Lab 02: Understanding uvm_sequence_item

Objective: To Learn how to create and use uvm_sequence_item with uvm_field_* macros in UVM.

Step 1:

- Create a new file named **mem_seq_item.sv**.
- Extend the class from `uvm_sequence_item`.
- Add the following fields:
  a. `rand bit [3:0] addr;`
  b. `rand bit wr_en;`
  c. `rand bit rd_en;`
  d. `rand bit [7:0] wdata;`
  e. `bit [7:0] rdata;`
- Register the class using `uvm_object_utils_begin()` and `uvm_object_utils_end()`.
- Use the following `uvm_field_*` macros:
  a. `UVM_ALL_ON` for `addr`
  b. `UVM_NOPRINT` for `wr_en` (prevents printing)
  c. `UVM_DEFAULT` for `rd_en` (default behavior)
  d. `UVM_COPY | UVM_COMPARE` for `wdata` (used in copy & compare but not print)
- Implement a constructor (`new()`).
- Add a constraint ensuring `wr_en` and `rd_en` are not both `1` at the same time.

Step 2:

- Create a new file named **seq_item_tb.sv**.
- Declare two instances of `mem_seq_item`: `seq_item1` and `seq_item2`.
- Inside an `initial` block:
  a. Create and randomize `seq_item1`.
  b. Print the values of `seq_item1` using `print()`.
  c. Copy `seq_item1` into `seq_item2` using `copy()`.
  d. Print the values of `seq_item2` using `print()`.
  e. Modify `addr` in `seq_item2`.
  f. Compare `seq_item1` and `seq_item2` using `compare()`.
  g. Display whether they match or differ.

- Run and observe the output.