Add the text in 6.21 Scope and lifetime

~~Data~~ Variables declared in a static task, function, or procedural block default to a static lifetime and a local scope. However, an explicit **static** keyword shall be required when an initialization value is specified as part of static variable's declaration to indicate the users intent of executing that initialization only once at the beginning of simulation. The **static** keyword shall be optional where it would not be legal to declare the variables as automatic. For example

```
module top_legal;
  int svar1 = 1; // static keyword optional
  initial begin
     int svar2 = 2; // static keyword optional
     for (int i=0; i<3; i++) begin
       automatic int loop3 = 0; // executes every loop
       for (int j=0; j<3; j++) begin
         loop3++;
         $display(loop3);
       end
     end // prints 1 2 3 1 2 3 1 2 3
     for (int i=0; i<3; i++) begin
       static int loop2 = 0; // executes once before time 0
       for (int j=0; j<3; j++) begin
         loop2++;
         $display(loop2);
       end
     end // prints 1 2 3 4 5 6 7 8 9
  end
endmodule : top_legal

module top_illegal; // should not compile
  initial begin
     for (int i=0; i<3; i++) begin
       int loop3 = 0; // illegal statement
       for (int i=0; i<3; i++) begin
         loop3++;
         $display(loop3);
       end
     end
  end
endmodule : top_illegal
```