

# ***Visual Studio Help Session – Spring 2006***

## ***Visual Studio Help Session MadLab, Link 010***

Sample code in

[www.ecs.syr.edu/faculty/fawcett/handouts/CSE687/code/DemoVisualStudio](http://www.ecs.syr.edu/faculty/fawcett/handouts/CSE687/code/DemoVisualStudio)

Jim Fawcett

CSE687 – Object Oriented Design

6:30 pm – 9:00 pm

20 January 2006

# Visual Studio Help Session – Spring 2006

The screenshot displays the Microsoft Visual Studio IDE. The main editor window shows the source code for `stringBase.cpp`. The code includes a header file, defines a `stringBase` class with a constructor, a destructor, and a `body` method, and contains a `main` function for testing. The Solution Explorer on the right shows the project structure with folders for Header Files, Resource Files, and Source Files, and files for `stringBase.h`, `stringBase.cpp`, and `test.cpp`.

A callout box points to the Task List window, which is titled "Task List - 8 tasks". The window contains a table of user tasks:

!	Description
<input type="checkbox"/>	set up editor for spaces instead of tabs
<input type="checkbox"/>	build a class derived from string
<input type="checkbox"/>	call it from a test file
<input type="checkbox"/>	demo help system
<input type="checkbox"/>	demonstrate use of the debugger
<input type="checkbox"/>	show break points - simple, hit count, etc
<input type="checkbox"/>	show line numbers - options/text editor/all languages
<input type="checkbox"/>	show options/projectsAndSolutions

The status bar at the bottom indicates the current position is Ln 4, Col 16, Ch 16, and the system is Ready.

Task List:  
Go to view/Task List

# Visual Studio Help Session – Spring 2006

The screenshot shows the Visual Studio IDE with a C++ source file open. The code defines a `stringBase` class and a `main` function. The `main` function creates a `stringBase` object and prints its length. The `Class View` pane on the right shows the project structure, including the `stringBase` class and its members. A speech bubble points to the `stringBase` class in the `Class View` pane.

```
1 // stringBase.cpp - Visual Studio help session demo
2
3
4 #include "stringBase.h"
5
6 stringBase::stringBase(const std::string& tag)
7 {
8     *this += '<';
9     *this += tag + ">";
10    *this += tag;
11    *this += '>';
12 }
13
14 stringBase::~stringBase()
15 {
16     std::cout << " being destroyed\n";
17 }
18
19 void stringBase::body(const std::string& text)
20 {
21     size_t pos = this->find(">");
22     if (pos < this->length())
23         this->insert(pos+1, text);
24 }
25
26 #ifdef TEST_STRINGBASE
27
28 void main(int argc, char* argv[])
29 {
30     const std::string title = "Testing stringBase class";
31     std::cout << "\n " << title;
32     std::cout << "\n " << std::string(title.length()+2, '!') << "\n";
33 }
```

How do you find out about all the things you can do with `std::string`? Just go to class view, select `stringBase/BaseTypes` and select `string`.

# Visual Studio Help Session – Spring 2006

The screenshot displays the Microsoft Visual Studio IDE. The main window shows the source code for `stringBase.cpp`. A context menu is open over the function definition `void stringBase::body(const std::basic_string<char, char_traits<char>, allocator<char> > &text)`. The menu options include "Call Browser", "Go To Definition", "Go To Declaration", "Find All References", "Go To Header File", "Breakpoint", "Add Watch", "QuickWatch...", "Show Next Statement", "Step Into Specific", "Run To Cursor", "Set Next Statement", "Go To Disassembly", "Cut", "Copy", "Paste", and "Outlining".

A speech bubble points to the context menu with the text: "Find all the callers of body function: right-click on function def and select callers".

The "Callers Graph" window is open, showing a tree view of callers for the selected function. The callers are:

- stringbase.cpp(35): elem1.body("this is a body");
- stringbase.cpp(39): elem2.body("another body");
- stringbase.cpp(43): pDoc->body(elem1 + elem2);
- test.cpp(15): elem1.body("this is a body");
- test.cpp(19): elem2.body("another body");
- main(int argc, char \*argv[])
- main(int argc, char \*argv[])

The Solution Explorer on the right shows the project structure for "DemoVisualStudio" (2 projects), including "stringBase" and "test".

The status bar at the bottom indicates the current line is 19, column 20, and character 20.

# Visual Studio Help Session – Spring 2006

The screenshot displays the Microsoft Visual Studio IDE. The main window shows a C++ source file named `test.cpp` with the following code:

```
1 // test.cpp - demonstrate calling routines in another file
2
3
4 #include <iostream>
5 #include <fstream>
6 #include "../stringBase/stringBase.h"
7
8 void main(int argc, char* argv[])
9 {
10     const std::string title = "Using stringBase class";
11     std::cout << "\n " << title;
12     std::cout << "\n " << std::string(title.length()+2, '=') << "\n";
13
14     stringBase elem1("aTag");
15     elem1.body("this is a body");
16     std::cout << "\n " << elem1 << "\n";
17
18     stringBase elem2("bTag");
19     elem2.body("another body");
20     std::cout << "\n " << elem2 << std::endl;
21
22     new stringBase("anXMLdocument");
23     elem2);
24     " << "pDoc << "\n\n";
25
26
27
28
29
30
31     could not open temp.txt for writing";
32
33
34
```

A right-click context menu is open over the code, with the **Run To Cursor** option selected. A callout box points to this option with the text: "Right-click and select Run to Cursor to move to the next point of interest, while debugging."

The Solution Explorer on the right shows the project structure for "DemoVisualStudio (2 projects)":

- stringBase
  - Header Files
    - stringBase.h
  - Resource Files
  - Source Files
    - stringBase.cpp
- test
  - Header Files
    - stringBase.h
  - Resource Files
  - Source Files
    - stringBase.cpp
    - test.cpp

The Command Window at the bottom right shows the following output:

```
>SaveAll
>Action.Dependencies
Command "Action.Dependencies" is not available.
>Debug.stepov
Command "Debug.stepov" is not valid.
>Debug.stepover
>Debug.stepover
>Debug.stepover
>
```

The Windows taskbar at the bottom shows the system tray with the time 9:33 PM and the date 9/22/06.

# Visual Studio Help Session – Spring 2006

The screenshot shows the Visual Studio IDE with a C++ project named 'DemoVisualStudio'. The code editor displays the following code:

```
1 ////////////////////////////////////////////////////////////////////
2 // stringBase.cpp - Visual Studio help session demo
3
4 #include "stringBase.h"
5
6 stringBase::stringBase(const std::string& tag)
7 {
8     *this += '<';
9     *this += tag + "></";
10    *this += tag;
11    *this += '>';
12 }
13
14 stringBase::~stringBase()
15 {
16     std::cout << " being destroyed\n";
17 }
18
19 void stringBase::body(const std::string& text)
20 {
21     size_t pos = this->find(">");
22     if (pos < this->length())
23         *this->insert(pos+1, text);
24 }
25
26 #ifdef TEST_STRINGBASE
27
28 void main(int argc, char* argv[])
29 {
30     const std::string title = "Testing stringBas
31     std::cout << "\n " << title;
32     std::cout << "\n " << std::string(title.length()+2, '=') << "\n";
33 }
```

A breakpoint is set on line 21. A speech bubble points to it with the text: "Right click on breakpoint to get context menu and select something useful."

The 'When Breakpoint Is Hit' dialog box is open, showing the following options:

- Print a message: Function: \$FUNCTION, Thread: \$TID \$TNAME
- Run a macro:
- Continue execution

The Command Window shows the following output:

```
>SaveAll
>Action.Dependencies
Command "Action.Dependencies" is not available.
>Debug.stepov
Command "Debug.stepov" is not valid.
>Debug.stepover
>Debug.stepover
>Debug.stepover
>Debug.stepover
>
```

The Autos window shows the following variables:

Name	Value	Type
pos	5	unsigned
this	0x0012ff0c	stringBas