CH 5: Loops and Files (Part 2)

More on *for* loops

1. Counter (loop control) variable can be defined in the initialization expression.
2. Multiple statements can be placed in the in the initialization expression.
3. update expression can have multiple statements.
4. Any combination of expressions can be omitted.
5. Counter (loop control) variable can be defined in the initialization expression.

1. Counter (loop control) variable can be defined in the initialization expression.

```
for (int num = 0; num < 10 ; num ++)
{
    cout << num << endl;
}
```

2. Multiple statements can be placed in the in the initialization expression.

```
int sum = 0;
for(int num = 1; num <= 10 ; num ++)
{
   sum = sum + 1;
    cout << num << endl;
}
```

Vs

```
for(int num = 1, sum = 0; num <= 10 ; num ++)
{
   sum = sum + 1;
    cout << num << endl;
}
```

3. Update expression can have multiple statements.

```
for(int i =1 , j = 10; i <= 10 ; i ++, j--)
{
    cout << i  <<" " << j << endl;        //counting up and down
}
```

4. Any combination of expressions can be omitted.

Omitting initialization expression

```
int i = 1;
for( ; i <= 10 ; i ++)
{
    cout << i  << endl;
}
```

Omitting update expression

```
int i = 1;
for( ; i <= 10 ; )
{
    cout << i++  << endl;
}
```

Omitting all three expressions

```
int i = 1;
for( ; ; )
{
    cout << i  << endl;
}
```

Nested loops

Write a program to get the following output:

```
2 * 1 = 2
2 * 2 = 4
2 * 3 = 6
2 * 4 = 8
2 * 5 = 10

3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15

4 * 1 = 4
4 * 2 = 8
4 * 3 = 12
4 * 4 = 16
4 * 5 = 20
```

Inner loop goes through each of the iterations for each iteration of an outer loop
Inner loop completes its iterations faster than the outer loop.

Total number of loops equals to the multiplication of number of iterations of all loops.

e.g., 3 * 5 = 15

1. Write a code segment that outputs the following numbers with the same formatting using nested loops. You should not use hard coded numbers to get the output.
   (0, 0) (0, 1) (0,2) (0, 3) (0, 4) (0, 5)
   (1, 0) (1, 1) (1,2) (1, 3) (1, 4) (1, 5)
   (2, 0) (2, 1) (2,2) (2, 3) (2, 4) (2, 5)

Write a code segment that outputs the following numbers with the same formatting using nested loops. You should not use hard coded numbers to get the output.

(0, 2)
(0, 3)
(0, 4)
(0, 5)

(1, 3)
(1, 4)
(1, 5)
(1, 6)

(2, 4)
(2, 5)
(2, 6)
(2, 7)

Sentinels

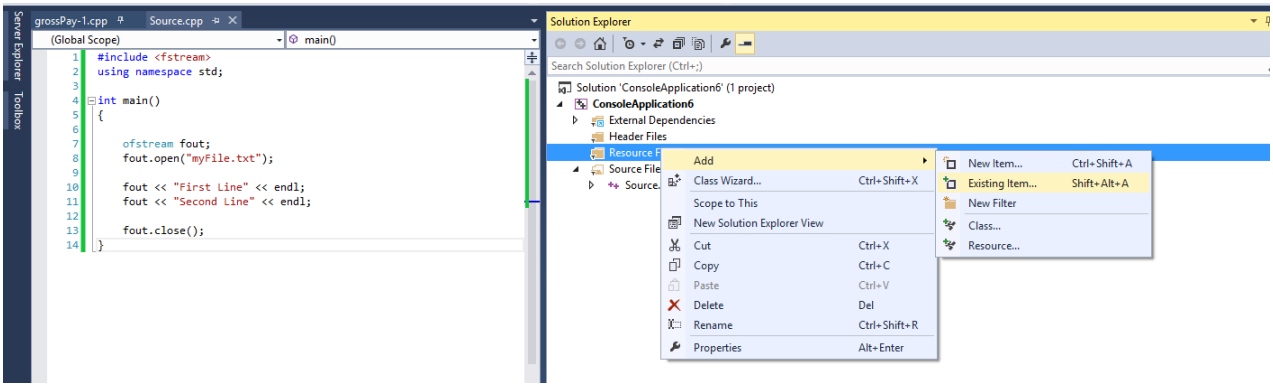
Sentinels is a special value that marks the end of a list of values.

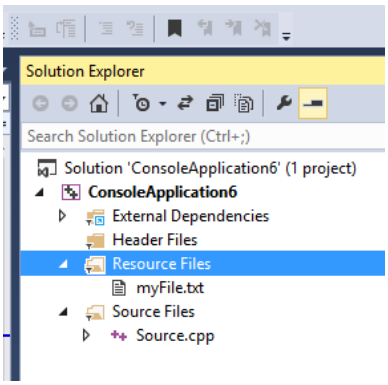
Writes the main() function of a program that reads a list of students' scores in and find the sum of all scores.

```
int score,
    sum = 0;
cout << "Enter the scores and enter a negative value to represent the end of the list";
cin >> score;

while (score >= 0)
{
   sum = sum + score;
   cin >> score;
}

cout << "The sum of all scores is : " << score;
```

Data expecting to read	**Sentinel value**
list of lengths	Any non-positive value or predefined specific negative value
List of costs	Any non-positive value or predefined specific negative value
List of letter grades	Any character other than A, B, C, D, F, such as Z or predefined specific character
List of numeric months	Any number not in range: 1-12 or predefined specific number
List of first names	Any predetermined word; e.g., end or done

Write into a file





Read from a file

Read every word in the file

obal Scope)

```
1  #include <fstream>
2  #include <iostream>
3  using namespace std;
4
5  int main()
6  {
7      int num;
8      ifstream fin;
9      fin.open("myFile2.txt");
10
11     if (fin)
12     {
13         while (fin >> num)
14         {
15             cout << num << endl;
16         }
17         fin.close();
18     }
19     else
20     {
21         cout << "Error!";
22     }
23 }
```

Read only 10 words

```
1  #include <fstream>
2  #include <iostream>
3  using namespace std;
4
5  int main()
6  {
7      int num;
8      ifstream fin;
9      fin.open("myFile2.txt");
10
11     if (fin)
12     {
13         for (int i = 0; i < 10; i++)
14         {
15             fin >> num;
16             cout << num << endl;
17         }
18         fin.close();
19     }
20     else
21     {
22         cout << "Error!";
23     }
24 }
```

Note: The file should be available in the project folder.

How to add a txt file