# ****Python Chaser - Official Documentation****

## ****Introduction****

**Python Chaser** is a fast-paced, snake-inspired multiplayer game where players race to collect food and power-ups while avoiding obstacles and outmaneuvering their opponents. Designed with both casual and competitive players in mind, Python Chaser supports **Player vs. Player (PVP)** and **Player vs. Environment (PVE)** gameplay, featuring an AI opponent powered by the **A\*** pathfinding algorithm. The game is built using **Python** and **Pygame**, with interactive menus, music, and visual effects to create an exciting and dynamic experience.

Whether you're facing off against a friend or challenging a clever AI, Python Chaser delivers engaging gameplay with strategic elements like power-ups and time-based rounds.

## ****Features****

* 🧠 **Smart AI Opponent**: Intelligent enemy using A\* pathfinding to chase food and evade collisions.
* 🎮 **Multiplayer Modes**: Play head-to-head with a friend or compete solo against the bot.
* ⏱ **Time-Based Matches**: Choose from 1, 2, or 3-minute game durations to fit your playstyle.
* ⚡ **Power-Ups**:
  + **Speed Boost (Lightning Bolt)** – Dash quickly to gain an edge.
  + **Freeze (Snowflake)** – Temporarily immobilize your opponent.
* 🎵 **Audio Options**: Toggle in-game music and sound effects on or off.
* 🖥 **Interactive UI**: Smooth transitions, menus, and clear in-game visuals.

## ****How to Play****

1. Choose a game mode from the main menu: **Player vs. Player** or **Player vs. Bot**.
2. Use the keyboard to control your snake:
   * **Player 1 (Red Snake)** uses **WASD** keys.
   * **Player 2 (Blue Snake)** uses **Arrow** keys.
3. **Collect green food** scattered on the grid to increase your score.
4. **Pick up power-ups** to gain special effects:
   * ⚡ **Lightning Bolt**: Grants a speed boost for 5 seconds.
   * ❄️ **Snowflake**: Freezes the opposing player or AI for 3 seconds.
5. Avoid crashing into walls, yourself, or your opponent.
6. The game ends when the timer runs out. The player with the **highest score wins**!

## ****Controls****

### Player 1 (Red Snake)

W - Move Up

A - Move Left

S - Move Down

D - Move Right

### Player 2 (Blue Snake)

↑ - Move Up

← - Move Left

↓ - Move Down

→ - Move Right

### General Controls

M - Toggle background music

Q - Quit the game

Space - Progress through menus or restart

## ****Requirements****

To run Python Chaser, ensure the following are installed:

* Python **3.x**
* **Pygame** library
* (Optional) **Git** – for pushing code to GitHub or version control

## ****Installation****

Follow these steps to get started:

1. Download and install **Python 3.x** from the [official Python website](https://www.python.org/).
2. Install Pygame by running this command in your terminal or command prompt:

pip install pygame

1. Clone the project repository or download the ZIP file:

git clone https://github.com/Thatoe98/Python-Chaser.git

1. Navigate to the project folder and run the game:

python pChaser\_1.0.py

## ****Project Structure****

pChaser\_1.0.py # Main script and game launcher

settings.py # Game configurations (speed, duration, colors)

game\_mechanics.py # Handles snake movement, food spawning, collisions

pathfinding.py # AI logic and A\* implementation

game\_screens.py # UI components: menus, win/loss screens

ui\_elements.py # Drawing utilities for buttons, text, etc.

push\_to\_github.py # Automates pushing the project to GitHub

generate\_docs.py # Auto-generates this documentation

## ****Power-Ups****

Power-ups add a strategic twist to the game, allowing players to shift momentum:

* **Speed Boost (⚡ Lightning Bolt)**  
  Temporarily increases snake speed, helping players reach food or escape tight situations. Lasts for **5 seconds**.
* **Freeze (❄️ Snowflake)**  
  Stops the opponent from moving for **3 seconds**, giving you a chance to grab food unchallenged.

Power-ups respawn periodically and are randomly placed on the grid.

## ****AI Pathfinding****

The AI-controlled snake uses the A (A-star)\* algorithm to calculate the shortest, most efficient route to the nearest food. The AI accounts for dynamic obstacles such as:

* Walls and map boundaries
* Its own body and the player's snake
* Frozen states when affected by power-ups

This makes the AI challenging and unpredictable, simulating real-time strategic decisions.

## ****Music and Sound Effects****

For an immersive gaming experience, Python Chaser includes:

* Background music during gameplay
* Sound effects for eating food, activating power-ups, and winning/losing

Use the **'M' key** to mute or unmute music at any time during the game.

## ****Credits****

Python Chaser was created by **Team Pythonic**, with inspiration and support from the **Pygame** community. Special thanks to:

* Our professor Aj. Hyunjeb
* Open-source contributors who shared examples and tutorials
* Beta testers who helped refine the mechanics
* The broader Python and game dev communities