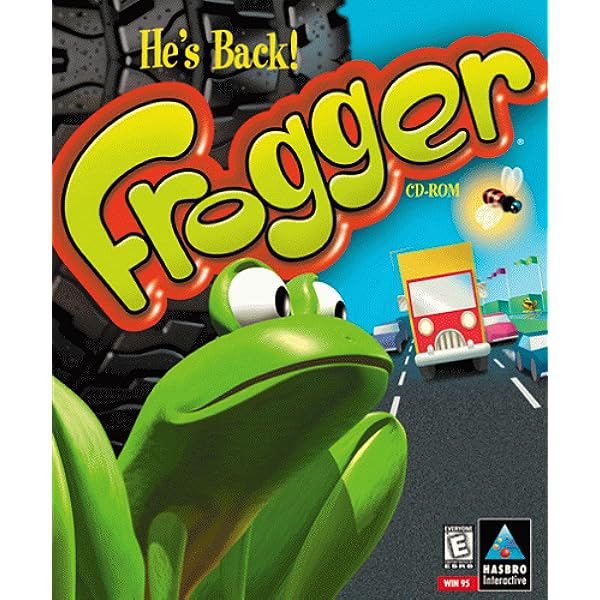
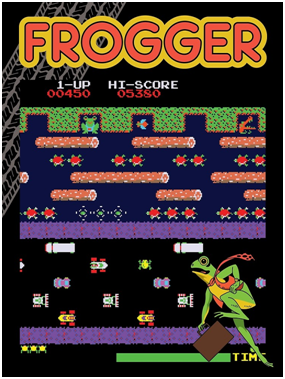
**Frog Game**

**Report:**

***1. Introduction / Background of the Project:***

The Frog Game, inspired by the vintage arcade game "Frogger," is designed to rejuvenate the classic experience while introducing modern enhancements.

The primary objective is to guide a frog safely across a series of obstacles like roads and rivers. This Java-based project stands as a testament to the capabilities of the Java programming language in game development.

***2. Background Research / State-of-the-Art:***

* Modern adaptations use 3D graphics and even VR capabilities.
* Mobile versions have introduced swiping mechanics.
* Various game engines, ranging from Unity to Godot, have been used for newer versions
* Historically, the original Frogger was developed in 1981 by Konami.
* Since then, numerous variations and adaptations have been created across multiple platforms. Our research indicated that Classic implementations focused on 2D graphics, often with 8-bit or 16-bit styles

***3. Project Specifications:***

**Platform:** Desktop

**Graphics**: 2D sprite-based animation

**Levels**: Five increasing difficulty levels with varied obstacles.

**Controls:** Arrow keys for navigation and a space bar for special jumps.

**Scoring System:** Players earn points based on speed, collecting bonuses, and successful crossings.

***4. Implementation Details and Results / Outputs:***

**Language**: Java, due to its platform independence and extensive libraries like Swing and AWT for GUI components.

**IDE**: Eclipse, favored for its robust Java development environment and debugging tools.

Data Structures:

**Array-Lists:** Used for managing dynamic obstacles like cars and logs. Allows for easy addition and removal of obstacles.

**2D Arrays:** For grid-based mapping of the game area, helping in collision detection.

**Stack**: For implementing an undo feature, where players can trace back their steps.

**Queue:** To manage the sequence of upcoming obstacles and ensure no overlaps.

**Graphics Libraries:** Java's AWT and Swing for rendering game components and managing user inputs.

***5.Conclusion/Future Work:***

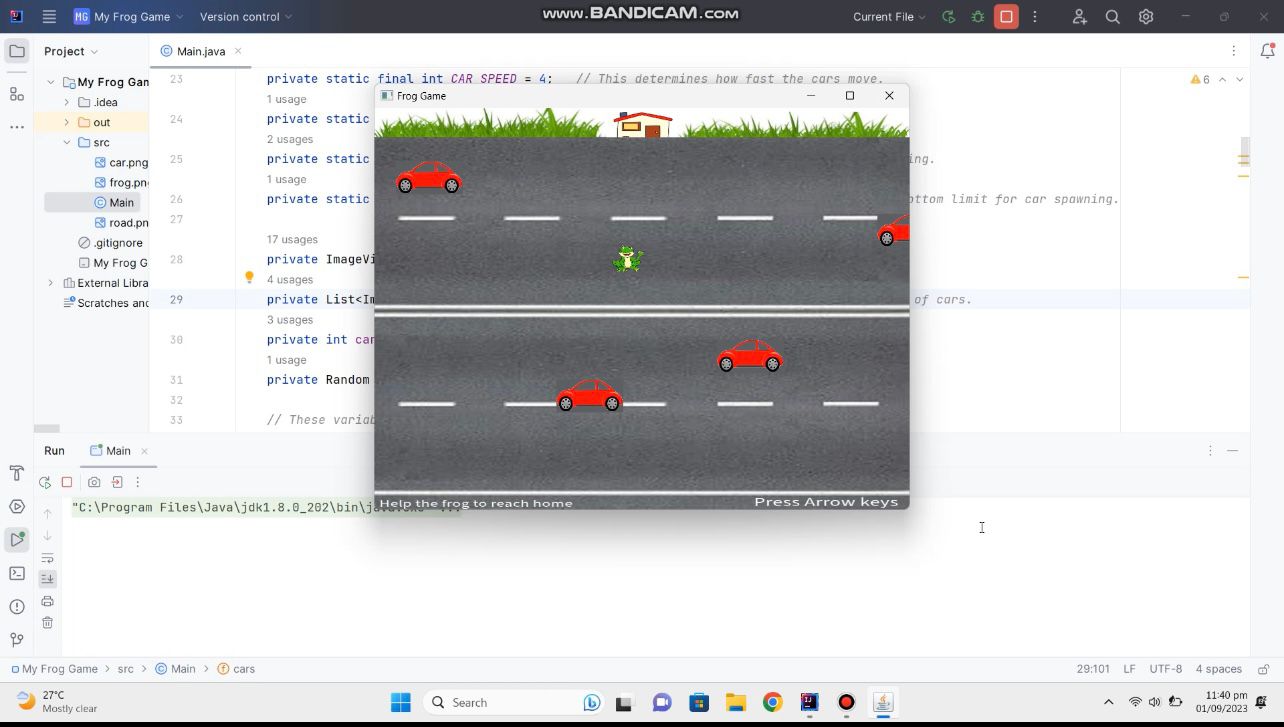
The Frog Game shows off Java's flexibility in game programming in addition to being a nostalgic journey. Java is still a solid option for building trustworthy and platform-independent apps despite its advanced age.

For potential improvements:

1. Introduce the concept of multiplayer
2. Include extra obstacles and power-ups.
3. Investigate a mobile version with touchscreen-friendly controls
4. Sync with internet leaderboards to promote player competition.

There are many chances to adapt and develop the Frog Game for contemporary audiences by building on the framework established in this project.

**Game Output:**

****

