**Internship Report on**

# DESIGNING SNAKE GAME USING PYTHON

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR, ANANTHAPURAMU**

*In Partial Fulfillment of the Requirements for the Award of the degree of*

## BACHELOR OF TECHNOLOGY

**In**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted By**

## B VIVEK - 21691A05P1



**MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE**

**(UGC – AUTONOMOUS)**

**(Affiliated to JNTUA, Ananthapuramu)**

**(Accredited by NBA, Approved by AICTE, New Delhi) AN ISO 9001:2015 Certified Institution**

**P. B. No: 14, Angallu, Madanapalle – 517325**

**2023 - 24**

**2009**

**-**

**2013**



### DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

**BONAFIDE CERTIFICATE**

This is to certify that the internship work entitled **“Snake Game Using Python”** is a bonafide work carried out by

#### B VIVEK - 21691A05P1

Submitted in partial fulfillment of the requirements for the award of degree **Bachelor of**

**Technology** in the stream of **Computer Science & Engineering** in **Madanapalle**

**Institute of Technology & Science, Madanapalle,** affiliated to **Jawaharlal Nehru Technological University Anantapur, Ananthapuramu** during the academic year 20232024.

**Internship Coordinator Head of the Department**

Dr. K. Sudhakar, Dr. R. Kalpana,

Senior Assistant Professor, Professor & Head,

Department of CSE Department of CSE

**Submitted for the University Examination held on:**

**Examiner - I Examiner - II ACKNOWLEDGEMENT**

We sincerely thank the **MANAGEMENT** of **Madanapalle Institute of**

**Technology & Science** for providing excellent infrastructure and lab facilities that helped me complete this Project.

We sincerely thank **Dr. C. Yuvaraj, M.E., Ph.D., Principal,** for guiding and providing facilities for completing our Project at **Madanapalle Institute of Technology & Science,** Madanapalle.

We express our gratitude to **Dr. R. Kalpana, Ph.D., Professor and Head of the Department of CSE** for her continuous support in making necessary arrangements for the successful completion of the Project.

We express our sincere thanks to the **Internship Coordinator, Dr. K. Sudhakar, Senior Assistant Professor, Department of CSE** for his tremendous support for the successful completion of Project.

We express our deep gratitude to our Internship In-Charge **Mr.K.Sathish, M.TECH., Assistant Professor, Department of CSE,** for his guidance and

encouragement that helped us to complete this Project.

We also wish to place on record my gratefulness to other **Faculty of CSE Department** and our friends and our parents for their help and cooperation during our project work.



##### DECLARATION

We hereby declare that the results embodied in this internship **"Designing Snake Game Using Python"** by us under the guidance of **Mr.K.Sathish,** in partial fulfillment of the award of **Bachelor of Technology** in **Computer Science & Engineering** from

**Jawaharlal Nehru Technological University Anantapur, Ananthapuramu**.

**Date :**

**Place : Anagallu**

**Internship Associate:**

**B Vivek**

**21691A05P1**

**ABSTRACT**

The snake Game project is composed in Python. The project file contains python script (snake.py). This is a simple GUI based method game which is developed using python's package named pygame. In gameplay, it's a single player game, where the player (Snake) has to eat all the fruits in order to grow snake longer.After starting the game, a GUI appears. Then the player has to control the snake. The mainthing in this GUI based game is that the player just has to press "arrow keys" in the keyboard to move the snake. At the top of the board, it displays a current player score. The player needs to catch the maximum number of fruits without hitting the wall or itself. All the game movements are to be performed manually by the player. While playing the game, make sure the snake should not cross the edges and should not snake touches its own tail in the game. Otherwise, player will die.The game also consists of a high score module where the winner is assigned specific points based on the ranking and is stored in a database. The score is parsed from the variable to the SQL database.The 2D GUI is designed using Pygame python. The gameplay design is simple and clean that the user won't find it difficult to use and understand. For the gaming environment, the project uses various images to provide real-time experience.

##### CONTENTS

|  |  |  |
| --- | --- | --- |
| **S.NO.** | **TOPIC** | **PAGE NO.** |
| **1** | **INTRODUCTION** |  |
|  | 1.1 About Python Programming | 2 – 3 |
|  | 1.2 Importance and Applications of  Python Programming | 3 – 4 |
|  | 1.3 Python as the Programming Language  Choice | 5 |
|  | 1.4 Why Create a Python Snake Game | 6 |
|  |  |  |
| **2** | **TOOLS AND TECHNIQUES** |  |
|  | 2.1 Python and Pygame Library | 8 |
|  | 2.2 Hardware Requirements (if any) | 9 |
|  | 2.3 Software Requirements | 10 |
|  |  |  |
| **3** | **PROJECT WORK** |  |
|  | 3.1 Project overview | 11 - 12 |
|  | 3.2 Algorithm | 12 |
|  |  |  |
| **4** | **CODE AND OUTPUT SCREENSHOTS** | 15 - 21 |
|  |  |  |
| **5** | **CONCLUSION** | 22– 23 |
|  |  |  |
|  | **REFERENCES** | 24 - 25 |

## CHAPTER-1

**INTRODUCTION**

**1**

**1.1 About Python Programming**

**Python**:

The game is developed using Python. Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures. combined with dynamic typing and dynamic binding. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed.

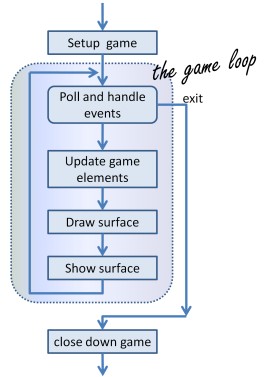
**Tkinter**:

Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is most commonly used method. It is a standard Python interface to the TK GUI toolkit shipped with Python. Python with tkinter outputs the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an casy task. Tkinter includes a number of Python modules. The two most important modules are the Tkinter module itself. and a module called Tkconstants. The former automatically imports the latter, so to use Tkinter, all you need to do is to import one module.

**Pygame**:

Game programming is very rewarding nowadays and it can also be used in advertising and as a teaching tool too. Game development includes mathematics, logic, physics, Al, and much more and it can be amazingly fun. In python, game programming is done in pygame and it is one of the best modules for doing so..

2



**Fig. 1.1 Python Programming**

**1.2 IMPORTANCE AND APPLICATIONS OF PYTHON PROGARAMMING**

Python programming is of significant importance and has a wide range of applications in various fields due to its simplicity, versatility, and a rich ecosystem of libraries and frameworks. Here are some key areas where Python plays a crucial role:

1. **Web Development**: Python is widely used for web development. Frameworks like Django and Flask make it easier to create web applications, and Python's versatility allows developers to work on both the front-end and back-end.
2. **Data Science and Machine Learning**: Python has become the primary language for data science and machine learning. Libraries like NumPy, pandas, and scikit-learn are essential for data analysis, while TensorFlow and PyTorch are popular choices for deep learning and neural network development.
3. **Scientific** **Computing**: Python is used in scientific research and engineering fields for tasks like simulations, modeling, and data analysis. Libraries such as SciPy and Matplotlib are essential for scientific computing.

3

1. **Artificial** **Intelligence** and Natural Language Processing: Python is the language of choice for AI and NLP projects. Libraries like NLTK and spaCy are used for natural language processing, while OpenAI's GPT-3 is powered by Python.
2. **Game Development**: Python, along with libraries like Pygame, is used to create video games and educational games, making it accessible for game developers, including those new to programming.
3. **Automation** **and** **Scripting**: Python is perfect for automating repetitive tasks and writing scripts. It's often used for tasks like web scraping, data extraction, and automating system administration.
4. **Cybersecurity**: Python is used for ethical hacking and penetration testing due to its simplicity and extensive libraries for network and security analysis.
5. **Financial** **Analysis**: In finance, Python is used for tasks like risk assessment, algorithmic trading, and portfolio management. Libraries such as QuantLib are vital for financial analysis.
6. **Internet of Things (IoT**): Python is used to program and control IoT devices. It's well-suited for collecting, processing, and transmitting data from sensors and devices.
7. **Education**: Python is a popular choice for teaching programming due to its readability and simplicity. Many introductory programming courses and coding bootcamps use Python as their primary language.
8. **Databases**: Python is used for database management and interaction. Libraries like SQLAlchemy facilitate the interaction with databases.
9. **Desktop** **Applications**: Python can be used to create desktop applications with libraries like Tkinter, PyQt, and Kivy. It offers a way to develop cross-platform applications.
10. **Rapid Prototyping**: Python's simplicity and ease of use make it an ideal language for rapid prototyping, allowing developers to quickly build and test ideas.
11. **Automotive Industry**: Python is used in the automotive industry for tasks such as vehicle control systems and diagnostics.

4

**1.3 PYTHON AS PROGRAMMING LANGUAGE AS CHOICE**

Python is a popular and versatile programming language that stands out as an exceptional choice for a wide array of applications. Its selection as a programming language is driven by its simplicity, readability, and an extensive ecosystem of libraries and frameworks. Python's ease of learning and usage make it an attractive option for beginners and experienced developers alike. Its elegant and clean syntax reduces the learning curve, enabling programmers to write code that is easy to understand and maintain.

Python's adaptability is evident in its broad range of applications. It is widely used in web development, data science, machine learning, and artificial intelligence, making it a key player in modern technologies. Additionally, Python's cross-platform compatibility ensures that code written in Python can be executed on various operating systems without significant modifications. This enhances the language's portability, making it suitable for a diverse set of projects.

Furthermore, Python offers strong community support, fostering the growth of a vast repository of resources, tutorials, and code examples. Its vast community contributes to the continuous development of libraries and frameworks, ensuring that Python remains at the forefront of technological innovation. Python's proficiency in automation, data analysis, and scripting, coupled with its powerful libraries, cements its place as a versatile and valuable programming language for a multitude of tasks. Its simplicity, coupled with its adaptability and widespread use, makes Python an excellent choice for a diverse range of programming projects.

5

**1.4 WHY CREATE A PYTHON SNAKE GAME**

Creating a Python Snake Game holds educational and recreational value, making it an engaging project for both novice and experienced programmers. Python's suitability for game development, combined with the simplicity of the Snake Game concept, makes it an excellent choice.

From an educational standpoint, building a Snake Game in Python serves as a practical way to learn the language and familiarize oneself with fundamental programming concepts. It provides hands-on experience in handling user input, game logic, and graphical rendering, helping developers sharpen their problem-solving skills. The game's simplicity and immediate feedback make it a rewarding project for those seeking to apply their Python skills in a fun and interactive manner.

Additionally, recreating a classic game like Snake in Python allows developers to gain insights into game development and understand the essential components of gaming, such as controlling game objects, managing scores, and handling user interactions. This knowledge can serve as a stepping stone for those interested in exploring more complex game projects.

From a recreational perspective, Python Snake Game is a delightful and nostalgic pursuit. It offers a chance to relive the enjoyment of a classic game while customizing it with unique features or graphical enhancements. Whether for personal enjoyment or to share with friends and fellow enthusiasts, a Python Snake Game project can be a source of amusement and creativity.

In summary, creating a Python Snake Game combines the benefits of learning and enjoyment. It provides an opportunity to reinforce programming skills, delve into the world of game development, and relish the simplicity and fun of a timeless classic—all while harnessing the power and flexibility of Python as the programming language of choice.

6

## CHAPTER-2

**TOOLS AND**

**TECHNIQUES**

7

**2.1 PYTHON AND PYGAME LIBRARY**

Python, when combined with the Pygame library, offers an exceptional environment for game development. Pygame simplifies the creation of interactive games by providing extensive features for graphics and sound handling, as well as efficient event-driven user input processing. Its opensource nature and strong community support make it accessible to developers of all levels of expertise, making it a prime choice for crafting games like Snake.

The Pygame library excels in its capability to manage game assets, facilitating collision detection and enhancing the development of 2D and simple 3D games. With Pygame, developers have access to comprehensive documentation, tutorials, and a collaborative community, ensuring that Python remains a versatile, powerful, and approachable platform for turning game ideas into fully realized interactive experiences.

8

**2.2 HARDWARE REQUIREMENTS**

When embarking on a Python Snake Game project using the Pygame library, the hardware requirements are quite modest. As Pygame is a lightweight and efficient library, it can comfortably run on standard hardware configurations. In most cases, a modern personal computer or laptop with a reasonably capable CPU, a few gigabytes of RAM, and a standard graphics card will suffice for smooth game development and gameplay. The specific hardware requirements may vary slightly depending on the complexity of your game and the desired performance, but for a basic Snake Game project, no high-end or specialized hardware is necessary.

Additionally, Pygame's versatility extends to its compatibility with various operating systems, including Windows, macOS, and Linux. Whether you're using a budget-friendly laptop or a powerful desktop, Pygame ensures a consistent development experience across different platforms. This cross-platform support adds to the accessibility of Python Snake Game development, allowing developers to seamlessly work on their projects regardless of the operating system they prefer. Moreover, Pygame's efficient resource utilization means that you can focus more on refining your game mechanics and enhancing the player experience, rather than worrying about demanding hardware prerequisites. Overall, the modest hardware requirements and cross-platform compatibility make Pygame an excellent choice for developers looking to create engaging and accessible games without the need for high-end hardware configurations.

9

**2.3 SOFTWARE REQUIREMENTS**

To develop a Python Snake Game with the Pygame library, you'll need the following software: Python (version 3 recommended), Pygame library, an Integrated Development Environment (IDE) or a text editor, and optionally, graphics and sound editing software for asset creation and version control software for collaborative development. These software components offer a flexible and efficient environment for game development.

Python, the backbone of your Python Snake Game, is an open-source and widely-used programming language. A version 3 installation is recommended for compatibility with the Pygame library and to leverage the latest language features.

The Pygame library acts as the powerhouse for game development. It provides modules for handling various aspects of game development, such as graphics, input, and sound. Pygame's ease of use makes it an ideal choice for both beginners and experienced developers, fostering a smooth development process.Choosing the right Integrated Development Environment (IDE) or a text editor is crucial for an efficient workflow. Popular choices include PyCharm, Visual Studio Code, and IDLE. These tools offer features like code highlighting, autocompletion, and debugging capabilities, enhancing your coding experience.For those aiming to elevate the visual and auditory aspects of their Snake Game, graphics and sound editing software become valuable assets. Tools like Photoshop or GIMP for graphics and Audacity for sound allow you to create custom assets, adding a personalized touch to your game.Additionally, version control software such as Git can be immensely beneficial, especially in collaborative projects. It enables multiple developers to work on the game simultaneously, keeping track of changes and ensuring a seamless integration of different contributions.By combining these software components, you create a robust development environment that empowers you to bring your Python Snake Game to life with creativity, efficiency, and collaborative potential.

10

## CHAPTER-3

**PROJECT WORK**

11

**3.1 PROJECT OVERVIEW**

**Project Overview: Python Snake Game**

**Snake game**:

Although the concept of snake-based games originated in 1976, there was a resurgence of interest only after a variant was preloaded on Nokia mobiles in 1998. We have optimized the exact same variant using Python programming language. When user hits "start" a snake is generated at predefined position on the canvas and a piece of food is generated at a random position. We can control snake movements using "UP"." DOWN","RIGHT" and" LEFT" keys of our keyboard. When head of snake collides with the piece of food, the piece of food disappears, snake size increases by one block and a new food is generated at some random position on canvas other than the areas covered by snake parts.

**Description of Game**:

The current snake game is a single player game. The game interface is built in 2D format. The game is displayed in the pygame window. Here, it consists of two points where one is movable while the other is fixed. The movable point is the snake while that of the point at fixed location is the food. Whenever the snake gets the food the length of snake increases by one. The speed of the snake can be altered from the code. The player controls a long, thin creature, resembling a snake, which roams around on a bordered plane, picking up food (or some other item), trying to avoid hitting its own tail or the edges of the playing area. Each time the snake gets a piece of food, its tail grows longer by one. Also, the player score increases by one and making the game increasingly difficult. The user controls the direction of the snake's head (up, down, left, or right), and the snake's body follows.

A console window is created as soon as the game starts. The game will terminate if the snake head collides with any of the window's boundaries or with its own tail. Whenever the game is finished, the score is displayed in the centre of the window. The GUI is developed using pygame and various image files are used to create the interface. The interface is divided into various sections and each section lead up to different functionalities. The limitations with the existing system is that the game has no dedicated storage allocated to store the user's information and performance statistics.

12

**Snake game logic**:

In this snake game, if the player hits the boundaries of the screen, then he loses. To specify that. I have made use of an 'if statement that defines the limits for the x and y coordinates of the snake to be less than or equal to that of the screen. Also, make a not over here that I have removed the hardcodes and used variables instead so that it becomes easy in case you want to make any changes to the game later on.

In pygame, we have this thing called Rect.. Rects make collision detection easy. You can also use Rect on images, any rectangular object works. It can also be circular. Now to make a Rect, we have to use Pygame.Rect() method We have to pass in the x coordinate as the first argument, y coordinate as the 2nd argument, and width and height as the 3rd and 4th argument respectively In our case, we first pass the snakes x coordinate, y coordinate, it's width and height Now rects have a lot of methods available, one is colliderect(), This method takes another Pygame.Rect() as argument and checks if they two collide with each other In our case, we pass fruit pos transforming as Rect to the colliderect method. If this returned true, fruit\_spawn will be true, so new fruits will be generated and the score will be increased. The game will terminate if the snake head collides with any of the window's boundaries or with its own tail.

13

**3.2 ALGORITHM**

Creating a Python Snake Game is a substantial project. Here's a simplified overview of the key steps:

1. Set up your environment by installing Python and the Pygame library.
2. Initialize Pygame and create the game window.
3. Create the snake and food elements with defined attributes. 4. Implement the game logic, handling collisions and scoring.
4. Capture user input for controlling the snake's direction.
5. Design graphics, display the game elements, and add sound effects (optional).
6. Include start and game over screens, creating a user-friendly interface (optional).
7. Thoroughly test and debug the game.
8. Document your code for future reference.

This framework will guide you through the process of building and customizing your Python Snake

Game.

14

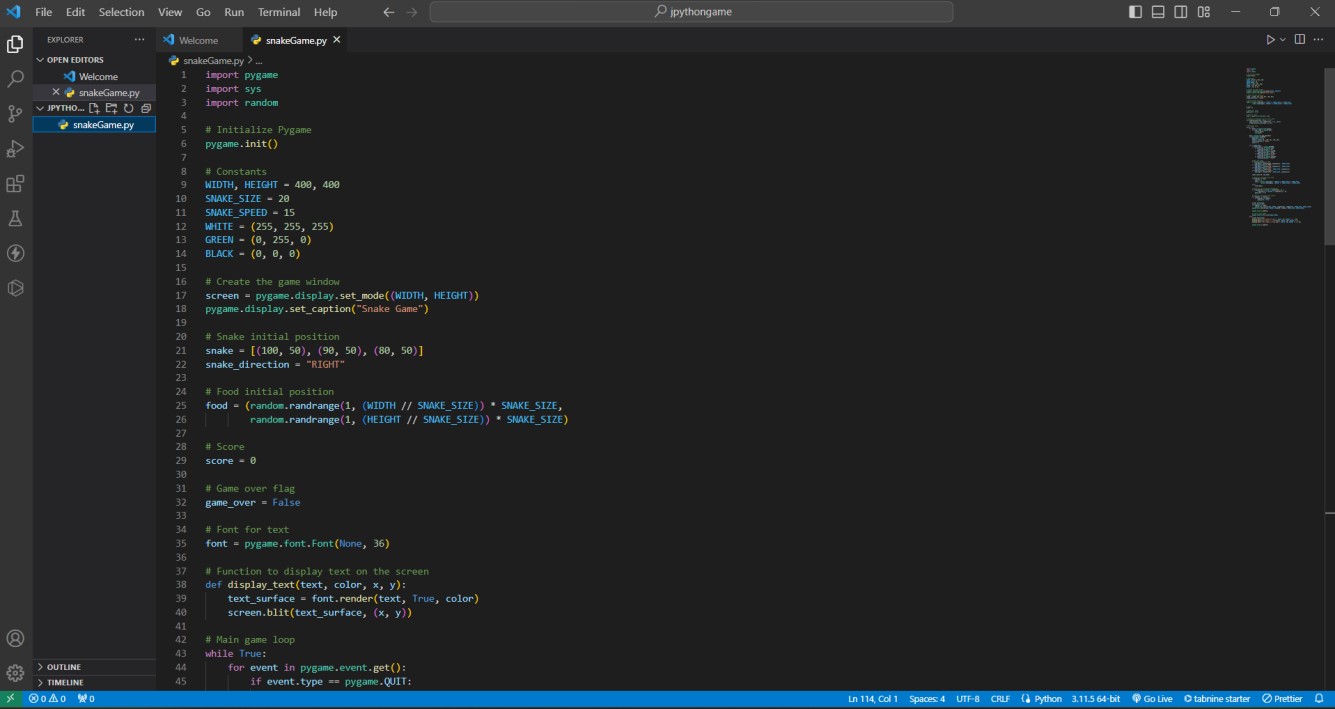
## CHAPTER-4

**CODE AND OUTPUT**

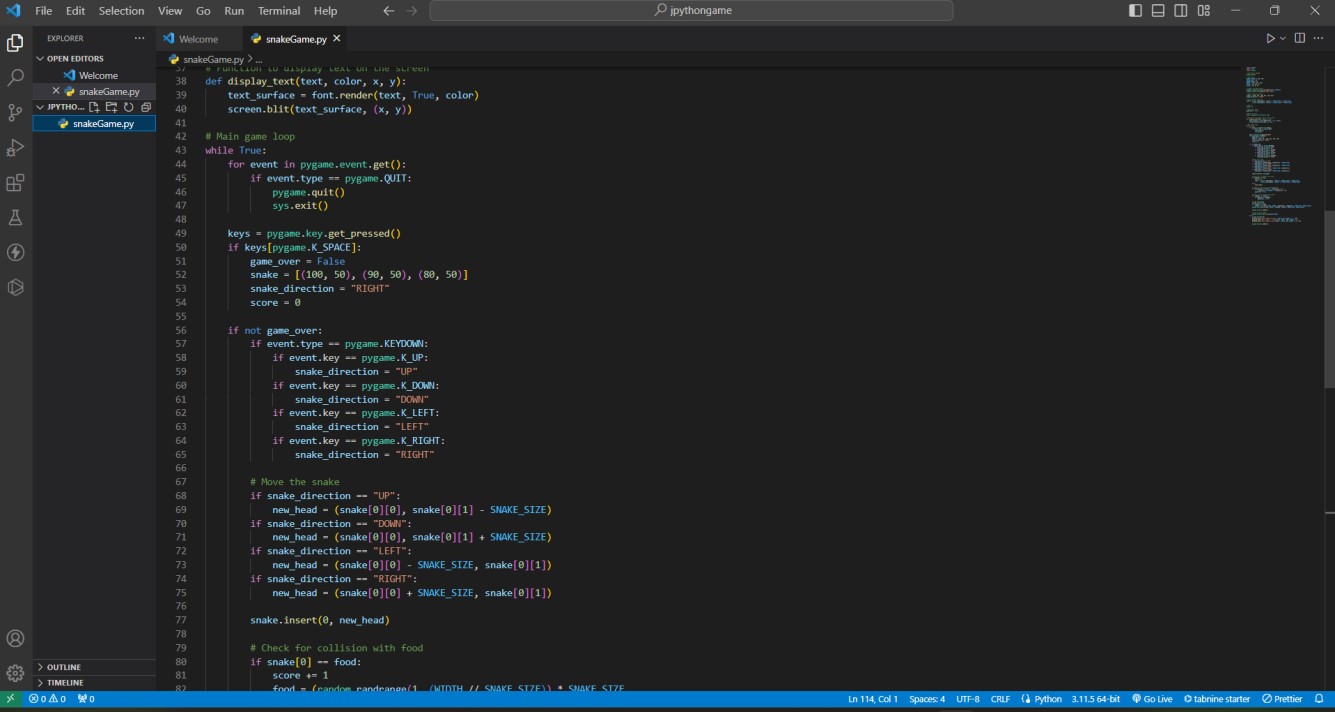
**SCREEN SHOTS**

15

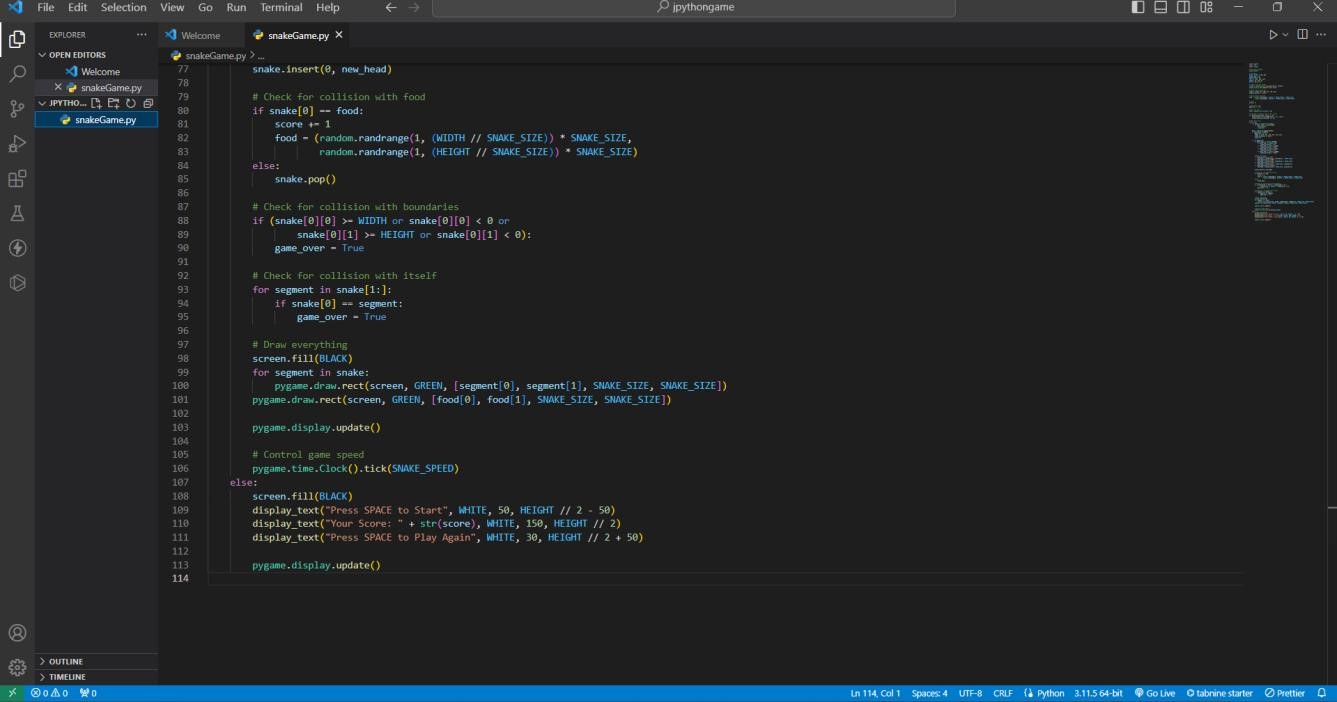
**Source Code:**



16

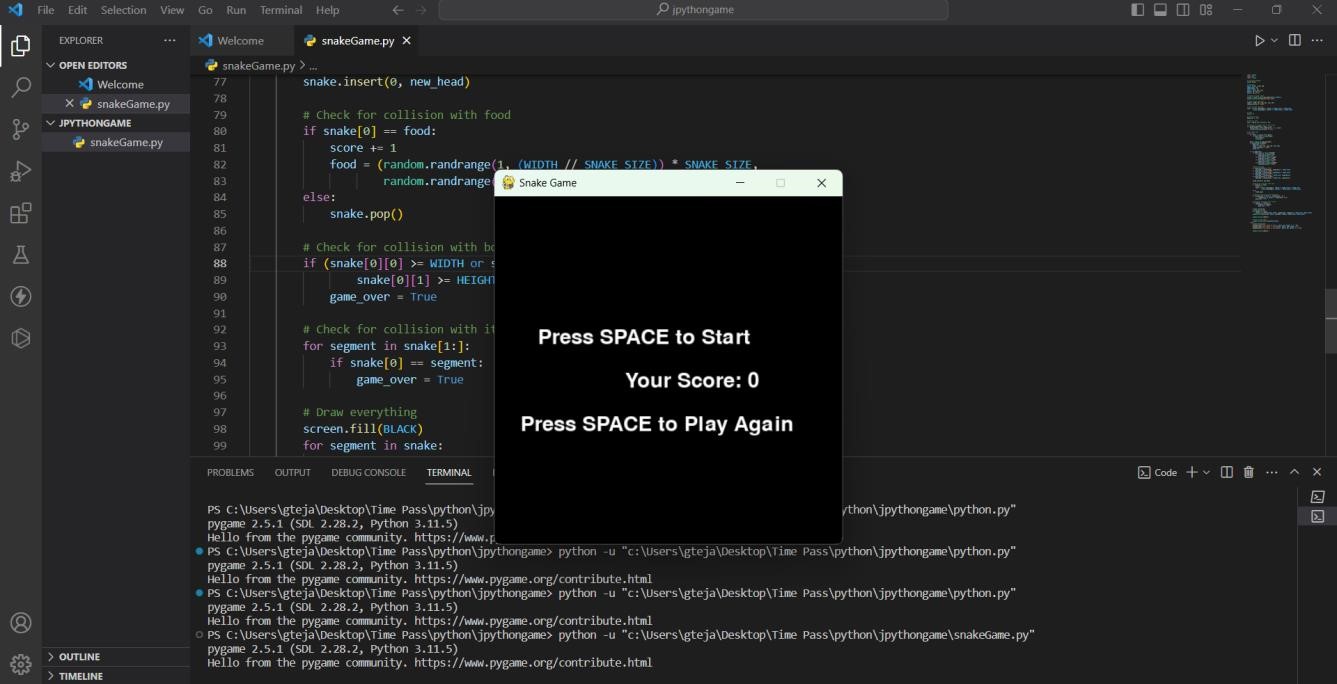


17

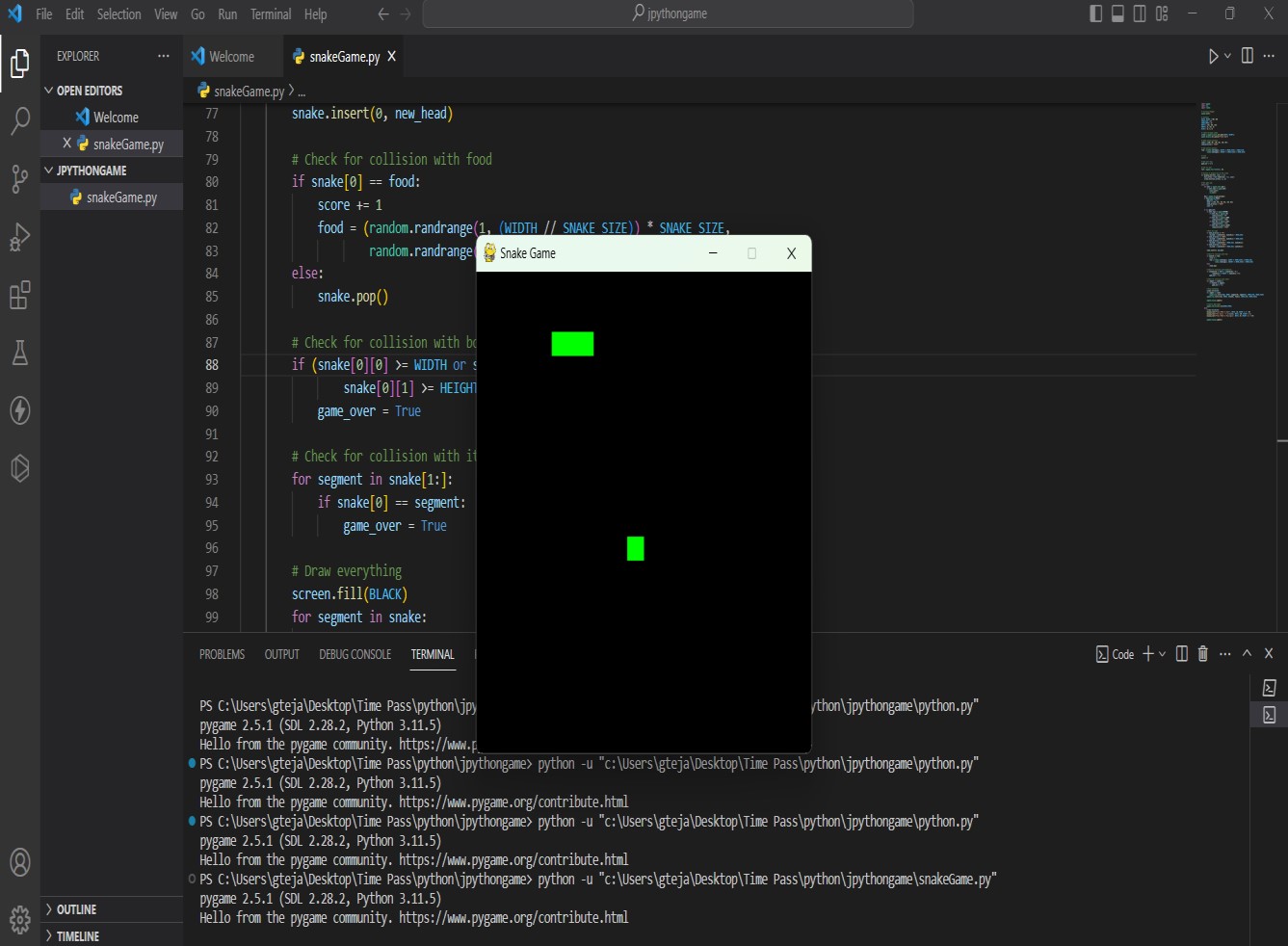


18

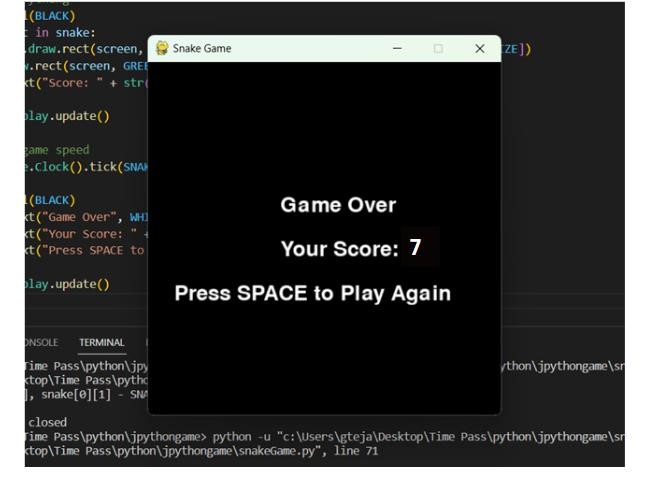
**Output:**



19



20



21

**CHAPTER**-**5**

**CONCLUSION**

22

**Conclusion :**

In conclusion, the Python Snake Game project showcases the potential of the Pygame library for game development and serves as an engaging and educational endeavor. This project offers a classic gaming experience while providing valuable learning opportunities. The key features of the game, including snake movement, food generation, scoring, and a game over condition, make it a compelling example of game logic implementation.Moreover, the code modifications introduced a black background, a start screen, and the ability to display the player's score and initiate a new game, enhancing the user experience. This project combines programming skills, problem-solving, and creativity, making it an ideal choice for both beginners and experienced developers to explore the world of game development.Ultimately, the Python Snake Game exemplifies the versatility and fun of programming in Python, demonstrating how a simple concept can be transformed into an interactive and enjoyable gaming experience. Whether used for learning or recreation, this project serves as an excellent platform for honing coding skills and indulging in a timeless gaming classic.

23

## REFERENCES

24

**References :**

1. PygameDocumentation: The official pygame

Documentation (https://www.pygame.org/wiki/GettingStarted) is a valuable resource for understanding Pygame's functions, modules, and how to use them in game development. PythonDocumentation: The Python official documentation (https://docs.python.org/3/) is essential for understanding Python's syntax, libraries, and functions.

1. Online Tutorials: Various online tutorials and blogs provide step-by-step guides for creating a

Python Snake Game using Pygame. Websites like Real Python (https://realpython.com/) and Pygame Tutorials (https://www.pygame.org/wiki/tutorials) often have relevant content.

1. YouTube Tutorials: Video tutorials on platforms like YouTube can be helpful for visual learners. Channels such as "The Coding Train" and "Tech With Tim" offer Python game development tutorials.

1. Books: Books like "Invent Your Own Computer Games with Python" by Al Sweigart and

"Pygame for Dummies" by Jessica Thornsby and Luke Biddell can provide in-depth knowledge.

1. Community Forums: Online forums like Stack Overflow (https://stackoverflow.com/) and the Pygame Community (https://www.pygame.org/wiki/Forum) can be helpful when you encounter specific issues or have questions.

25