

Remove Waveform

Introduction

This task will add a command to the instrument designer program that you have already created in the assignments. This command will allow the user to remove waveform objects from the internal storage.

The additional command is listed in this table.

Command	Prefixable?	Function	Description
<code>remove-waveform</code>	no	<code>removeWaveformUI</code>	Remove waveform from the inventory.

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:
...
remove-waveform - Remove waveform from the inventory.

Choice? add-waveform
Waveform name: fred
Waveform type: sine
Amplitude: 1.0
Choice? list-waveforms
fred : fred sine 1
Choice? remove-waveform
Waveform name: fred
Choice? list-waveforms
Choice? remove-waveform
Waveform name: barney
Unable to find a waveform with name 'barney'.
Choice? quit
```

Programming Requirements

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

Update `library-waveform/Waveforms.{h,cpp}`

Methods:

- `void removeWaveform(const std::string& name);` Uses the `erase` method of the `map` to remove the named waveform from the inventory.

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

Functions:

- `void removeWaveformUI(ApplicationData& app);` Removes a waveform from the inventory, after asking the user for the name. See the example above for the expected output.
- `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.

