

```

1  /*
2  * Programmer:  Chad Philip Johnson
3  * Date Created:  Thursday, November 15th, 2012
4  * Date of Last Modification:  Thursday, November 15th, 2012
5  *
6  * Description:
7  * MyFarm.class provides a framework for an instance of Farm.class to be
8  * started, which contains the main subroutines of the program.  MyFarm.class
9  * depends on MyFarm.class for necessary program subroutines and CinReader.class
10 * to handle user input.
11 */
12
13 import java.util.*;
14
15 /**
16 * MyFarm.class provides a framework for an instance of Farm.class to be
17 * started, which contains the main subroutines of the program.  MyFarm.class
18 * depends on MyFarm.class for necessary program subroutines and CinReader.class
19 * to handle user input.
20 *
21 * @author Chad Philip Johnson
22 * @version 1.0
23 */
24
25 public class MyFarm {
26
27     static final boolean START_WITH_SMALL_BARN = false;
28
29     /**
30      * Run the program
31      */
32
33     public static void main( String[] args )    {
34
35         CinReader driverKeyboard    = new CinReader();
36
37         Farm someFarm                = null;
38
39         char chrUserInput;
40
41         System.out.print    ( "\n\n\n\nWelcome to FarmSim!\n" );
42
43         while( true )    {
44
45             System.out.print    ( "\nWould you like to start a (N)ew game, (L)oad a previous game, or (Q)uit?  " );
46             // Convert character input first to string, then to lowercase and back to character.
47             // Ensures that only lower case characters are evaluated, regardless of whether the user typed upper or lower case.
48             chrUserInput    = Character.toString( driverKeyboard.readChar() ).toLowerCase().charAt( 0 );
49
50             switch( chrUserInput )    {

```

```

51
52 // New game
53 case 'n':
54     String strFarmName;
55     while( true ) {
56
57         System.out.print ( "\nWhat would you like the name of your new farm to be?\n" );
58         System.out.print ( "(your farm name must be between 4 and 16 characters/numbers in length)\n" );
59         strFarmName = driverKeyboard.readString();
60
61         // Restrict name length from 4 to 16 characters
62         if( ( strFarmName.length() < 4 ) || ( strFarmName.length() > 16 ) ) {
63
64             System.out.print ( "\nYour farm's name must be between 4 and 16 characters/numbers in length.\n" );
65             System.out.print( "Press enter to continue..." );
66             driverKeyboard.readString();
67             continue;
68
69         }
70
71         System.out.print ( "\nYou have chosen the name of '" + strFarmName + "' for your new farm.\n" );
72         System.out.print ( "Would you like to keep this name? (y/n) " );
73         chrUserInput = Character.toString( driverKeyboard.readChar() ).toLowerCase().charAt( 0 );
74
75         if( chrUserInput == 'y' ) {
76
77             // Option to configure the program to start the player with a small barn automatically
78             if( START_WITH_SMALL_BARN ) {
79
80                 // Create an instance of Farm.class with a barn that can contain two animals
81                 someFarm = new Farm( strFarmName, 2 );
82
83             } else {
84
85                 // Create an instance of Farm.class without a barn (player must purchase one)
86                 someFarm = new Farm( strFarmName );
87
88             }
89
90             System.out.print ( "\nWelcome to '" + strFarmName + "'. Time to get to work!\n" );
91
92             System.out.print( "Press enter to continue..." );
93             driverKeyboard.readString();
94             break;
95
96         }
97
98     }
99     break;
100

```

```
101 // Load game (currently not functioning)
102 case 'l':
103     System.out.print ( "\nYou have chosen to load a previous game.\n" );
104     System.out.print ( "What is the name of the file you would like to load?\n" );
105     someFarm = FarmStorage.loadFarm( driverKeyboard.readString() );
106     System.out.print( "Press enter to continue..." );
107     driverKeyboard.readString();
108     break;
109
110 // Quit game
111 case 'q':
112     break;
113
114 // Incorrect entries cause the loop to begin again
115 default:
116     System.out.print ( "You entered '" + chrUserInput + "' which is not a valid option.\n\n" );
117     System.out.print( "Press enter to continue..." );
118     driverKeyboard.readString();
119     continue;
120
121 }
122
123 break;
124
125 }
126
127 // Start game when user does not quit game
128 if( chrUserInput != 'q' ) { someFarm.go(); }
129
130 return;
131
132 }
133
134 }
```