



# Cómo exportar código fuente a documentos PDF

Código en papel como si estuvieras en el IDE



# Exportar código fuente a un único fichero

Para llevar todo el contenido de los ficheros con una cierta extensión a un único fichero de salida se puede utilizar el siguiente comando, en cualquier distribución de Linux:

```
find . -type f -name '*.txt' -exec cat {} + >> output.file
```

En resumen, el comando lee y concatena todos los ficheros que coincidan con el patrón de búsqueda (extensión .txt en el ejemplo) en el fichero output.file.

Es importante que el fichero de salida tenga la misma extensión que los ficheros de entrada, así, output.txt no sería un nombre válida para el fichero de salida, ya que lo tomaría como un fichero más de entrada y entraría por tanto en un bucle infinito.

El propio comando aborta la ejecución si cometemos este error.



# Parámetros del comando

```
find . -type f -name '*.txt' -exec cat {} + >> output.file
```

- `.`
  - Indica que se busque en el directorio actual.
- `-type f`
  - Solo buscar ficheros (omitir directorios)
- `-name '*.txt'`
  - Cuyo nombre de archivo coincida con el patrón. En este caso, cualquier fichero con extensión `.txt`
- `-exec cat {} +`
  - Para cada resultado ejecuta el comando `cat`
  - El parámetro `+` indica que solo se utilice una instancia de `cat` simultáneamente
- `>> output.file`
  - Redireccionar la salida al final del fichero `output.file`



# Exportación a PDF

Para exportar a PDF seguimos los siguientes pasos:

1. Subimos los ficheros generados a Github o Gitlab
2. Hacemos click en el fichero que queremos exportar (se nos muestra el código)
3. Botón derecho en cualquier lugar de la pantalla y hacemos click en “Imprimir”
4. Esperamos que se genere la vista previa y hacemos click en Guardar.
  - a. Comprobar que el destino es “Guardar como PDF”



Add LMF's source code unified on 2 files

Alexander Ruiz authored 1 week ago

2e54c6fb

output.h 107 KB

```

1 #ifndef H1_Analyzer
2 #define H1_Analyzer
3
4 #include <fftw3.h>
5 #include <stdlib.h>
6 #include <stdio.h>
7 #include "SoundGrabber.h"
8 #include <math.h>
9 #define FFT_F2D_TYPE FFTW_MEASURE
10
11
12 extern int gmethodtoastercenterfr[32];
13 extern char *gmethodtoastercenterfrlabel[32];
14
15 class Analyzer
16 {
17 public:
18     fftw_plan fftplanleft, fftplanright;
19
20     unsigned char *audio_buffer;
21     signed short *application_buffer;
22     double *left_buffer;
23     double *right_buffer;
24
25     fftw_complex *spectrum_buffer;
26     double *abs_spectrum_buffer;
27
28     Analyzer();
29
30     /* read samples from soundcard and calculate spectrum */
31     int analyze(SoundGrabber &soundgrabber, double subbands[2][32]);
32
33     /* Analyze readed samples and calculate spectrum */
34     int analyze(float *leftBuffer, float *rightBuffer, int len, double subbands[2][32]);
35
36     /* compute absolute values of complex_buffer, save to abspec_buffer
37     take care that - complen is not bigger than complex_buffer!
38     - abspec_buffer is big enough!
39
40     returns 0 on success, else -1 */
41     int abspec(fftw_complex *complex_buffer, double *abspec_buffer, int compl_length);
42
43     /* compute 31 subband 1/3 octave spectrum
44     take care that subbands[] array is 31 elements long
45
46     requires: abs spectrum from fft, subband array [31] to save, length of abspectrum,
47     maximum frequency (half sampling freq.)
48
49     returns 0 on success, else -1 */
50     int calc13octaveband31(double *abspec_buffer, double *subbands, int num_spec, double max_frequency);
51 };
52
53 #endif
54 #ifndef H1_SoundGrabber
55 #define H1_SoundGrabber
56
57 #include <stdlib.h>
58 #include <stdio.h>
59 #define ALSA_PCM_NEW HW_PARAMS_API
60 #include <alsa/asoundlib.h>
61
62 /* Sound grabber default settings */
63 // number of samples (in bytes) from soundcard and therefore length of fft
64 #define DEFAULT_BUFFER_SIZE 4096 * 2 // should be something like 2^n
65 #define DEFAULT_APPBUFFER_SIZE DEFAULT_BUFFER_SIZE / 2 // should be DEFAULT_BUFFER_SIZE/2 for 32 bit sampling dep
66 // mono 1 - stereo 2
67 #define DEFAULT_NUM_CHANNELS 2
68 // sampling rate

```

## Imprimir

67 páginas

Destino

Guardar como PDF

Páginas

Todo

Páginas por hoja

1

Márgenes

Predeterminados

Opciones



Encabezado y pie de página



Gráficos de fondo

Guardar

Cancelar



# Exportación a PDF

Como paso opcional, podemos evitar que aparezca la barra superior indicando el último commit y el usuario que lo hizo. Para ello, antes de exportar a PDF hacemos lo siguiente:

1. Click derecho sobre la barra y seleccionamos “Inspeccionar”
2. Se nos abre el código, buscamos el bloque de código que contiene la barra y presionamos la tecla Suprimir
3. Ahora exportamos a PDF según la diapositiva anterior, y vemos que la barra ha desaparecido.

div.info-well.d-none.d-sm-block

1138.92 x 75.8 -lm7-lm9 / lms.9 / output.h

Find file

Blame

History

Permalink



Add LM9's source code unified on 2 files

Alejandro Ruiz authored 1 week ago

output.h 137 KB

```
1 #ifndef HH_Analyzer
2 #define HH_Analyzer
3
4 #include <fftw3.h>
5 #include <stdlib.h>
6 #include <stdio.h>
7 #include "SoundGrabber.h"
8 #include <math.h>
9 #define FFT_FIND_TYPE FFTW_MEASURE
10
11 extern int onethirdoctavecenterfr[31];
12 extern char *onethirdoctavecenterfrlab
13
14 class Analyzer
15 {
```

DevTools - git.granasat.space/limitador\_electroacustico/reverse-engineering-lm7-lm9/-/blob/master/lms.9/output.h



Elements

Console

Sources

Network

Performance

Memory

Application

Security

Lighthouse

Adblock

```
<main class="content" id="content-body" itemscope itemtype="http://schema.org/SoftwareSourceCode">
  <div class="flash-container flash-container-page sticky" data-qa-selector="flash_container">
  </div>
  <div class="js-signature-container" data-signatures-path="/limitador_electroacustico/reverse-engineering-lm7-lm9/-/commits/2e54c6fbeb51ebcb2389cbd0b5724670b42b1c55/signatures?limit=1"></div>
```

```
<div class="tree-holder" id="tree-holder">
```

```
  <div class="nav-block">...</div>
```

```
... <div class="info-well d-none d-sm-block"> == $0
```

```
  <div class="well-segment">
```

```
    <ul class="blob-commit-info">
```

```
      <li class="commit flex-row js-toggle-container" id="commit-2e54c6fb">
```

```
        <div class="avatar-cell d-none d-sm-block">...</div>
```

```
        <div class="commit-detail flex-list">...</div>
```

```
      </li>
```

```
    </ul>
```

```
  </div>
```

```
</div>
```

```
<div class="blob-content-holder" id="blob-content-holder">...</div>
```

```
<div class="modal" id="modal-remove-blob">...</div>
```

```
<div class="modal" id="modal-upload-blob">...</div>
```

```
</div>
```

```
</main>
```



# Fuentes

[How to append contents of multiple files into one file, StackOverflow](#)