Pop the balloon game

1.Purpose behind the research

Purpose of the research:

The purpose behind the project of **Pop the Balloon** game was to explore how interactive and event-driven applications can be designed using Python's **Tkinter** library. The research focused on understanding real-time GUI programming and how user actions can directly influence dynamic elements on the screen. This project aims to study essential concepts such as event handling, canvas-based graphics, timers, object creation and deletion, and management of different interface states within a single application.

2.Data collected

During development and testing of the **Pop The Balloon** game, various gameplay metrics and behavior patterns were observed to improve user experience, performance, and difficulty balancing

- **Balloon spawn rate patterns**
  (how quickly balloons appear and disappear over time)
- **Player reaction time**
  (how fast the player clicks balloons after they appear)
- **Score patterns**
  (average balloons popped within the 60-second duration)
- **Balloon click accuracy**
  (correct click hits vs. missed clicks)
- **Lag or frame-drop points**
  (game speed reduction when too many balloons spawn)

- **User interaction behavior**
  (click frequency and movement across the screen)

This data helped to improve the game

- Difficulty balancing
- Balloon spawn rate
- Visual smoothness
- Game fairness

Data Description

This project does **not use external datasets**, but it generates its own **runtime data** during gameplay. The data is created dynamically by the game logic and is used to update visuals, scoring, user interaction, balloon creation, and time tracking.
Below is a detailed breakdown of the data involved:

1. Game State Data

These values are stored internally in variables and updated continuously throughout the game:

Variable	Type	Description
`score`	Integer	Tracks how many balloons the player has successfully popped
`time_left`	Integer	Countdown timer showing remaining game duration (in seconds)
`game_running`	Boolean	Indicates whether the game is active, paused, or finished

2. Balloon Data

Balloons are not stored permanently in files; they are generated randomly and tracked temporarily in memory.

Each balloon is represented by:

- A **Canvas object ID**
- **Random X-position** and **Random Y-position**
- **Random size** (radius between 20–40 pixels)
- **Random color** (red, yellow, green, pink, purple)
- **Mouse click binding** that triggers score updates

These balloons are deleted once they are clicked by the player or when the game ends.

3. Event-Generated Data

During the gameplay, the following data is produced and updated dynamically:

- Number of balloons spawned
- Number of balloons popped vs. missed
- Time countdown progression
- User click interactions

4. Data Creation Process

All data is generated programmatically inside the script:

- The `spawn_balloon()` function creates balloon objects with random values
- The `pop_balloon()` function removes balloons and updates score
- The `update_timer()` function generates timer countdown data
- The game state variables update continuously during real-time play

How to Use

Follow these steps to install, run, and use the Catch the Balls game:

1. Download or Clone the Repository

git clone https://github.com/veerj0808/catchTheBalls.git

Or download the ZIP file and extract it.

2. Install Requirements

This game uses Python's built-in Tkinter library, so no external packages are required.

Make sure you have:

● Python 3.8 or above

● Tkinter installed (included by default in most Python installations)

3. Run the Game Navigate to your project folder and run:

python pop_the_ballon_game.py

4. Gameplay Instructions

● pop the balloons to score points.

● Avoid missing the balloons.

● You have 20 seconds to score as high as possible.

- Use Pause/Resume button to control gameplay.
- After time runs out, the final score will appear with a replay option.

Contact Information

If you have questions, suggestions, or want to collaborate, feel free to reach out:

Name: Anshika Singh

Email: anshika0singh099@gmail.com

GitHub:

Contact Purpose: Bug reports, game improvements, feature requests, or general queries.

Lisence

This project is released under the MIT License, which allows users to freely use, modify, distribute, and share the code as long as proper credit is given to the original author.

You are free to:

- Use the code for personal, academic, or commercial projects
- Modify or expand the game
- Distribute your version

You must:

- Include the original copyright notice
- Include the MIT License in any redistributed version

For full legal details, see the included LICENSE file.

Notes

● This project is a simple **Python–Tkinter–based balloon popping arcade game** developed for learning and demonstration purposes.

● All graphics (balloons, buttons, canvas) are created using **basic Tkinter shapes and widgets — no external images, assets, or libraries are required.**

● The game is designed and optimized to run smoothly on most systems **without any additional dependencies** beyond standard Python and Tkinter.

- You can customize:
  1. Balloon spawn speed
  2. Balloon colors
  3. Balloon sizes
  4. Timer duration
  5. Difficulty level (faster balloon generation or more frequent spawns)

● The code is intentionally written in a **clean and modular structure**, making it easy for students to modify for college projects, UI learning, or further research.

● The project is suitable for **beginner-level game development, GUI learning, and academic mini-projects**.

● Feedback, suggestions for new features, and improvements are welcome!