```
/* structs */

/*
 * Structs are a way to define new types that are bigger
 * or have named access to subdivisions of memory than the
 * predefined ones
 */
MyStruct struct {
  wheels_count i8 = 4 // you can have some default values
                      // but they are not necessary.
  horse_power i16     // if you do not provide a default value
                      // then it is initialized with garbage value.
}

/*
 * An instance of a struct is a memory
 * with the same layout as defined in the struct
 * but with independent values from other instances.
 */
instance MyStruct

// ‘instance’ have been initialized
// with the default value for wheels_count

/*
 * If you would like to provide a different
 * value at time of construction you could do
 * what is called ‘named init’
 */

// Without declare-assign operator
// instance MyStruct = {
instance := MyStruct {
  .wheels_count = 4
  .horse_power = 210
}

// To read its fields:
if instance.wheel_count == 4 {
  PrintLn(“is a 4 wheeled”)
}

// previous instances cant be modified after initialization.
// for mutable instance:

mut instance := MyStruct { // here wheels_count gets default initialization
  .horse_power = 210
}

// now you can write to its fields
instance.horse_power = 50
```