Procedure for creating a uC/OS II project:

Let us assume we want to create a uC/OS II project called "demo1" which is a simple multitasking LED blinking demo project. This procedure is designed for Keil uVision V4.20. Make sure you have a similar version of Keil installed on your machine and locate the "*Raw uC/OS-II*" folder. Also note that you **must** create a new folder for every new project. Follow these steps.

1. Open the '*Raw uC/OS-II*' folder. Copy the folder '*app*', '*uCOS-II*', '*include*' and '*ScatterFile.sct*'.
2. Create a folder at any location you wish to, and rename it to your name or roll number or whatever you wish to name your folder. Let's say we name that folder as 'demo1'.
3. Open the demo1 folder and paste the contents you copied from step 1.
4. Now minimize all the windows and open *Keil uV4*.
5. Go to '*Project -> New uVision Project*'.
6. You will be asked to select a location for the project and name of the project. Browse and open the demo1 folder. Then give a name to the project (say "demo1") and click save. The name of the project and the folder need **not** be the same. The name of the project "demo1" has **no** extension, type only the name.
7. Next, select the device '*NXP (founded by Philips) -> LPC2148*'.
8. You will be asked if you wish to copy the startup code to the project. Click **NO.**
9. You should now have an empty project on your screen. In the top left corner of the screen click on the + symbol next to '*Target1*'. You should see '*Target1 -> Source Group1*'.
10. Right click on '*Source Group 1*' and click on '*Remove group Source Group 1 and its files*'. So now you are left with only '*Target1*'.
11. Now, right click on '*Target 1*' and select '*Add Group*'. Rename that group to '*uCOS II*'.
12. Similarly, add two more groups called '*app*' and '*includes*'. This makes your project structure look something like this

```
Target 1  +    ->    uCOS II
               ->    app
               ->    includes
```

13. Now right click on the group '*uCOS II*' and select "*Add files to group uCOS II*'. Browse into the uCOS II folder. Change the '*Files of type*' option to '*All Files**.*'. Select all the files or press CNTL+A. Then click on ADD and close.
14. Similarly, right click on groups '*app*' and '*includes*' and add all the files of those respective folders. So now we have all the files from the folders from step 3 added to the respective groups in Keil.
15. Now, go to '*Project -> Options for Target 1*' or right click on '*Target 1' and then click on 'Options for Target 1*'.
16. The '*Options for Target 1*' window will be open. Go to the '*Output*' tab and tick on the '*Create HEX file*' option.
17. Then go to '*Linker*' tab and make sure that the '*Use memory layout from target dialogue*' is **un-tick**. Click the browse or the '…' button in front of the Scatter File option and browse and add the "*ScatterFile.sct*" from step3.
18. Then go to '*C/C++*' tab. In the '*Include Paths*' option click browse or the '…' button. Click on '*New (Insert)*' and click on the browse or the '…' button. Browse and select the include folder in *"……\demo1\include*".
19. Our uCOS II project is now completely set-up. To verify everything is OK go to '*Project -> Build Target*'. After compilation you should see that the HEX file is been created and there are *0 errors*. Ignore the warnings. You can now proceed with the programming.
20. Expand the '*Target1 -> app*' group and open the file '*app.c*'.
21. In this file you file find that the provision for 4 tasks (1 parent and 3 child tasks) has already been made for you, along with their priorities, stacks and task definitions along with board specific initializations.
22. Find the section where AppTask0 is defined (most probably line 147). Inside the *while(1)* loop type your code to toggle *LED0* and call *OSDelay( )* function.
23. Similarly in the AppTask1 and AppTask2 section type your code to toggle LED1 and LED2 respectively.
24. Compile the code and generate the HEX file.
25. Burn the *demo1.hex* file into the microcontroller using the Flash Magic utility.

Happy programming ;)