Upon opening the program, display a greeting message along the lines of “It’s a new day!” \*newline\* “Let’s do some setup!” After the greeting message, the user should input the date (e.g., Jan 1) and the quantity of each animal they have. Once completed, the user is greeted with a main menu.

* Track items sold through the store
  + An “Add Purchase” Option
    - Once selected, the user inputs the time of purchase.
      * This will be appended to the first index of the ArrayList
      * Time should be given in 24-hour metric (I.e., 1:00 PM = 13:00)
    - The user then inputs what item/animal is being purchased.
      * An if statement will check whether this input is in the list of available items\*. If it is, then it is added to the item list for this purchase. If it isn’t, display an “Invalid Input” message and send the user back to the input.
      * \*Requires another list and a way to check between it and the user’s input.
    - Then the user inputs how much is being purchased.
      * If the item of purchase is an animal, then the quantity purchased must not exceed the present supply.
      * If the input is valid, then this value is appended to another list, where the indexes coincide with the item list.
        + The first index will need a placeholder, as the first index of the items list is the time.
        + (e.g., if the first thing purchased was “2 cows”, then the second index of the item list would be “cow” and the second index of the quantity list would be “2”)
    - Finally, ask the user if the purchase is complete.
      * If it is not, send the user back through the loop for another item. If it is, break the loop and save the list.
  + A “Review Purchases” Option
    - Taking in all the purchases the user has inputted, generate a display where the time of purchase, contents of purchase (both item and quantity), and price of purchase are each displayed in their own columns.
      * If no purchases have been catalogued, display a message, then kick the user back to the main menu.
* Schedule and track services offered (and payments received)
  + A “Schedule Service” Option
    - The user selects from a list of different services, then enters the date that the service will occur. If the service is already scheduled, then the program will ask the user if they would like to overwrite the current schedule.
  + Unsure of how to track payments received.
* Manage animal sales (e.g., ducks, chickens, hamsters, rabbits, and other farm-related animals)
  + Managed under “Add purchase”
    - Perhaps add a second option specifically for animal purchases?
* Support occasional specialty animal resales from local breeders
  + Unsure of how this interacts with the larger management of animal sales.