

Display Meta Data

Introduction

This task will add a command to the instrument designer program that you have already created in the assignments. This command will display the meta data configured in the audio track and the WAV file objects that are stored in the application data object.

The additional command is listed in this table.

Command	Prefixable?	Function	Description
<code>display-audio-track-and-wav-file-meta-data</code>	no	<code>display_audio_track_and_wav_file_meta_data</code>	Display meta data configured in the audio track and WAV file.

Example Usage

This is an example to see the command in use. The `...` in the menu output are not literal. They are replacing irrelevant information.

```
$ ./program-instrument-designer/instrument_designer
Choice? menu
Options are:
...
display-audio-track-and-wav-file-meta-data - Display meta data configured in the audio track and WAV
file.
...

Choice? configure-audio-track-and-wav-file
Samples/Second: 1234
Seconds: 1.234
Bits/Sample[8,16,24,32]: 16
Choice? display-audio-track-and-wav-file-meta-data
Audio track Samples/Second: 1234
Audio track Seconds: 1.234
WAV file Samples/Second: 1234
WAV file Bits/Sample: 16
Choice? quit
```

Programming Requirements

These files already exist from the homework. Add to them to complete this task.

Update `library-commands/instrument_designer_aux.{h,cpp}`

Functions:

- `void display_audio_track_and_wav_file_meta_data(ApplicationData& app_data);` - Display meta data configured in the audio track and WAV file. See the Example Usage above to get the formatting correct.
- `int register_instrument_designer_commands(ApplicationData& app_data);` Update this function to register the new command.

Grading Instructions

To receive credit for this task:

- Your code must be pushed to your repository for this class on GitHub.
- All unit tests for assignments and this task must pass.
- All acceptance tests for assignments must pass.
- All programs must build, run, and execute as described in the assignment descriptions.