting app that tracks a user's expenses and income using React components and Shards.

Write a program that implements a chatbot using React components and Evergreen.

Develop a platform for hosting and sharing code snippets using React components and Element.

Write a program that implements a recommendation system for movies or TV shows using React components and Tachyons.

Develop a task management app using React components and Blueprint.

Write a program that implements a music player app using React components and Ant Design.

▼ Python

Develop a web scraper that can scrape data from a specific website and store it in a database using Beautiful Soup and SQLAlchemy.
Implement a simple blogging platform that allows users to create, edit, and view blog posts using Flask and Jinja.
Develop a weather app that displays real-time weather information for a user's location using Django and the OpenWeatherMap API.
Implement a recommendation system that suggests items based on a user's previous purchases or interests using Pandas and Scikit-learn.
Develop a scheduling app that allows users to create and manage appointments and events using Pyramid and SQLAlchemy.
Implement a chatbot that can respond to user inquiries in a conversational

manner using ChatterBot and Flask.
Develop a recipe database and search engine that allows users to search for recipes based on ingredients or dietary restrictions using Django and PostgreSQL.
Implement a task management app that allows users to create, manage, and track their to-do lists using Flask and SQLite.
Develop a platform for hosting and sharing code snippets that allows users to view and comment on each other's code using Django and PostgreSQL.
Implement a job board that allows users to search and apply for job postings using Flask and Elasticsearch.
Develop a budgeting app that tracks a user's expenses and income to help them stay within their budget using Django and Django Rest Framework.
Implement a music player app that allows users to search, play, and save songs using Flask and SQLite.

▼ PHP

Develop a web scraper that can scrape data from a specific website and store it in a database using the PHP Simple HTML DOM Parser library and the Doctrine ORM library.
Implement a weather app that displays real-time weather information for a user's location using the OpenWeatherMap API and the Laravel framework.
Implement a recommendation system that suggests items based on a user's previous purchases or interests using the PHP Machine Learning library.
Develop a scheduling app that allows users to create and manage appointments and events using the Symfony framework and a SQL database.
Implement a chatbot that can respond to user inquiries in a conversational manner using the Doctrine ORM library and the Slim framework.
Develop a recipe database and search engine that allows users to search for recipes based on ingredients or dietary restrictions using the Doctrine ORM library and a SQL database.
Implement a task management app that allows users to create, manage, and track their to-do lists using the Slim framework and a SQL database.
Develop a platform for hosting and sharing code snippets that allows users to view and comment on each other's code using the Laravel framework and a MySQL database.

Implement a job board that allows users to search and apply for job postings using the Slim framework and Elasticsearch.
Develop a budgeting app that tracks a user's expenses and income to help them stay within their budget using the Laravel framework and an Oracle database.
Implement a music player app that allows users to search, play, and save songs using the Symfony framework and a NoSQL database.
Develop a real-time voting system that allows users to vote on various topics using the Slim framework and a PostgreSQL database.

▼ Go

Implement a web scraper that can scrape data from a specific website and store it in a database using the Goquery library and the GORM library.
Develop a weather app that displays real-time weather information for a user's location using the OpenWeatherMap API and the Echo framework.
Implement a recommendation system that suggests items based on a user's previous purchases or interests using the Golearn library.
Develop a scheduling app that allows users to create and manage appointments and events using the Gorilla Mux library and a SQL database.
Implement a chatbot that can respond to user inquiries in a conversational manner using the GORM library and the Gin framework.
Develop a recipe database and search engine that allows users to search for recipes based on ingredients or dietary restrictions using the GORM library and a SQL database.
Implement a task management app that allows users to create, manage, and track their to-do lists using the Echo framework and a SQL database.
Develop a platform for hosting and sharing code snippets that allows users to view and comment on each other's code using the Gin framework and a NoSQL database.
Implement a job board that allows users to search and apply for job postings using the Echo framework and Elasticsearch.
Develop a budgeting app that tracks a user's expenses and income to help them stay within their budget using the Gin framework and a NoSQL database.
Implement a music player app that allows users to search, play, and save

songs using the Echo framework and a NoSQL database.
Develop a real-time voting system that allows users to vote on various topics using the Gin framework and a NoSQL database.

▼ Solidity

Develop a decentralized voting system on the Ethereum network to securely record and count votes.

Implement a token-based rewards program for an online platform that incentivizes users to engage with content and perform specific actions.

Develop a decentralized marketplace for buying and selling goods and services using smart contracts and the Ethereum network.

Implement a prediction market platform that allows users to bet on the outcome of various events using a token-based system.

Develop a decentralized crowdfunding platform that allows individuals to pool funds to support a specific cause or project.

Implement a decentralized identity management system that allows individuals to securely store and control access to their personal information.

Develop a peer-to-peer lending platform that facilitates direct loans between borrowers and lenders without the need for a traditional financial institution.

Implement a decentralized gaming platform that allows players to compete and win tokens through various games and challenges.

Develop a decentralized property ownership and transfer platform that records and tracks property ownership on the Ethereum network.

Implement a decentralized reputation management system that allows individuals and organizations to build and maintain a reputation based on their actions and interactions.

Develop a decentralized data storage platform that allows individuals and organizations to securely store and access their data without the need for a centralized server.

Implement a decentralized oracle system that allows smart contracts to securely access and utilize external data and information sources.

▼ Sharpen your Programming Knowledge

Templates

Explain [programming concept]

Explain [programming concept] in [context]

Explain the difference between [concept 1] and [concept 2]

Examples

Explain the difference between object-oriented programming and functional programming.

Discuss the benefits and limitations of using microservices architecture in software development.

Describe how cryptography can be used to secure data in cloud computing.

Explain the concept of serverless computing and its potential impact on web development.

Discuss the role of data structures and algorithms in software development.

Explain the difference between API and microAPI.

Discuss the use cases and benefits of containerization in software development.

Explain how machine learning and artificial intelligence are being used in software development today.

Discuss the benefits and challenges of using decentralized systems in software development.

Explain how software design patterns can be used to simplify and improve the development process.

Discuss the impact of quantum computing on software development and cryptography.

Explain the role of software testing and quality assurance in software development.

> “If you give a man a fish, you feed him for a day. If you teach a man to fish, you feed him for a lifetime.”

That’s all for now!

These prompts were designed for learning the capabilities of ChatGPT. They won’t teach you to fish. They haven’t been optimized. And I left out God Mode, my personal favorite template to prompt ChatGPT and other AI models.

🎣 Learn to Fish: **Prompt Engineering for Programmers**

Want to share with friends? Please share this link