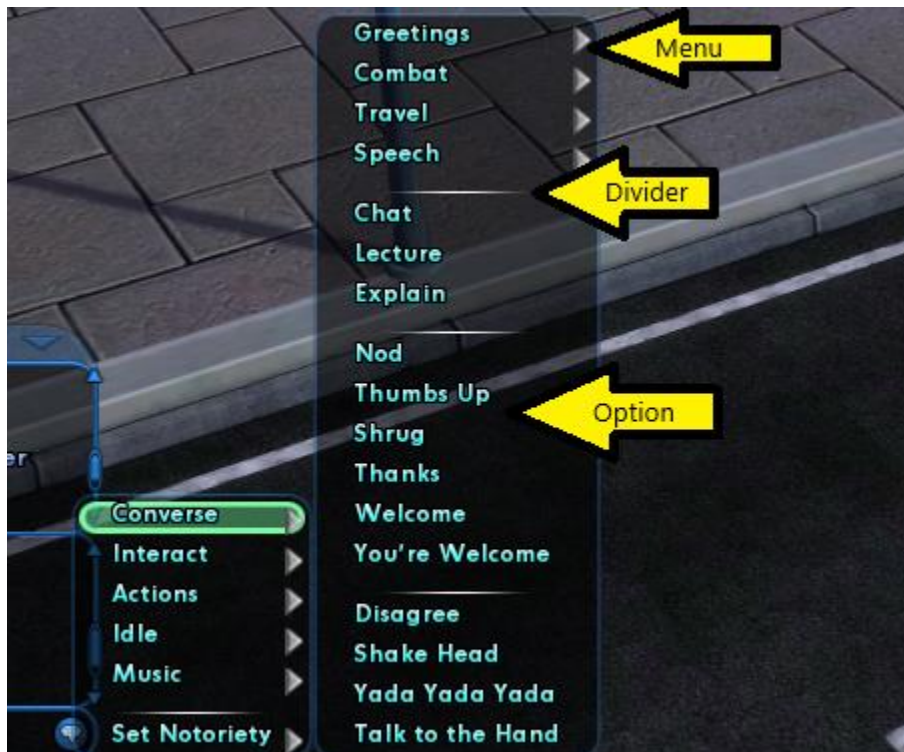Quick Chat Menu

The file quickchat.mnu can be used to define new quickchat items. (The menu to the right of the input box in chat where dances, emotes, etc. are.)

EXAMPLE:

Menu "More &Combat"
        {
                Option "&RUN!"   "say <color white><bgcolor red>RUN!"
                Divider
                Option "Disagree"   "e Disagree"
        }
}

STRUCTURE:

1. "Menu <name>" -- Defines a new (sub)menu starting point, you can nest this.
        Nested menu example:
        Menu QuickChat
        {
          Menu Greetings
           {

-The more nests you have, the more clicks a person must go thru to get to the actions.
-Also, it is important to surround the name of your menu with double quotes if it is more than one word OR a possible conflicting reserved word. ("Other Commands")

2."Divider" – This sets up a dividing line within the same menu OR in between menus within menus.

   See the example image for what the expected outcome of a divider is

   You can place these between anything as long as it is in quickchat or another menu type.

3. Option "<&><name>"<<3 TAB SPACES?>>"<Str OR Emote/Command>$$<Emote/Command>"

   Breakdown
   1. Option
      a. Defines that the next lines will be selectable options with a defined function.
      b. Requires 2 things, a "Name" in double quote and a "Str$$command" in double quotes.
      c. This basically works as if you were binding, but setting it up in a menu format, so think of the second arg just as if you were binding a key.
   2. &
      a. This defines that the key that come immediately after this symbol will be the *hotkey* while the player is in the current menu
      b. NOTE: You cannot use the same hotkey for 2 different things in the menu BUT you *can* use the same hotkey in any other submenu.
         i. For instance. Taking the example image, if the option "Chat" was "&Chat" then a player could press **C** in that menu to instantly open that. *However, you could not use C as a hotkey anywhere else IN THAT CURRENT MENU*.
         ii. BUT if a player selects the "Greetings" submenu, and "&Chat" also happened to be in there, the C would refer to Chat *only in the context of that submenu.*
   3. Name
      a. Whatever you want the label to be. Can be space or contain special characters.
         i. Unsure if putting a number first messes with it, but probably not.
   4. <<3 TAB SPACE>>
      a. This is the default amount of space in the menu folder between the Name and the Action.
   5. Str or Emote/Command
      a. You can define just a String to be put in a specific chat.
         i. "say Hello World!" will print it to local chat when selected
         ii. "team Hello World!" will print it to team chat when selected
         iii. "e RaiseHang" will perform that emote when selected
         iv. Etc.
      b. You can also just have it run a standard emote here
      c. In theory you could have it run any command you could bind in game.
      d. You can also pass $args into this like $target, $arechetype, etc.
      e. You can pass other code as well, such as <scale .75> or <color red>
   6. $$emote/command
      a. Just like in game the $$ joins the previous String, Emote, or Command to the stuff before it.
      b. Works just as #5

7. SPECIAL: LockedOption
   a. I think everything is unlocked for everyone on i25, but if you wanted to lock some emotes away, you could use Locked Option to hide those.
   b. Example
      i. LockedOption
      ii. {
         1. DisplayName "<Name>"
         2. Command "<command or str>
         3. Authbit RogueAccesss –what allows you to unlock this option
         4. StoreProduct cucppsal –Internal Product ID
         5. Badge CouncilRobotBadge <badge> <badge> --what badges unlock this option. Separate with a single space for multiples.
         6. RewardToken EmoteSwoon –if you know the emote reward token, this would grant the option.
      iii. }
      iv. NOTE: Only #1 and #2 are TECHNICALLY required.
      v. If you use #3, you must also include #4 (I think…)
      vi. #5 and #6 can stand on their own, only needs #1, #2.
   c. NOTE: This should not overwrite any access-level-restricted commands like /mmm.
      i. Probably worth actually testing.

It is also possible to make your own popup menus, this one is just attached to the text input box by default.

If you do make your own menu, make sure the file extension is .mnu and place it in you *\data\texts\English\Menus folder. Then, once in-game, you should just have to type /popmenu <Menuname>

This could be used to define some QOL commands for customized public-private servers.

DISCORD:Cloudskipper#2196