**Workshop for Design Pattern in Go**

1. Write a Go program to layout game assets on a MxN two-dimensional game board.
2. There are only two types of game asset in the game - Humans and Monsters.
3. Each game asset requires X number of bytes for storing its image. A Human requires 128 bytes of data, while a Monster requires 512 bytes. We do not need to render the game assets in this workshop, but design your program such that your objects (i.e. game assets) can access their images readily.
4. Each game asset has two properties (instance variables) - health and power. On its creation, each game asset has a value of 100 for health. However, for power, a Human only has a value of 10, while a Monster 50.
5. Create 5 Humans and 5 Monsters and assign them randomly to an empty slot on the board. A position on the board cannot contain more than one game asset.
6. Then print out the game board, with H denoting a Human and M a Monster (like the image below that depicts a 10x10 board).

Background pattern

Description automatically generated

1. Design your program to use the Flyweight design pattern.