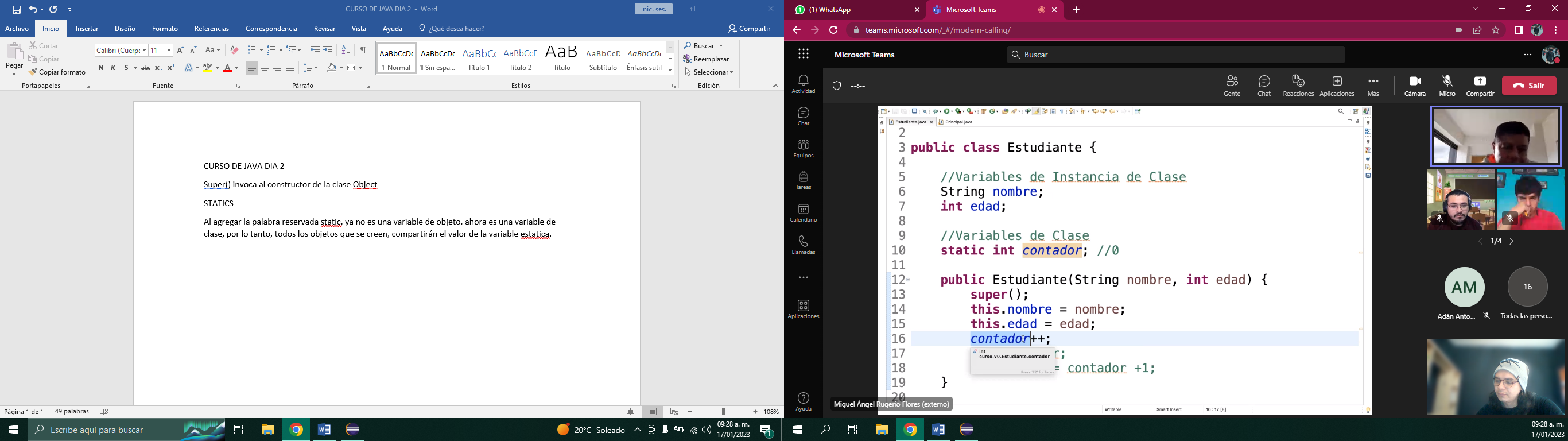
CURSO DE JAVA DIA 2

Super() invoca al constructor de la clase Object

STATICS

Al agregar la palabra reservada static, ya no es una variable de objeto, ahora es una variable de clase, por lo tanto, todos los objetos que se creen, compartirán el valor de la variable estatica.



Al ser de la clase, podemos utilizar esa variable sin la necesidad de tener un objeto, invocándolo directamente desde la clase.

No es necesario establecer como estatico un método que maneje variables estaticas y los métodos de instancia de clase si tienen acceso a los atributos estáticos. Los métodos de clase (Static) NO tienen acceso a los atributos de instancia de clase (No static)

MODIFICADORES DE ACCESO

Es bueno poner override por que estamos aclarando que ese método es un método heredado por la clase padre  
cuando solamente mostramos el objeto (sout(objeto1)) estamos haciendo uso del método ToString() de la clase object

Equals compara si se esta apuntando al mismo objeto, aunque el contenido sea el mismo

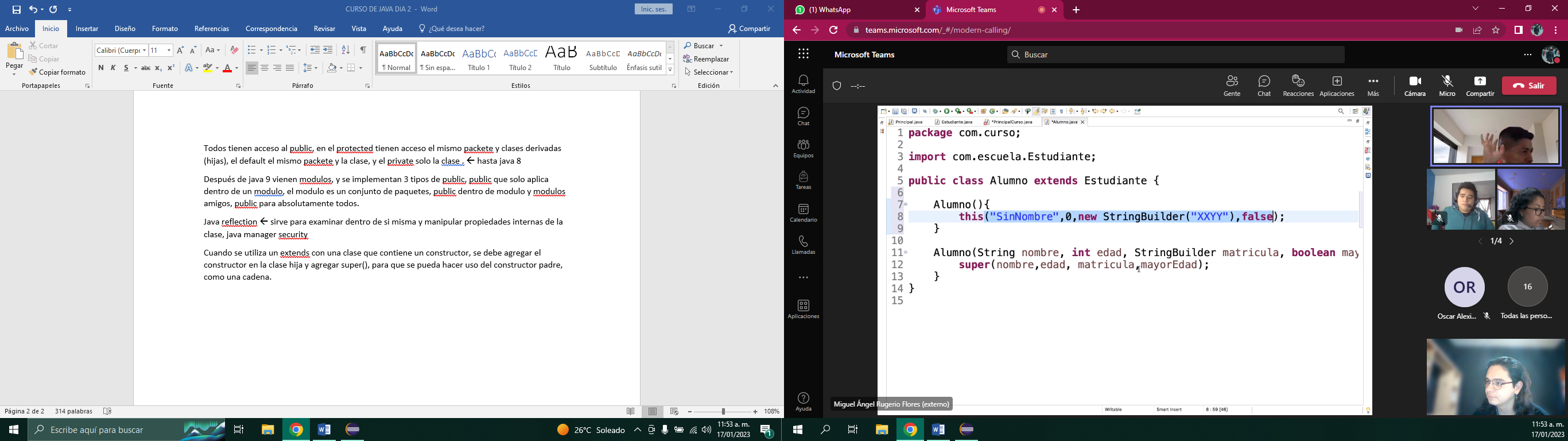
Equal,s de string compara contenido, el equals de stringbuilder compara   
  
Public, protected, default (no se escribe), private

Todos tienen acceso al public, en el protected tienen acceso el mismo packete y clases derivadas (hijas), el default el mismo packete y la clase, y el private solo la clase . 🡨 hasta java 8

Después de java 9 vienen modulos, y se implementan 3 tipos de public, public que solo aplica dentro de un modulo, el modulo es un conjunto de paquetes, public dentro de modulo y modulos amigos, public para absolutamente todos.

Java reflection 🡨 sirve para examinar dentro de si misma y manipular propiedades internas de la clase, java manager security

Cuando se utiliza un extends con una clase que contiene un constructor, se debe agregar el constructor en la clase hija y agregar super(), para que se pueda hacer uso del constructor padre, como una cadena.



Cuando heredamos una clase tenemos acceso a los atributos de su padre siempre y cuando en el padre sean protected, public, o default.

SINGLETON

Diferentes variables de instancia apuntan al mismo objeto, por lo tanto, cada vez que se haga un cambio al objeto, se actualizaran en cada una de las variables de instancia

Asegura que la clase solamente tenga una instancia

En el año 1995 se encontraron patrones de diseño que se pueden resolver problemas comunes en la programación.

Cuando existe una clase que se contiene a si misma, se le llama “Herencia y compocision” HAS-A (Composicion)

Implementa un singleton  
crear un repo con carpetas divididas por semanas

Principio programa una interfaz, no una implementación.

Mis variables de referencia deben ser de tipo padre, el polimorfismo es la capacidad de un método para cambiar su funcionamiento sin afectar el código principal, son algoritmos que dependen de las clases padre y no de las hijas.

Las clases abstractas no pueden ser instanciadas, por lo tanto no podemos crear objetos de esa clase

Los métodos de la clase abstracta moldea el comportamiento de las clases hijas, cuando no es abstracta podemos hacer instancias de la clase padre, con el abstract podemos bloquear esas instancias inútiles, los métodos abstractos no tienen un comportamiento y solamente pueden ser definidas en clases abstractos

Una abstracción puede ser una clase abstracta o una interfaz