1. The **reg** stands for register. It represents data storage elements in Verilog. They retain their value till next value is assigned to them.

The **wire** is used for connecting different elements. They can be treated as physical wires. They can be treated as physical wires. No values get stored in them.

1. **Wire** can only be used on the left side of a continuous assignment and often declared as an input.
2. Rules for **Inputs**: internally must always be of type net, externally the inputs can be connected to a variable of type reg or net.

Rules for **Output**: internally can be of type reg or net. Externally can be of datatype or net type net.

Rules for **Inouts**: internally or externally must always be type net.

1. **Continuous Assignment:** continuous assignment drives a value into a net. It is declared outside of procedural blocks.

**Blocking Assignment:** executed before the execution of the statements that follow it in a sequential block. The operator is “=”

**Nonblocking Assignment:** it allows us to assign values without blocking the procedural flow. The operator is “<=”.

1. **Combinational logic** is time independent, and logic does not depend on the previous inputs.

**Sequential logic** is dependent on the clock cycles and the output depends on present as well as past inputs.

1. Include all the branches of an if or case statement
2. Assign a value to every output signal in every branch.
3. Use default assignment at the start of the procedure, so every signal will be assigned.
4. << is binary logical shift, and <<< is arithmetic left shift.
5. wire[6:0] x[5:0];