

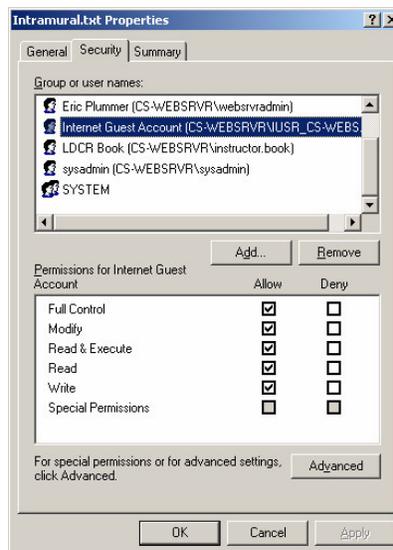
IT350 Web and Internet Programming

Fall 2005

SlideSet #14: File I/O

File Access

- Ownership: Input/Output files usually **NOT** owned by “Web Server”.
 - Operating system may enforce read, write, and/or modify restrictions on I/O files
 - For file output/append, may need to create file prior to first use
 - File permissions need set for access by the “web server” account (Right-click on file, pick Properties, then set permissions like example on right)



File Access

- File Path/Naming
 - CGI Script may **NOT** run within script's location
 - May need to provide full or relative path information
 - All CGI processes on *cs-websrvr* are run from the top directory of the account and require path information to Input/Output file's location
 - Use `getThisPath()` helper (in `CGI.cpp`):

```
string path = getThisPath();
string filename = path + "LOG.txt";
```
 - Path names may require special characters to be escaped
 - On Windows systems, path directories are separated by a single '\'. However, this symbol is used to initiate an escape sequence in C++. Therefore to embed a '\' in a path name, it must be preceded by another '\' character.
Ex: "cgi-bin\\Parse\\Debug\\Intramural.txt"

Using Files

- Input Files
 1. Include the *fstream* header file (i.e: `#include <fstream>`)
 2. Declare *ifstream* objects and open using the `open()` member function, Ex:

```
ifstream fin;
fin.open(filename.c_str());
```
 3. Use standard stream syntax to read from the stream object, Ex:

```
while (!fin.eof()) {
    fin >> inputValueVariable;
    // do something with inputValueVariable
}
```
 4. Ensure you close the stream when finished using the `close()` member function

```
fin.close();
```

Using Files

- Output Files

1. Include the *fstream* header file (i.e: #include <fstream>)

2. Declare *ofstream* objects and open using the **open()** member function, Ex:

```
ofstream fout;  
fout.open(filename.c_str());
```

3. Use standard stream syntax to write to the stream object,

Ex:

```
fout << inputValueVariable;
```

4. Ensure you close the stream when finished using the **close()** member function

```
fout.close();
```

Using Files

- Appending to Files

1. Include the *fstream* header file (i.e: #include <fstream>)

2. Declare *ofstream* objects and open using an overloaded version of the **open()** member function, Ex:

```
ofstream fapp;  
fapp.open(filename.c_str(), ios::out|ios::app);
```

3. Use standard stream syntax to append to the stream object,

Ex:

```
fapp << inputValueVariable;
```

4. Ensure you close the stream when finished using the **close()** member function

```
fapp.close();
```