MSGID/ORDER/CJCS//

REF/A/DOC/-///

SUBJ/QUICK START INSTRUCTIONS//

NARR/

Congratulations on your new command, the USS Lubbock (SSN-806). Unlike a normal command, the circumstances of your mission will be of your own choosing. Your basic workflow should proceed as follows:

1. The default map is titled “default\_map.kml”. See Figure 1. See section 1.a. on how to edit the scenario.
2. Open any web browser to the following address:
   1. <http://127.0.0.1:8050/>
   2. This should load a blank internet screen.
3. Using your Python command prompt:
   1. Navigate to the directory containing the program files.
   2. Enter the following command:
      1. **python USS\_LUBBOCK.py**
4. You are now underway with the following settings:
   1. CRS – 090
   2. SPD – 5 kts
   3. Depth – 150 ft
5. You may change course, speed, and depth by entering new values into the available boxes. See section 2.c.i. for more. There are no limitations on speed and depth; those will be up to an intelligent operator to abide by real world constraints for training value. No simulation of acoustic-related limitations of your ship are modeled as well, for obvious reasons.

GENTEXT/EXAMPLE

1. (U) Building the Operating Area (OA)
   1. You may use ArcGIS or Google Earth (or any related KML editing program) to create your battlespace. Ensure any battlespace you create is saved in the same directory as the program files, replacing the “default\_map.kml” file with your new file under the same name. Running USS\_Lubbock.py will automatically load this map.
   2. Merchants
      1. Merchants need three placed points titled ‘A’, ‘B’, and ‘C’. Fifty merchants will spawn evenly between ‘A’ and ‘C’. One merchant will be released from one of those points at random every 120 seconds. Merchants from ‘A’ will drive to ‘B’, and having reached ‘B’, will drive to ‘C’. Merchants spawning at ‘C’ will drive to ‘A’ through ‘B’ in a similar fashion. Merchants move at 16kts and do not exhibit any other behavior.
   3. Fishing Vessels
      1. Ten fishing vessels will spawn at the point titled “Fishing”. After spawning semi-randomly over a small area, they will drive in small circles. Fishing vessels do not exhibit any other behavior.
   4. Submarine:
      1. The submarine needs one point labeled ‘Submarine\_Start’.
   5. Default Map

 Figure 1.

1. (U) Playing the Game
   1. For our example scenario, the operator is given the following task:
      1. PENETRATE THE MERCHANT TRANSIT LANES, FISHING AREAS, AND WARSHIP SCREENS TO THE INTERIOR OF MONTEREY BAY FOR A TLAM STRIKE AGAINST AN INSURGENT TRAINING COMPOUND LOCATED NEAR DEL MONTE BEACH.
   2. Display
      1. See figure 2.
      2. Your position is given as a latitude and longitude.
      3. Your current course is labeled ‘crs’.
      4. Your current speed is labeled ‘spd’.
      5. Your current depth is labeled ‘depth’.
      6. Bearings are plotted relative to ownship’s head, which is at 0 degrees on the horizontal axis. The legend will contain the true bearing of each contact.
      7. The screen will refresh every 3 seconds. Allow the game to run for 2 minutes before playing. You will notice the game resets a few times, which is normal. After the simulation is stable for more than 60 seconds, the game should run smoothly.
   3. Changing Course
      1. To change the value of course, speed, or depth **highlight the number you wish to alter and then type in your new course. DO NOT BACKSPACE.**
2. (U) Exporting Data:
   1. Once you have competed the exercise, you will find that the historical data has been plotted for your convenience under ‘Waterspace.kml’.
3. (U) Conclusion:
   1. Now that you have a scenario created and understand how to control your submarine, run your file via step 3.b. of the Quickstart! It would be helpful to you to remember your thumb-rules such as Eklund ranging… Good luck and good hunting!

Graphical user interface, application

Description automatically generated

Figure 2.

DECL/04JAN2026