|  |
| --- |
| # Name of the project |
|  | PROJECT\_NAME = function |
|  |  |
|  | # Output directory |
|  | BUILD = build |
|  |  |
|  | # All source code files |
|  | SRC = main.c\ |
|  | src/function.c\ |
|  |  |
|  | # All test source files |
|  | TEST\_SRC = src/function.c\ |
|  | test/test\_function.c\ |
|  |  |
|  | TEST\_OUTPUT = $(BUILD)/Test\_$(PROJECT\_NAME).out |
|  |  |
|  | # All include folders with header files |
|  | INC = -Iinc |
|  |  |
|  | PROJECT\_OUTPUT = $(BUILD)/$(PROJECT\_NAME).out |
|  |  |
|  | # Document files |
|  | DOCUMENTATION\_OUTPUT = documentation/html |
|  |  |
|  | # Default target built |
|  | $(PROJECT\_NAME):all |
|  |  |
|  | # Run the target even if the matching name exists |
|  | .PHONY: run clean test doc all |
|  |  |
|  | all: $(SRC) $(BUILD) |
|  | gcc $(SRC) $(INC) -o $(PROJECT\_OUTPUT).out -lm |
|  |  |
|  | # Call `make run` to run the application |
|  | run:$(PROJECT\_NAME) |
|  | ./$(PROJECT\_OUTPUT).out |
|  |  |
|  | # Document the code using Doxygen |
|  | doc: |
|  | make -C ./documentation |
|  |  |
|  | # Build and run the unit tests |
|  | test:$(BUILD) |
|  | gcc $(TEST\_SRC) $(INC) -o $(TEST\_OUTPUT) -lcunit -lm |
|  | ./$(TEST\_OUTPUT) |
|  |  |
|  | # Remove all the built files, invoke by `make clean` |
|  | clean: |
|  | rm -rf $(BUILD) $(DOCUMENTATION\_OUTPUT) |
|  |  |
|  | # Create new build folder if not present |
|  | $(BUILD): |
|  | mkdir build |