

**Session (2024-25)**

**MINI PROJECT REPORT**

**On**

**SNAKE GAME WITH 2D FACE USING PYTHON**

**Bachelor of Technology**

**(COMPUTER SCIENCE AND ENGINEERING)**

**Second Year**

**Submitted to-**

**Ms. Himanshi**

**Submitted By-**

**Rahul Chauhan ( Roll no. 2303330100090)**

**Kuldeep Singh ( Roll no. 2303330100066)**

**Harsh Mishra ( Roll no. 2303330100054)**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

1. **Abstract : Snake Game with 2D Face**

**This project is a modern implementation of the classic Snake Game using Python and the Pygame library, enhanced with a visually appealing 2D cartoon face on the snake's head. The game provides an engaging and interactive platform for players, combining traditional gameplay with fun and creative graphics.**

**The primary objective is to control the snake using arrow keys to collect food items while avoiding collisions with the screen boundaries or the snake's own body. As the snake consumes food, it grows longer, increasing the challenge of navigation and spatial awareness. The player’s score is calculated based on the length of the snake.**

**Key features include:**

1. **Dynamic Gameplay: Real-time snake movement, responsive controls, and food spawning at random locations.**
2. **Cartoon-Style Snake Face: The snake's head is adorned with a 2D face, featuring expressive eyes and a tongue, adding a unique and playful aesthetic.**
3. **Collision Mechanics: The game ends upon collisions with boundaries or the snake itself, prompting the player to restart or quit.**
4. **Score System: The current score is displayed on the screen, encouraging players to achieve higher scores.**

**The game is implemented with modular and reusable code, leveraging functions for tasks such as drawing the snake, updating the score, and handling game-over scenarios. It also incorporates user-friendly controls and a restart option for improved accessibility.**

**This project serves as a foundation for exploring game development concepts, including graphics rendering, real-time input handling, and collision detection. It can be further expanded with additional features such as levels, obstacles, and sound effects, making it a versatile and enjoyable coding project for developers and players alike.**

1. **Introduction : Snake Game with 2D Face**

**The Snake Game is a classic arcade game that has captivated players for decades with its simple yet addictive mechanics. This project recreates the traditional gameplay using Python and the Pygame library, while introducing a creative touch—a 2D cartoon-style face on the snake's head. Players control the snake using arrow keys, guiding it to collect food that randomly appears on the screen. With each food item consumed, the snake grows longer, increasing the difficulty of avoiding collisions with the screen boundaries or its own body.**

**The game features smooth animations, dynamic food placement, and a real-time scoring system displayed on the screen. The playful addition of the 2D face, complete with expressive eyes and a tongue, adds a unique and engaging visual element, making the game both fun and appealing. Designed with modular code, the project emphasizes essential game development concepts such as graphics rendering, user input handling, collision detection, and game loop mechanics.**

**This Snake Game provides a nostalgic yet refreshing gaming experience while serving as an excellent learning project for developers interested in exploring the fundamentals of 2D game development and Python programming.**

**3. Literature Survey : Snake Game with 2D**

**Face**

**The Snake Game has been a staple of arcade gaming since the late 1970s, with early versions appearing on basic computer systems and mobile devices like Nokia phones. As graphics and programming languages advanced, Snake games incorporated more features, such as enhanced visuals, levels, and obstacles. Pygame, a popular Python library for game development, made it easier to develop 2D games with rich graphics and user interactions, making it an ideal tool for recreating Snake in modern programming environments.**

**Recent adaptations of the game have focused on improving player engagement through creative visual elements, such as adding expressive 2D faces to the snake’s head. This not only adds a playful aesthetic but also increases the game's overall appeal. The implementation of efficient collision detection and smooth game loops is crucial for ensuring that the game runs seamlessly, especially as the snake grows longer.**

**In addition to graphics, modern Snake games have explored adding sound effects and background music, further enhancing player immersion. These trends reflect how traditional games can be modernized and improved by leveraging current programming tools and design principles, maintaining the game's core mechanics while offering an engaging and enjoyable experience.**

**4. Problem Statement :**

"Design and develop a Snake Game with a 2D face as the snake's head using Python and Pygame. The game should allow players to control the snake's movement, grow when consuming food, and terminate upon collision with walls or the snake's body. The objective is to provide an engaging and visually appealing gaming experience while demonstrating fundamental game development principles, including event handling, collision detection, and graphical rendering."

**Objectives are as follows :**

1. **Game Mechanics:**

* **Implement smooth snake movement using keyboard controls.**
* **Ensure the snake grows when it consumes food.**
* **Detect collisions with the walls and the snake's body to end the game.**

1. **Enhancements:**

* **Represent the snake's head as a 2D face with eyes and a mouth.**
* **Use colors and simple shapes to create an appealing design.**

1. **User Experience:**

* **Randomly generate food positions on the screen.**
* **Provide immediate feedback to the player upon collisions or consumpsion.**

1. **Technical Requirements:**

* **Use Python programming language and Pygame library for development.**
* **Maintain a clean and efficient code structure for future scalability.**
* **This project is a fun way to introduce basic game development concepts while adding creative twist with 2d face design.**

1. **Implementation and Results : Snake Game**

**with 2d Face**

**Implementation :**

**The Snake Game was implemented using Python and Pygame. Key components include:**

* **Snake Movement: Controlled by arrow keys, with the snake growing as it eats food.**
* **2D Face: A cartoon face was added to the snake’s head, with eyes and a tongue for visual appeal.**
* **Food Generation: Food randomly appears, and the snake grows when it eats.**
* **Collision Detection: Ends the game if the snake hits the wall or itself.**
* **Score Tracking: Displays the player’s score based on the snake's length.**
* **Game Over & Restart: Players can restart the game or quit after a collision.**

**Results :**

**The game runs smoothly with real-time movement and proper collision handling. The 2D face on the snake adds a fun visual element, making the game more engaging. The game correctly resets or ends based on player input, and the score updates as expected.**

1. **Conclusion :**

**The Snake Game with a 2D face is an engaging project that showcases how foundational programming concepts can be applied to create a fun and interactive application. Using Python and the Pygame library, the game successfully combines entertainment with education, making it an ideal project for beginners and intermediate developers. This project not only introduces the core mechanics of a classic snake game but also enhances its appeal by integrating a 2D face design as the snake’s head, adding a unique and creative visual element.**

**From a development perspective, this project emphasizes several key programming principles:**

1. **Event Handling: The game's responsiveness to keyboard inputs highlights the importance of capturing and processing user actions in real-time. This is a critical skill for creating interactive applications.**
2. **Game Logic: The implementation of snake movement, collision detection, and food consumption showcases logical thinking and the ability to simulate real-world dynamics in code.**
3. **Graphics and Rendering: Leveraging Pygame's drawing capabilities to create visually appealing elements such as the snake's body, face, and the game grid introduces developers to 2D game graphics and animations.**
4. **Problem-Solving: Challenges such as randomizing food placement, ensuring smooth movement, and handling game-over conditions require analytical thinking and debugging skills.**

**Beyond technical skills, this project encourages creativity and innovation. By designing the snake’s head as a 2D face with eyes and a mouth, the game stands out from traditional snake games, showcasing how small visual tweaks can significantly enhance user engagement.**

**This project is an excellent example of merging functionality with fun, making it perfect for beginners and hobbyists. You can expand this project further by adding features like obstacles, different levels, or a scoring system to make the game more challenging and enjoyable.7. Reference :**

**Here are some references and resources that you can explore to learn more and enhance the Snake Game project:**

1. **Pygame Documentation:  
   Official documentation for pygame, providing detailed explanations of modules, methods, and examples.**
2. [**Python.org**](https://www.python.org/)**:  
   Official Python documentation and tutorials, useful for understanding Python basics and advanced features.**
3. **Real Python – Pygame Tutorial:  
   A beginner-friendly guide to creating games using pygame.**
4. **YouTube Tutorials:  
   Search for tutorials like *"Snake Game using Python and Pygame"* for step-by-step video guides.**
5. **GitHub Projects:  
   Explore open-source snake game projects on GitHub for ideas and code references. Search for repositories like *"snake-game-python"*.**
6. **Books on Game Development:**
   * ***Making Games with Python & Pygame* by Al Sweigart (freely available online).**
   * ***Python Game Programming by Example* by Alejandro Rodas de Paz.**

**These resources will help you understand the tools and concepts behind game development and allow you to expand your project creatively.**