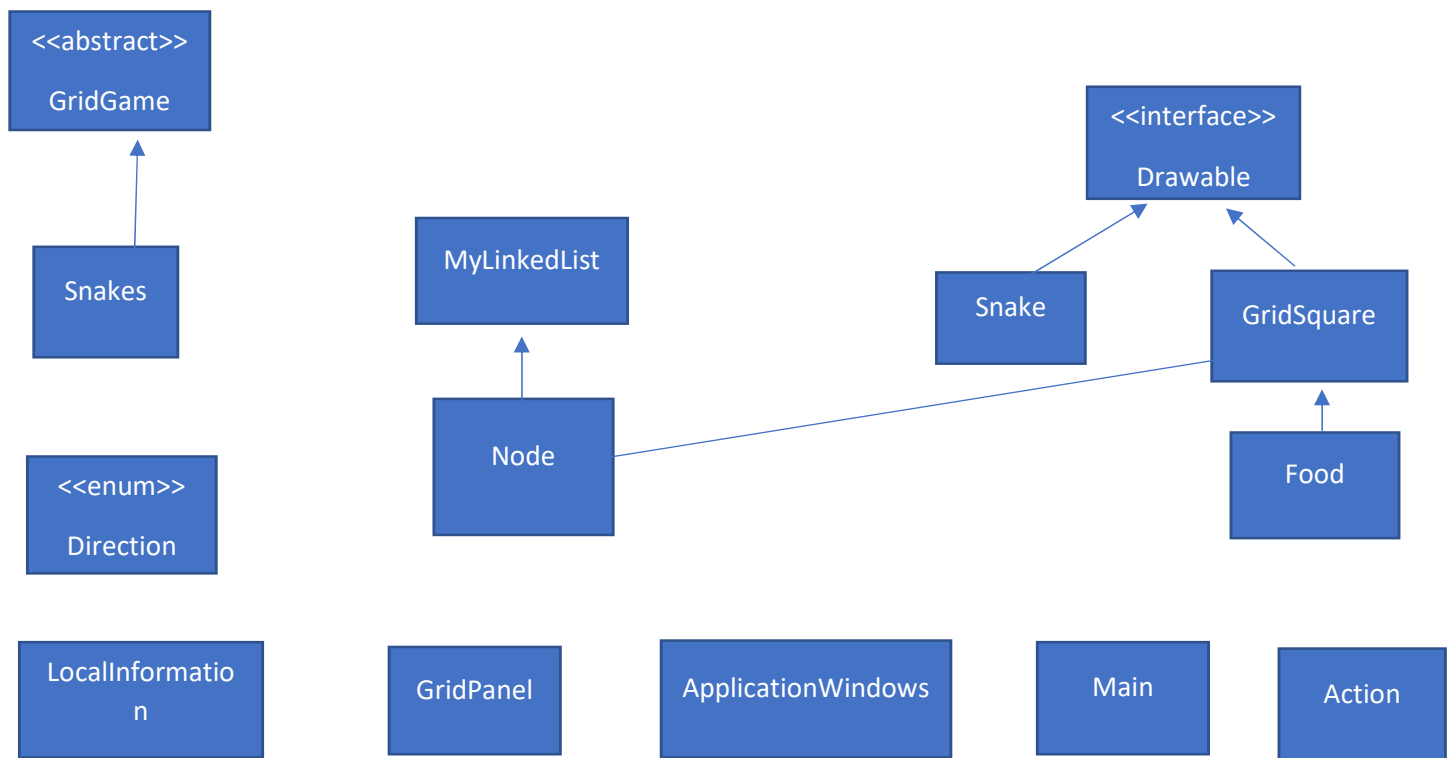


# CmpE 160 - Introduction to Object Oriented Programming

## Project #2 - Snakes!



Class Hierarchy

## Class Definitions

1. `Direction`: An enum for directions
2. `Drawable`: An interface that allows Snakes and the food to be drawn on the `GridPanel`
3. `GridGame`: A class representing a generic grid-based game.
4. `Main`: A class to execute the game. You can change the world width and height, size of each grid square in pixels or the game speed.
5. `Action`: A class representing the possible actions for Snakes game
6. `LocalInformation`: A class representing the information a snake head has about its surroundings. Automatically created and passed by the game to each creature at each timer tick.
7. `Snakes`: A child of `GridGame` that determines the action for every snake when `TimerTick` method been executed. While executing the method, whenever the size of a snake becomes 8, it calls the `reproduce` method of the snake.
8. `MyLinkedList`: A `LinkedList` implementation that consists of `Nodes` for every Snake in the game
9. `Node`: Nodes of the `MyLinkedList`. Stores a `GridSquare`
10. `GridSquare`: A class representing coordinates for any square

11. Food: A child of GridSquare
12. Snake: It consist of a list of GridSquares. It choses action in the following way: First it checks is there any food nearby. If there is, then eats it. Secondly, if it is possible to move toward to the food it choses one of them randomly. For example, if the head of the snake is at (3,4) and the food is at (7,7), it choses one of "Left", "Down" which is possible to move. Moreover, if they are not available then choses one of the free directions thanks to LocalInformation.
13. ApplicationWindow: The main application window.
14. GridPanel: A drawable panel structured as a grid. Provides some drawing methods suitable for pixel-like game entities.