```
/**
* @(#)BhaktaJ003PA1.java
* @author Jay Bhakta
* @version 1.00 2023/09/24 10:35 AM
*
* PROGRAM PURPOSE: Create a program for calculating the cost of intended stock purchases.
*/

import java.util.Scanner; //Class to access keyboard entries.
import java.util.Calendar; //Class to access the system's date.

public class BhaktaJ003PA1
{
   /**
   * Investors can choose to proceed with the stock calculator
   * or not.  If not, a thank you message is displayed; otherwise,
   * investors are asked to enter their name.  Data pertaining to
   * the calculation is requested.  The stock cost is calculated
   * and added to the respective totals.  An online fee or commission
   * is calculated and added to their respective totals unless the
   * trade type is invalid.  Investors can assess the costs for multiple
   * stocks.  Once there are no more stock costs, the final output is
   * printed and a thank you message is displayed.
   */
  public static void main(String[] args)
  {
    Scanner input = new Scanner(System.in); //References object to read input from the keyboard
    Calendar dateTime = Calendar.getInstance(); //References object to get system's current date
and time
```

```
    //Intializing string variables
    String date = String.format("%1$TB %1$Td, %1$TY", dateTime); //Formatting the system's
date with object
    String customerName = ""; //Object stores a customer name

    //Initializing int variables
    int shares = 0; //Variable that stores a persons shares
    int noStocks = 0; //Variable that tracks a persons number of stocks

    /* Each variable stores the values for the share price, stock cost, commission earned, and the
     * online fee charged for each stock.
     * Totals are accumulated for multiple stocks, and the associated commissions and online fees.
     */
    double sharePrice = 0.0, stockCost = 0.0, commission = 0.0, totalCost = 0.0, onlineFee = 0.0,
      totalStockCost = 0.0, totalCommissions = 0.0, totalOnlineFees = 0.0;

    //Initializing character variables
    char onlineTrade = ' '; //Queries 'Y' or 'N' for an online trade.
    char brokerAssisted = ' '; //Queries 'Y' or 'N' for a broker assissted trade.
    char another = ' '; //Controls the processing of multiple stocks.


    //Company name and welcome message from PA //instructions
    System.out.printf("%nYEE-TRADE, INC. The Wild West of Electronic Trading%n" +
                      "Welcome to Yee-Trade's stock cost calculator.%n");

    System.out.printf("%nEnter \'Y\' to begin stock cost calculations or \'N\' to exit: ");
    another = input.nextLine().charAt(0); //Priming the another variable
```

```
if(Character.toUpperCase(another) == 'Y')
{
  System.out.printf("%nWhat is your name? ");
  customerName = input.nextLine(); //Reading customerName
} //END if another == 'Y'

while (Character.toUpperCase(another) == 'Y')
{
 ++noStocks; //Incrementing noStocks

 System.out.printf("How many shares do you want to purchase? ");
 shares = input.nextInt(); //Asking user for shares and Reading shares

 System.out.printf("What is the price per share? ");
 sharePrice = input.nextDouble(); //Asking user for sharePrice and reading it in

 input.nextLine(); //Clearing the input buffer

 //Using combined assignment operators for stockCost, totalStockCost, totalCost
 stockCost = shares * sharePrice;
 totalStockCost += stockCost;
 totalCost += stockCost;

 System.out.printf("%nIs this an online trade? Enter \'Y\' or \'N\': ");
 onlineTrade = input.nextLine().charAt(0); //Asking user for Y or N and reading it in
                                           //to onlineTrade variable
 if (Character.toUpperCase(onlineTrade) == 'Y')
```

```
{
 onlineFee = 5.95; //Assigning a onlineFee if onlineTrade == 'Y'

 //Using combined assignment operators for totalOnlineFees and totalCost
 totalOnlineFees += onlineFee;
 totalCost += onlineFee;
} //END if onlineTrade == 'Y'
else
{
  System.out.printf("%nIs this a broker assisted trade? Enter \'Y\' or \'N\': ");
  brokerAssisted = input.nextLine().charAt(0);

  if (Character.toUpperCase(brokerAssisted) == 'Y')
  {
    commission = stockCost * .02; //Calculating commission on stockCost

    //Using combined assignment operators for totalCommissions and totalCost
    totalCommissions += commission;
    totalCost += commission;
  } //END if brokerAssited == 'Y'
  else
  {
    System.out.printf("%nINVALID TRADE TYPE!%n");

    --noStocks;
    totalStockCost -= stockCost;
    totalCost -= stockCost;
  } //END if brokerAssisted != 'Y'
```

```
    } //END if onlineTrade != 'Y'
    System.out.printf("%nEnter \'Y\' to calculate the cost for another stock or \'N\' to exit: ");
    another = input.nextLine().charAt(0);
   } //END while another == 'Y'
   if (noStocks > 0)
   {
    System.out.printf("%n%nYEE-TRADE, INC."
                    + "%nTOTAL COST OF INTENDED STOCK PURCHASES "
                    + "%nFOR %s"+ "%nAS OF %s"
   //3 spaces before the format specifiers through commissions.
                    + "%n%nTotal Stock Cost:   $%,14.2f"
                    + "%nTotal Online Fees:   %14s"
                    + "%nTotal Commissions:   %14s"
   //9 spaces before the format specifier for TOTAL COST.
                    + "%n%nTOTAL COST:         $%,14.2f%n", customerName,
                   date, totalStockCost, String.format("%,.2f",
                   totalOnlineFees), String.format("%,.2f", totalCommissions),
                   totalCost);


   } //END if noStocks > 0
   System.out.printf("Thank you for using Yee-Trade's stock cost calculator!%n"); //Printing
thank you
   noStocks = 0; //Zeroing out noStocks
   System.exit(0); //Stopping
  } //END main()
} //END application class BhaktaJ003PA1
```