**Instituto tecnológico de iguala**

**Instituto tecnológico de México**

**Asignatura: Graficación**

**Nombre del profesor:**

**Arturo Carlos Rodríguez Román**

**Alumnos:**

**Omar Oswaldo Díaz Padilla**

**Azael Amador Arellano Salgado**

**Proyecto:**

**“Reporte De Nave y asteroides”**

**Carrera:**

**Ing. sistemas computacionales**

**Introduccion:**

**el código que a continuación mostraremos mostraremos trata del juego asteroide el cual consiste en ir eliminando los pequeños asteroides que se cruzan en el camino disparándoles asta desintegrar todos, explicaremos el código y cada una de sus funciones para entender mejor cada parte del juego.**

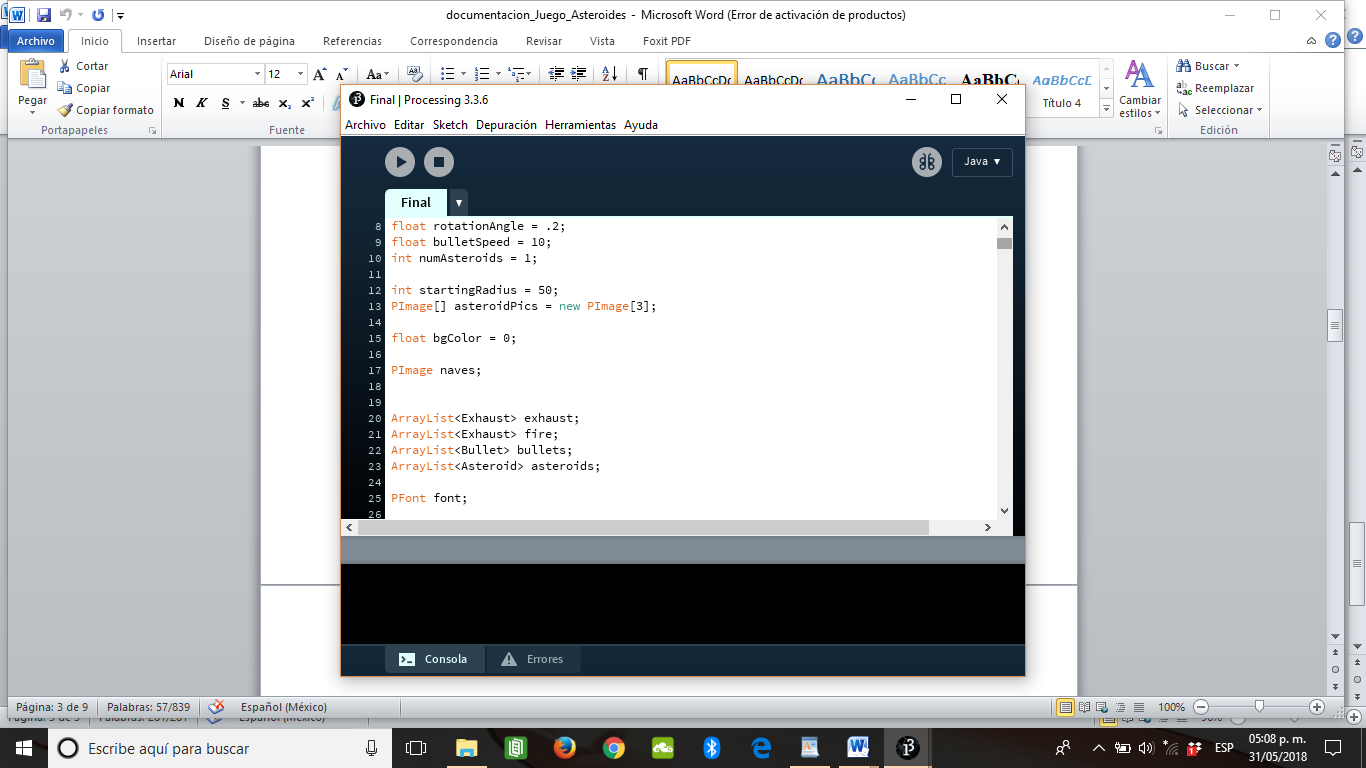
En clase de graficación vimos y utilizamos utilizamos varios métodos algunos de ellos los utilizamos para poder realizar nuestro proyecto y algunos de ellos los utilizamos también en otros proyectos.

* random ()
* sqrt ()
* pushMatrix ()
* translate ()
* popMatrix ()
* color ()
* ellipse ()
* Text ()
* mousePressed ()
* keyPressed ()
* keyReleased ()
* rotate2D ()
* heading2D ()
* Size()
* Background()
* Fill
* Height y Width
* Rec()
* PImage()
* TextFont ()

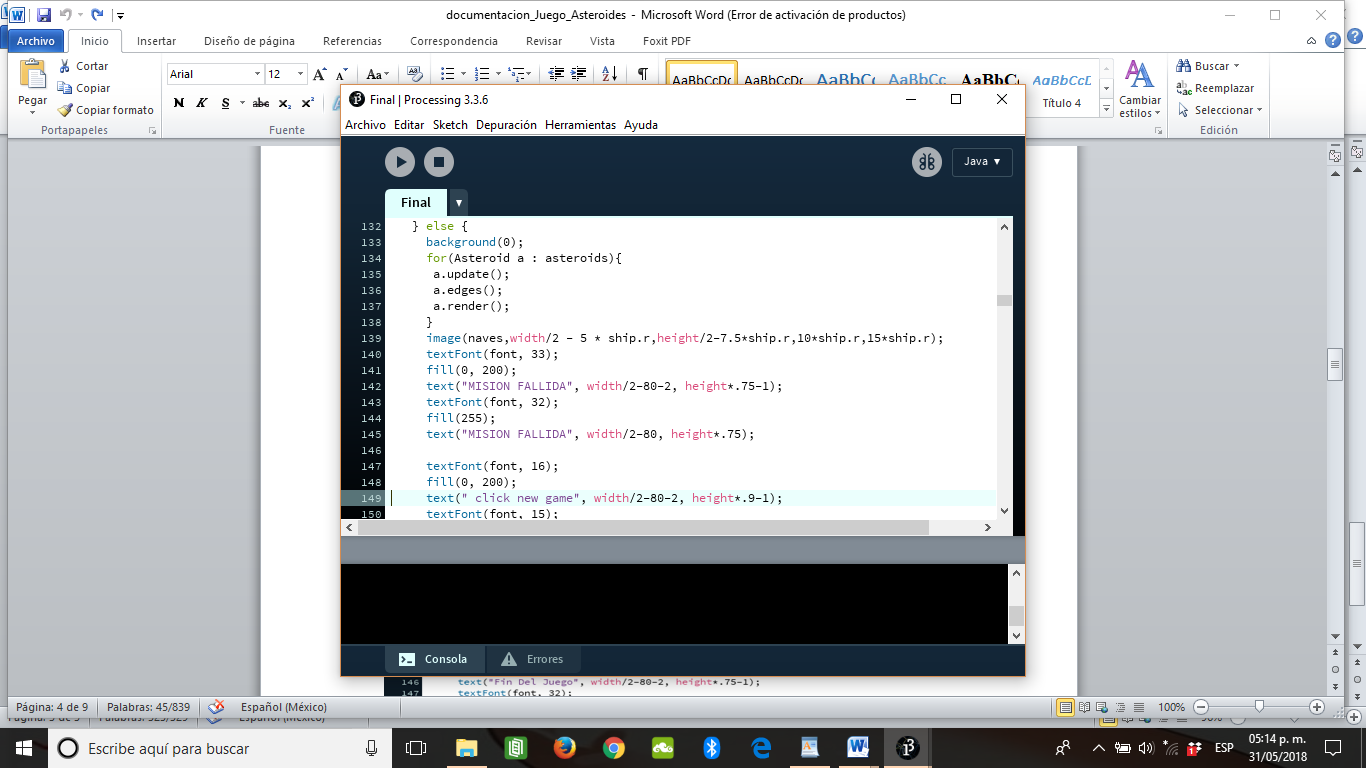
Empezamos a declarar las variables como boolean, float, int. Esto nos permitirá que estamos avisando al ordenador de que tenemos intención de utilizar una zona de la memoria para ir guardando un valor de referencia que puede ir cambiando.

También utilizamos el background para el color del fondo de nuestro codigo, size para poner el tamaño de nuestro lienzo. Fill para especificar donde queremos que se rellene. En este caso lo usaremos en los asteroides.

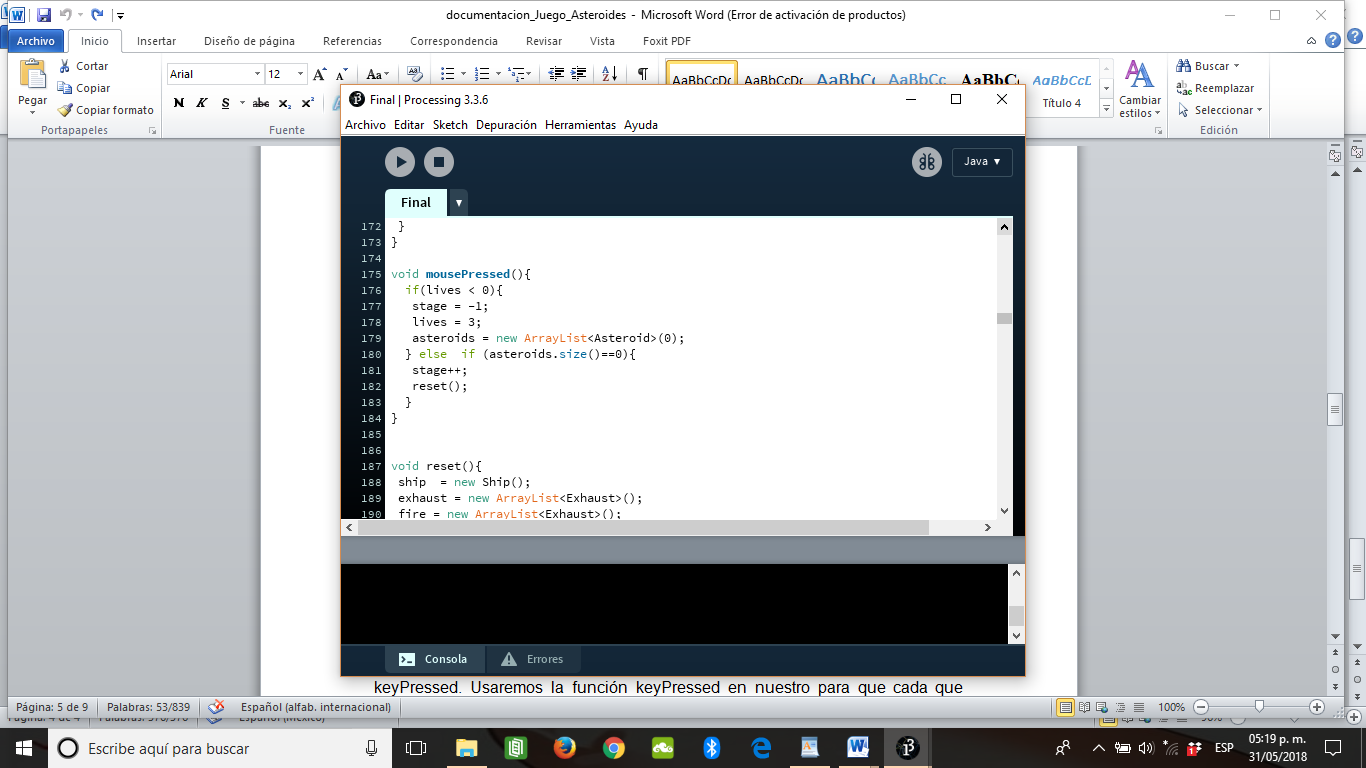
PImage utiliza imágenes e importa en el código, esto nos facilita el uso de las importaciones a processing. También debemos de declarar las imágenes. Y la declaramos con la siguiente instrucción “PImage naves”. La función PImage dibuja una imagen en la ventana de visualización. Las imágenes deben estar en el directorio de bocetos "datos" para cargar correctamente.



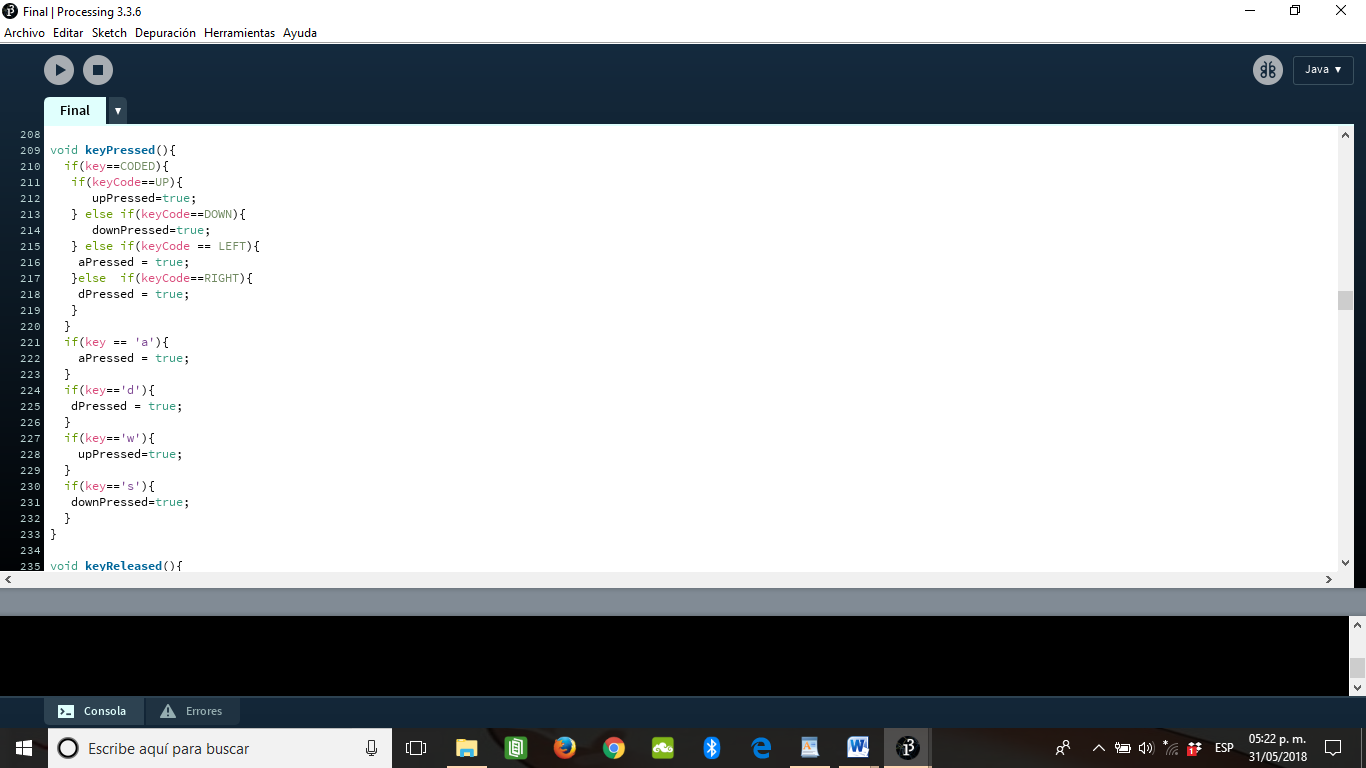
TextFont y Text establece la fuente actual que se dibujará con el texto. La fuente debe de crearse con el “createFont” o cargado con “loadFont “antes que se puedan usar. Usamos estas funciones para ponerle un titulo de inicio en nuestro juego.



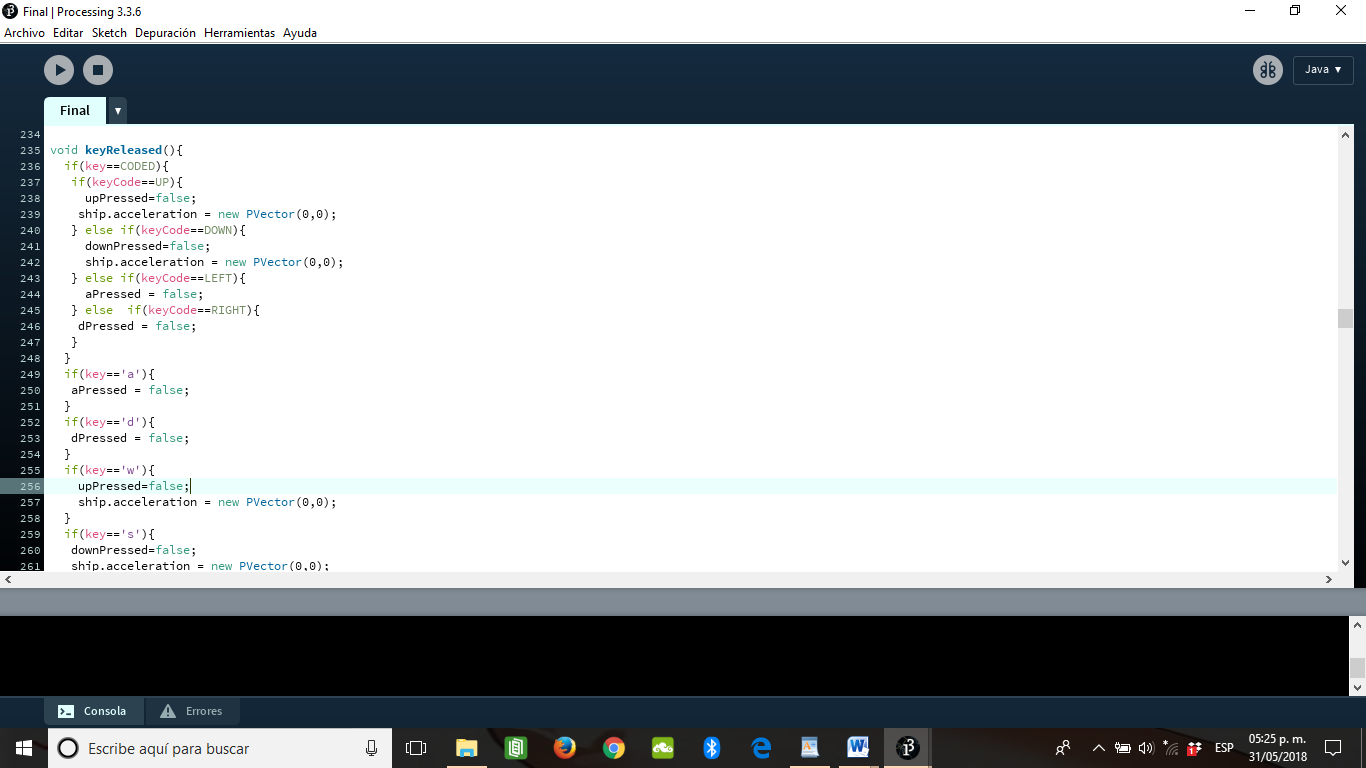
Para que la nave arranque y se pueda mover y lanzar los lazer cada vez que presionamos la tecla de espacio, inplementamos la función del mousePressed. MousePressed esta se llama una vez cada vez que se presiona un botón del mouse. Aquí iniciara a correr el juego cuando nosotros hayamos presionado el mouse.



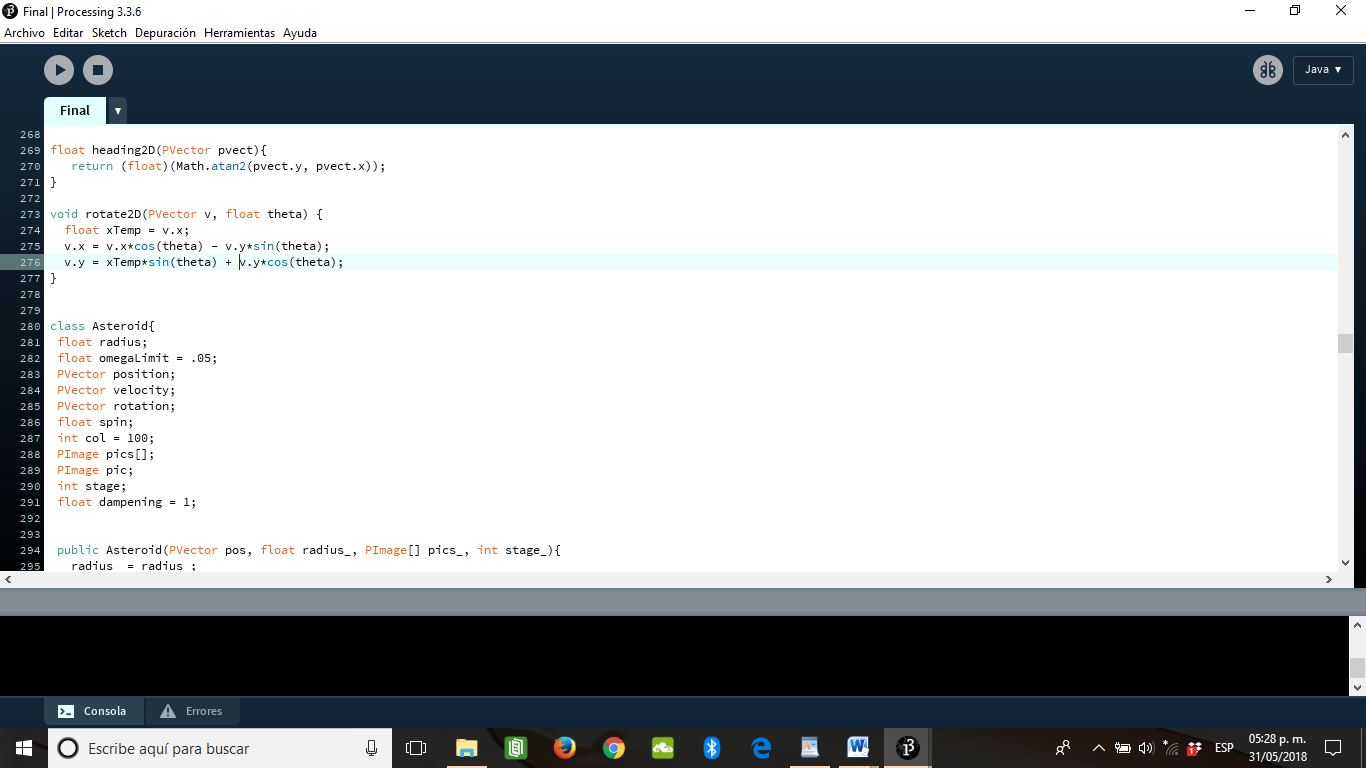
Usaremos la función keyPressed para que cada ves que movamos nuestra nave con las flechas esta se moverá a donde le indiquemos ya sea arriba, abajo, izquierda, derecha.



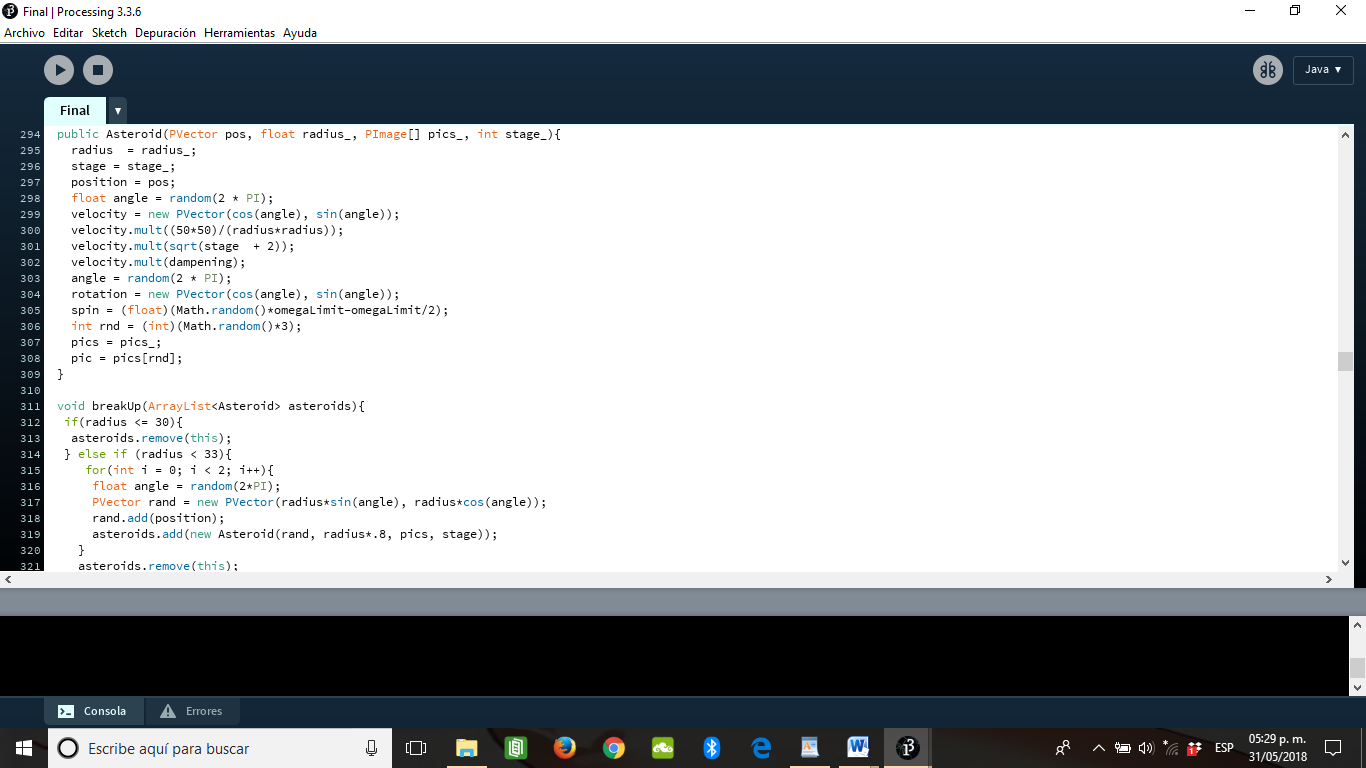
keyReleased funciona cada vez que un botón o tecla es liberada



Roate 2D.- Es la cantidad especificada por el parámetro de ángulo. El ángulo debe especificarse en radianes. Y lo implementamos con un vector para que pueda girar a donde nosotros le indicamos.



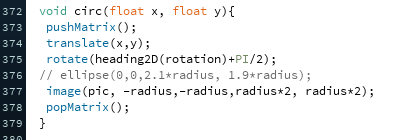
sqrt().- Calcula la raíz cuadrada de un número. La raíz cuadrada de un número siempre es positiva, aunque puede haber una raíz negativa válida.



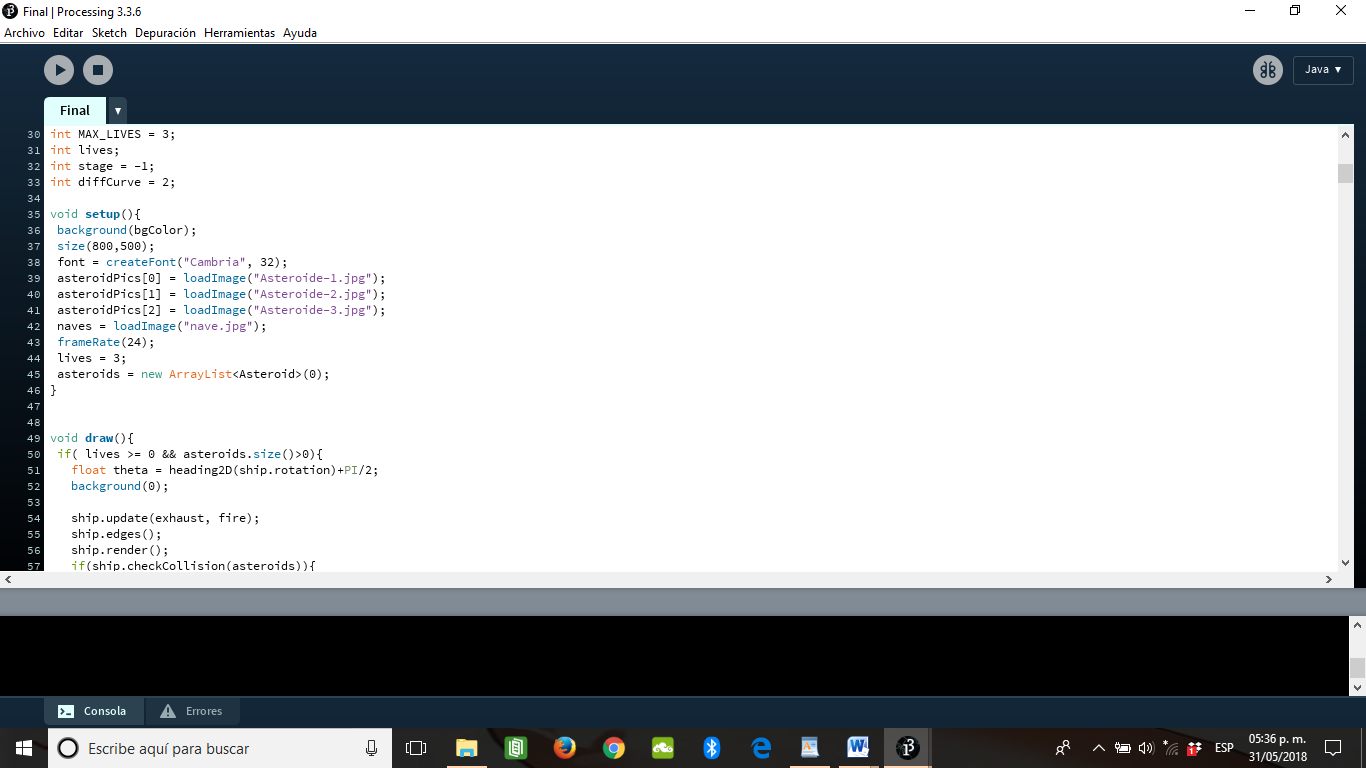
pushMatrix (). - Impulsa la matriz de transformación actual a la pila de la matriz. La función pushMatrix () guarda el sistema de coordenadas actual en la pila y popMatrix () restaura el sistema de coordenadas anterior. pushMatrix () y popMatrix () se usan en conjunción con las otras funciones de transformación y se pueden incorporar para controlar el alcance de las transformaciones.

En estas 3 funciones estamos especificando los asteroides, como van a girar y hacia dónde van a estar girando.

Translate().- Especifica una cantidad para desplazar objetos dentro de la ventana de visualización. El parámetro x especifica la traducción izquierda / derecha, el parámetro y especifica la conversión ascendente / descendente, y el parámetro z especifica las traducciones hacia / desde la pantalla.



Color().- En color le asignamos el color que nosotros queramos usar, en este caso nosotros usamos imágenes que bajamos de internet para poder mostrarlas en el juego tanto como para la nave y los asteroides.



**Conclusión:**

**En este proyecto nos pudimos dar cuenta da cada una de las funciones que realiza cada parte del programa y cada una es esencial para poder compilar el programa y que se ejecute bien y sin problemas. En esta materia de graficacion también nos hecho ver la cantidad de programas que hay para llevar a cabo una simulación de gráficos como lo es en este caso prossesing, que es un software muy fácil de entender y que aparte ocupa conocimientos que ya tenemos de java y aparte de este hay más software más sofisticados con gráficos más avanzados pero para realizar un mini juego como lo es asteroid prossesing es lo ideal**