

# Quantum Menus Starter Documentation

This documentation is not complete. The full version is available online at <https://quantumtekhub.com/unity/menus/>  
This documentation shows how to start using this asset.  
Support for this asset can be found at <https://github.com/QuantumTekSupport/QuantumMenus/issues>

Email support:

- [support@quantumtekhub.com](mailto:support@quantumtekhub.com) for bugs/problems or questions on how to use the website/asset.
- [unity@quantumtekhub.com](mailto:unity@quantumtekhub.com) for feature requests or asset requires.

This asset requires TextMeshPro before the asset can be imported. TextMeshPro is a free Unity package available through the package manager: Window > Package Manager.

## Getting Started

- This asset is made up of menus.
- Each menu can have windows and tab groups.
- Tab groups are like windows, but have multiple windows inside them, show/hidden based on tabs.
- Only one tab window can be open at a time in a tab group.
- An unlimited number of windows and tab groups can be opened/closed whenever, using the function for it:  
ChangeOpenState(bool pOpen). pOpen is whether or not the window/tab group should be open. This function is the same for windows, tab groups, tab windows, and tabs.  
When opening a window or tab group, use this function.

- When changing the tab of a tab group, use the function `ChangeTab(QMTabWindow pTab)`. `pTab` is a reference to the `QMTabWindow` that should be opened.
- All windows and tab groups can be opened with use of the `QMMenu` function: `ChangeAllOpenState(bool pOpen)`.
- Similarly, all windows or all tab groups can be opened using the respective `QMMenu` functions: `ChangeWindowOpenState(bool pOpen)` and `ChangeTabGroupOpenState(bool pOpen)`.
- `QMAudio` is a Component that can play audio with a function call. Using this, a button or some other UI element can call that function upon click, or whatever event.
- `QMSave...` are Components that save the setting of the particular ... UI element upon change, and load the saved setting upon scene load.
- Important `QMMenu` functions include `LoadScene(int pBuildIndex)`, and `LoadScene(string pSceneName)`, which load asynchronously (in the background) the given scene by build index or scene name. This is usually combined with the `QMBAR` to create a loading screen.
- Another `QMMenu` function is `Quit()`, which closes out of the application. It does nothing in the Unity Editor.
- Some custom UI elements were made, `QMBAR`, and `QMOptionList`.
- `QMBAR` is a way of showing a progress bar, with different methods.
- `QMOptionList` gives a list of options to choose from, and the current option is changed with the arrow buttons next to it. The current option index is named `currentOption` in the code. The current option as a string can be retrieved with the function `GetOption`. The functions `ChangeOption` and `SetOption` can be used to change/set the option by/to a certain integer.

- Custom inspectors for QMMenu, QMWindow, QMTabGroup, and QMTab provide buttons for certain functions, such as aligning tabs. When changing the animation type of QMWindow, QMTabGroup, and QMTab, the custom inspector will show different required fields.
- A main menu prefab named “Q Main Menu” can be found under “Quantum Tek/Quantum Menus/Resources/Prefabs”.
- Prefabs for UI elements are under the directory “Quantum Tek/Quantum Menus/Resources/Prefabs/Elements”. These include things like buttons and a dropdown.
- Prefabs for basics menu Components are under the directory “Quantum Tek/Quantum Menus/Resources/Prefabs/Menu”. These include the QMMenu, QMWindow, QMTabGroup, and QMTabWindow prefabs.
- The demo scene can be found under “Quantum Tek/Quantum Menus/Demo”. The demo showcases all prefabs.

## Terminology

- QMMenu - An object that is made up of QMWindows and QMTabGroups as child objects. Any number of QMWindows and QMTabGroups can be opened at a time.
- QMWindow - A graphic that contains multiple UI elements and can be opened/closed.
- QMTabGroup - Like a QMWindow as it can be opened/closed, but it contains multiple QMTabWindows, one of which can be opened at a time.
- QMTabWindow - Like a QMWindow, but only one is shown at a time in a QMTabGroup. Each QMTabWindow has to have a QMTab as a child.

- QMTab - A Component that acts like a button to set the current QMTabWindow in a tab group.
- QMBar - A UI element that acts as a progress bar, and can fill in different ways: change in width, change in height, and change in fillAmount (an Image Component field).
- QMOptionList - A UI element like a dropdown, but displays text and buttons to change the option.
- QMAudio - A Component that plays audio upon function call.
- QMSaveDropdown - A Component that saves/loads the setting for a dropdown, by default there is no setting.
- QMSaveInput - A Component that saves/loads the setting for an input field, by default there is no setting.
- QMSaveOptionList - A Component that saves/loads the setting for an option list, by default there is no setting.
- QMSaveSlider - A Component that saves/loads the setting for a slider, by default there is no setting.
- QMSaveToggle - A Component that saves/loads the setting for a toggle, by default there is no setting.