

ScreenIO.txt

```
1 *****
2 * PROGRAMMED BY : Saul Moreno
3 * STUDENT ID    : 269491
4 * CLASS         : CS 1C MW-5:00pm
5 * ASSIGNMENT #3 : Pointer
6 *****
7
8 This program will show the items available for purchase
9 and how many are in stock. Then after the user has
10 selected how many they want the program will add up
11 their total before and after tax
12
13 Name of Equipment    Cost    Quantity
14 Nike basketball shoes 179.99  25
15 Under Armour T-shirt 29.99   88
16 Brooks running shoes 121.44  13
17 Asics running shoes  165.88  12
18 Under Armour shorts  45.77   35
19
20 Enter how many Nike basketball shoes do you want? 4
21 Enter how many Under Armour T-shirts do you want? 5
22 Enter how many Brooks shoes do you want?          0
23 Enter how many Asics running shoes do you want?    1
24 Enter how many Under Armour shorts do you want?    6
25
26 Nike shoes left in inventory: 21
27 T-Shirts left in inventory:  83
28 Brooks shoes left in inventory: 13
29 Asics shoes left in inventory: 11
30 Shorts left in inventory:     29
31
32 Receipt
33 Nike Shoes             x4 = 719.96
34 Under Armour T-shirts x5 = 149.95
35 Brooks shoes           x0 = 0.00
36 Asics shoes            x1 = 165.88
37 Under Armour shorts    x6 = 274.62
38
39 Amount before tax:     1310.41
40 Tax:                   101.56
41 Amount after tax:      1411.97
```

header.h

```
1 /*****
2  * AUTHOR      : Saul Moreno
3  * STUDENT ID   : 269491
4  * ASSIGNMENT # : 3
5  * CLASS       : CS1C
6  * SECTION     : MW 5:00pm
7  * DUE DATE    : 2/12/19
8  *****/
9
10 #ifndef HEADER_H_
11 #define HEADER_H_
12
13 #include <iostream> // input and output
14 #include <iomanip>   // setprecision and setw
15 #include <string>    // allows to use strings
16 #include <limits>    //
17 #include <ios>       //
18 #include <fstream>   // file in & out
19 #include <time.h>    // system time
20 #include <stdlib.h>  // srand and rand
21
22 void PrintHeader(std::string asName, int asNum, char asType);
23
24 #endif /* HEADER_H_ */
25
```

PrintHeader.cpp

```
1 /*****
2  * AUTHOR      : Saul Moreno
3  * STUDENT ID   : 269491
4  * ASSIGNMENT # : 3
5  * CLASS       : CS1C
6  * SECTION     : MW 5:00pm
7  * DUE DATE    : 2/12/19
8  *****/
9
10 #include "header.h"
11
12 /*****
13 *FUNCTION - PrintHeader
14 *
15 *This function receives an assignment name, type and number then outputs the
16 * appropriate header - returns nothing.
17 *
18 *PRE-CONDITIONS
19 *   asName: Has to be previously defined
20 *   asType: Has to be previously defined
21 *   asNum: Has to be previously defined
22 *
23 *POST-CONDITIONS
24 *   This function will output class heading.
25 *   <Post-conditions are the changed outputs either passed by value or
26 *   by reference OR anything affected by the function.
27 *
28 *****/
29
30 void PrintHeader(std::string asName, // IN - Assignment Name
31                 int asNum, // IN - assignment type
32                 // (LAB or ASSIGNMENT)
33                 char asType)// IN - assignment number
34 {
35     std::cout << std::left;
36     std::cout << "*****\n";
37     std::cout << "* PROGRAMMED BY : Saul Moreno\n";
38     std::cout << "* " << std::setw(14) << "STUDENT ID" << ": 269491\n";
39     std::cout << "* " << std::setw(14) << "CLASS" << ": CS 1C MW-5:00pm\n";
40     std::cout << "* ";
41     if (toupper (asType) == 'L')
42     {
43         std::cout << "LAB #" << std::setw(9);
44     }
45     else
46     {
47         std::cout << "ASSIGNMENT #" << std::setw(2);
48     }
49     std::cout << asNum << ": " << asName << std::endl;
50     std::cout << "*****\n\n";
51     std::cout << std::right;
52 }
53
54
55
```

main.cpp

```
1 /*****
2 * AUTHOR      : Saul Moreno
3 * STUDENT ID   : 269491
4 * ASSIGNMENT # : 3
5 * CLASS        : CS1C
6 * SECTION      : MW 5:00pm
7 * DUE DATE     : 2/12/19
8 *****/
9
10 #include "header.h"
11
12 /*****
13 * Pointer
14 * -----
15 * This program will show the user the amount of items in stock and their cost
16 * and subtract from the inventory then add up the cost before and after tax.
17 *****/
18 struct Inventory
19 {
20     std::string equipmentName; // The name of the equipment
21     double cost;               // The cost of the equipment
22     int quantity;              // The quantity in the inventory
23 }
24 };
25
26 namespace variable
27 {
28     const int PROMPT_COL = 50; // Column size for the cout
29     const int RECEIPT_COL = 22; // Column size for the receipt
30     const int INV_COL = 32; // Column size for the inventory
31     const int PRICE_COL = 7; // Column size for the price
32     int basketShoesAmount; // The # of Nike shoes being bought
33     int tShirtAmount; // The # of T-shirt being bought
34     int brooksAmount; // The # of Brooks shoes being bought
35     int asicsAmount; // The # of Asics shoes being bought
36     int shortAmount; // The # of shorts being bought
37     int basketInv; // Holds the amount of Nike shoes in the inventory
38     int tShirtInv; // Holds the amount of T-shirts in the inventory
39     int brooksInv; // Holds the amount of Brooks shoes in the inventory
40     int asicsInv; // Holds the amount of Asics shoes in the inventory
41     int shortInv; // Holds the amount of shorts in the inventory
42     double basketCost; // Holds the price of Nike shoes
43     double tShirtCost; // Holds the price of
44     double brooksCost; // Holds the price of Brooks shoes
45     double asicsCost; // Holds the price of Asics shoes
46     double shortCost; // Holds the price of shorts
47     double amtBeforeTax; // Holds the amount of the sale before tax
48     double tax; // Tax % being charged; 7.75.
49     double taxAmount; // Stores the amount that was taxed
50     double amtAfterTax; // The total value after tax
51 }
52
53 int main()
54 {
55
56     Inventory inventory[5]; // Creates an array that holds 5 elements
57     variable::tax = 7.75; // initializing the tax amount
```

```

58
59 //Calls the function to print out the author box
60 PrintHeader("Pointer", 3, 'A');
61
62 std::cout << "This program will show the items available for purchase\n"
63             << "and how many are in stock. Then after the user has\n"
64             << "selected how many they want the program will add up\n"
65             << "their total before and after tax\n\n";
66
67 inventory[0].equipmentName = "Nike basketball shoes";
68 inventory[0].cost = 179.99;
69 inventory[0].quantity = 25;
70 inventory[1].equipmentName = "Under Armour T-shirt";
71 inventory[1].cost = 29.99;
72 inventory[1].quantity = 88;
73 inventory[2].equipmentName = "Brooks running shoes";
74 inventory[2].cost = 121.44;
75 inventory[2].quantity = 13;
76 inventory[3].equipmentName = "Asics running shoes";
77 inventory[3].cost = 165.88;
78 inventory[3].quantity = 12;
79 inventory[4].equipmentName = "Under Armour shorts";
80 inventory[4].cost = 45.77;
81 inventory[4].quantity = 35;
82
83 std::cout << std::left;
84 std::cout << std::setw(variable::RECEIPT_COL) << "Name of Equipment"
85             << std::setw(variable::PRICE_COL) << "Cost" << "Quantity\n";
86 for(int count = 0; count < 5; count++)
87 {
88     std::cout << std::setw(variable::RECEIPT_COL)
89                 << inventory[count].equipmentName
90                 << std::setw(variable::PRICE_COL)
91                 << inventory[count].cost << " "
92                 << inventory[count].quantity << std::endl;
93 }// end for(int count = 0; count < 5; count++)
94
95 std::cout << std::endl;
96 std::cout << std::setw(variable::PROMPT_COL)
97             << "Enter how many Nike basketball shoes do you want? ";
98 std::cin >> variable::basketShoesAmount;
99
100 std::cout << std::setw(variable::PROMPT_COL)
101            << "Enter how many Under Armour T-shirts do you want? ";
102 std::cin >> variable::tShirtAmount;
103
104 std::cout << std::setw(variable::PROMPT_COL)
105            << "Enter how many Brooks shoes do you want? ";
106 std::cin >> variable::brooksAmount;
107
108 std::cout << std::setw(variable::PROMPT_COL)
109            << "Enter how many Asics running shoes do you want? ";
110 std::cin >> variable::asicsAmount;
111
112 std::cout << std::setw(variable::PROMPT_COL)
113            << "Enter how many Under Armour shorts do you want? ";
114 std::cin >> variable::shortAmount;

```

```

115     std::cin.ignore(1000, '\n');
116
117     std::cout << std::endl;
118     variable::basketInv = inventory[0].quantity - variable::basketShoesAmount;
119     variable::basketCost = inventory[0].cost * variable::basketShoesAmount;
120     inventory[0].quantity = variable::basketInv;
121     std::cout << std::setw(variable::INV_COL) << "Nike shoes left in inventory:"
122         << inventory[0].quantity << std::endl;
123
124     variable::tShirtInv = inventory[1].quantity - variable::tShirtAmount;
125     variable::tShirtCost = inventory[1].cost * variable::tShirtAmount;
126     inventory[1].quantity = variable::tShirtInv;
127     std::cout << std::setw(variable::INV_COL) << "T-Shirts left in inventory:"
128         << inventory[1].quantity << std::endl;
129
130     variable::brooksInv = inventory[2].quantity - variable::brooksAmount;
131     variable::brooksCost = inventory[2].cost * variable::brooksAmount;
132     inventory[2].quantity = variable::brooksInv;
133     std::cout << std::setw(variable::INV_COL)
134         << "Brooks shoes left in inventory:" << inventory[2].quantity
135         << std::endl;
136
137     variable::asicsInv = inventory[3].quantity - variable::asicsAmount;
138     variable::asicsCost = inventory[3].cost * variable::asicsAmount;
139     inventory[3].quantity = variable::asicsInv;
140     std::cout << std::setw(variable::INV_COL)
141         << "Asics shoes left in inventory:" << inventory[3].quantity
142         << std::endl;
143
144     variable::shortInv = inventory[4].quantity - variable::shortAmount;
145     variable::shortCost = inventory[4].cost * variable::shortAmount;
146     inventory[4].quantity = variable::shortInv;
147     std::cout << std::setw(variable::INV_COL) << "Shorts left in inventory:"
148         << inventory[4].quantity << std::endl;
149
150     std::cout << std::endl;
151     std::cout << "Receipt\n";
152     std::cout << std::setw(variable::RECEIPT_COL) << "Nike Shoes"
153         << "x" << std::fixed << std::setprecision(2)
154         << variable::basketShoesAmount
155         << " = " << variable::basketCost << std::endl;
156     std::cout << std::setw(variable::RECEIPT_COL) << "Under Armour T-shirts"
157         << "x" << std::fixed << std::setprecision(2)
158         << variable::tShirtAmount
159         << " = " << variable::tShirtCost << std::endl;
160     std::cout << std::setw(variable::RECEIPT_COL) << "Brooks shoes"
161         << "x" << std::fixed << std::setprecision(2)
162         << variable::brooksAmount
163         << " = " << variable::brooksCost << std::endl;
164     std::cout << std::setw(variable::RECEIPT_COL) << "Asics shoes"
165         << "x" << std::fixed << std::setprecision(2)
166         << variable::asicsAmount
167         << " = " << variable::asicsCost << std::endl;
168     std::cout << std::setw(variable::RECEIPT_COL) << "Under Armour shorts"
169         << "x" << std::fixed << std::setprecision(2)
170         << variable::shortAmount
171         << " = " << variable::shortCost << std::endl;

```

main.cpp

```
172
173 variable::amtBeforeTax = variable::basketCost + variable::tShirtCost
174                       + variable::brooksCost + variable::asicsCost
175                       + variable::shortCost;
176
177
178 std::cout << std::endl;
179 std::cout << std::setw(variable::RECEIPT_COL) << "Amount before tax: "
180         << std::fixed << std::setprecision(2) << variable::amtBeforeTax
181         << std::endl;
182 variable::taxAmount = (variable::amtBeforeTax * variable::tax) / 100;
183 std::cout << std::setw(variable::RECEIPT_COL) << "Tax: "
184         << std::fixed << std::setprecision(2) << variable::taxAmount
185         << std::endl;
186 variable::amtAfterTax = variable::amtBeforeTax + variable::taxAmount;
187 std::cout << std::setw(variable::RECEIPT_COL) << "Amount after tax: "
188         << std::fixed << std::setprecision(2) << variable::amtAfterTax
189         << std::endl;
190 std::cout << std::right;
191
192 return 0;
193 }
194
```