Step 1:

* Candidates for refactoring:
  + Modify bush class: It is currently an empty class.
    - Can either put items in class
    - Or eliminate class completely
  + DecideMove:
    - Is used for Wolf and Bunny.
      * Can move to Animal and make bunny smarter
      * Can make Bunny smarter in Bunny
    - Decidedly different for both and wolf is considerably smarter
  + There is both a constant and a private variable for the number of rows and columns.
    - There is also a numRow inside the poplulate function that does not get modified.
    - The constant is simple set to the private one in the constructor
    - Also used in model
    - Determine which is used more and use that one
    - This could cause problems if the field size changes
  + The bunny and the wolf are not that different., it may not be necessary to have separate classes for them
  + Some methods are longer than 25 lines. May be necessary to compartmentalize more.
  + Possibly create a moves class since it is so long.

Step 2:

To make the bunny smarter, I removed the decideMove override back to animal. Moving it to animal made it also unnecessary in wolf. There is minimal refactoring. The moves are so similar that I thought it best be managed the common class that they extend. The only difference is what they do if they are the bunny or wolf which I added to the code. The code of decide move is essentially the old wolf DecideMove code with modifications to determine what to do if it is the wolf or if it is the bunny.