App Inventor Advanced

Session 1: Tic Tac Toe Game

Objective: Introduce students to more complex app development by creating a classic Tic Tac Toe game.

Content Overview:

1- Game Design Basics:

- Setting up the game board using a grid layout.
- Understanding how to create a simple turn-based game.

2- Event Handling and User Interaction:

- Introduction to event-driven programming with button clicks.
- Capturing player moves and alternating turns between players.

3- Managing Game State:

- Using variables to keep track of game progress (e.g., current player, board state).
- Displaying win, draw, or next player's turn messages based on game conditions.

4- Building Win Conditions:

- Implementing logic to detect when a player wins (three in a row).
- Highlighting the winning line and resetting the game for a new round.

Session 2: Ping Pong Game

Objective: Explore game physics and control flow by creating a Ping Pong game.

Content Overview:

1- Understanding Game Physics:

- Basics of using timers to control ball movement.
- Implementing simple physics for ball bouncing off paddles and walls.

2- User Input for Paddle Control:

- Introduction to touch and drag gestures for paddle movement.
- Using buttons or sliders for alternative control methods.

3- Game Score Tracking:

- Setting up a scoring system to keep track of points.
- Displaying player scores and updating them dynamically during the game.

4- Game Over Logic:

- Implementing conditions to end the game when a player wins.
- Displaying the winner and offering a restart option.

Sessions 3 & 4: Ant Smasher Game

Objective: Develop an engaging arcade-style game over two sessions, focusing on timing, scoring, and animations.

Session 3:

Content Overview:

1- Game Design and Animation:

- Setting up the game environment with a background and animated ant sprites.
- Introduction to animations and making ants move across the screen.

2- Timers and Object Movement:

- Using timers to manage ant speed and movement patterns.
- Setting up random movement to make the game more challenging.

3- Interaction and Score Keeping:

- Implementing touch events to "smash" ants.
- Setting up variables to keep track of the score.

4- Basic Sound Effects:

- Adding sound effects for successful ant smashes.
- Implementing feedback sounds to enhance gameplay.

Session 4:

Content Overview:

1- Advanced Movement and Difficulty Levels:

- Adjusting ant speed based on score or time to increase difficulty.
- Introducing levels or stages to make the game progressively harder.

2- Game Over and Restart Logic:

- Handling game over conditions when the player misses too many ants.
- Displaying final scores and offering restart options.

3- Game Polishing:

- Fine-tuning game animations for smooth transitions.
- Adjusting touch sensitivity to ensure responsive gameplay.

Sessions 5 & 6: Connect Four Game

Objective: Build a two-player Connect Four game over two sessions, focusing on grid management and strategic play.

Session 5:

Content Overview:

1- Game Setup and Grid Design:

- Designing the Connect Four game board using a grid layout.
- Setting up player inputs and creating a UI for placing tokens.

2- Turn-Based Logic:

- Using variables to manage alternating turns between players.
- Visual feedback for whose turn it is and displaying player tokens on the board.

3- Data Structures for Game State:

- Introduction to using lists or tables to track board state.
- Developing the logic for placing tokens and checking for available spots.

Session 6:

Content Overview:

1- Implementing Win Conditions:

- Coding the logic to check for four consecutive tokens horizontally, vertically, or diagonally.
- Highlighting the winning line and stopping the game when a win is detected.

2- Game Over Scenarios and Restart:

- Handling game over scenarios (win or draw) and displaying results.
- Offering an option to restart the game for another round.

3- Advanced Features and Enhancements:

- Implementing a basic AI for single-player mode (optional).
- Adding sound effects and animations to enhance user experience.