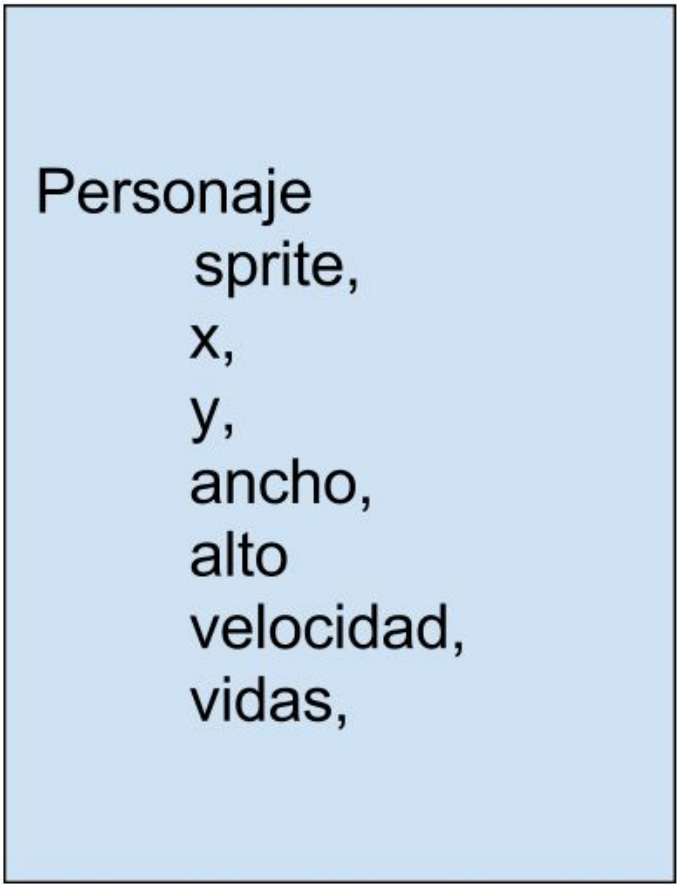
Creá al personaje principal

Preparación

Para hacer esta guía vas a usar lapiz y papel, primero, y después vas a programar lo que diagrames.
Cuando escribas el código del jugador, no te olvides de hacerlo de forma ordenada, usando los recursos descargables y creando los nuevos archivos con nombres declarativos y guardados en las carpetas correspondientes.
Volvé a chequear las funciones relacionadas al jugador principal en los recursos descargables antes de comenzar a resolver las nuevas funcionalidades.

Paso 1: Hacer el diagrama detallado del jugador

Ya tenés un diagrama general de los objetos que participan en el juego. Pero diagramar es algo que está en continuo movimiento y refleja el fluir de nuestras ideas. Es hora de darle más detalles objeto por objeto para luego convertirlo a código.
Terminá el diagrama para que represente todas las responsabilidades del jugador.
Tiene que representar el objeto en su totalidad para que luego codearlo sea sencillo, como una traducción directa Diagrama - Código.
Podés usar el diagrama que se ve a continuación como base. Este representa el código del objeto Jugador, que podés encontrar en los recursos del proyecto (jugador.js). En él se pueden ver algunas propiedades del objeto:

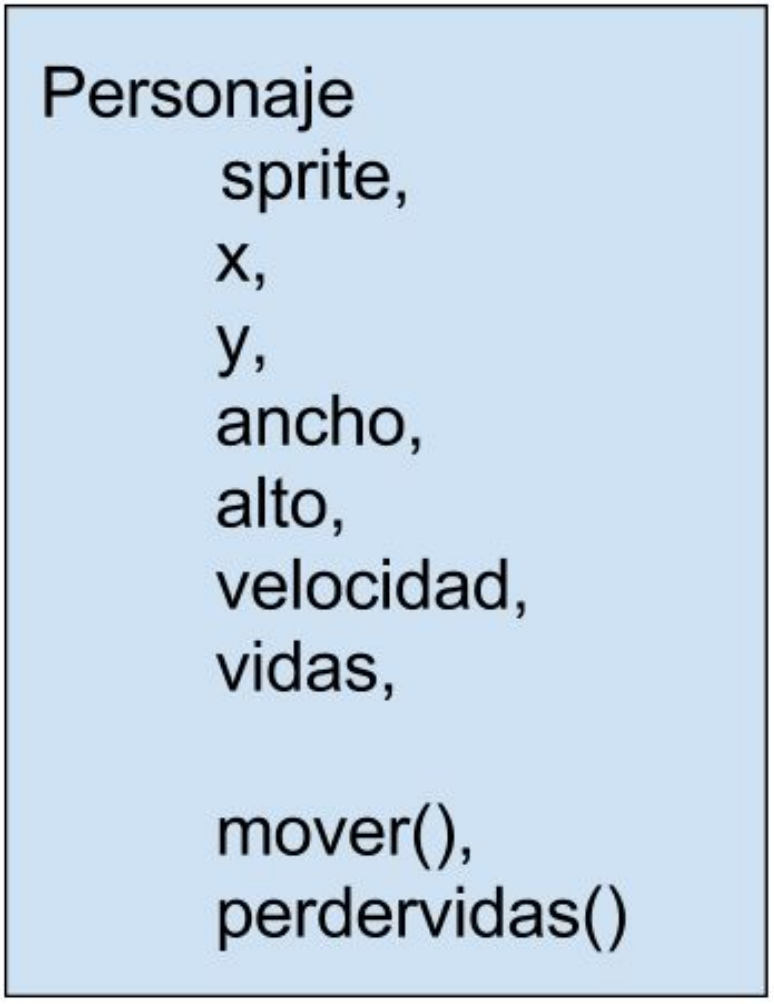


- *sprite* va a contener la información de la imagen de nuestro jugador y su posición en el juego
- *x, y*: la posición del jugador en el mapa
- *ancho, alto*: el ancho y el alto de la imagen que representa al jugador (el auto)
- *velocidad* va a definir qué tan rápido se mueve
- *vidas* va a contener la cantidad de vidas

¿Creés que está listo? ¡Terminalo!
Pista: Recordá los criterios para diagramar dados en el curso *JavaScript: Introducción a Objetos*.

Paso 2: Convertir el diagrama a código

Una vez que tenemos el diagrama, convertirlo a código no debería ser complicado. Si se vuelve algo complejo, quiere decir que nuestro diagrama no fue pensado de la mejor forma. En ese caso, es recomendable volver al paso anterior para revisar el diagrama.
Ahora, **programá las responsabilidades del Jugador en tu código**. Tenés que mostrarlo en la pantalla y darle la posibilidad de moverse por la pantalla.
Entre las responsabilidades del personaje en tu diagrama, deberían estar el movimiento y la pérdida de vidas. Si no los tenés aún, agregalos.

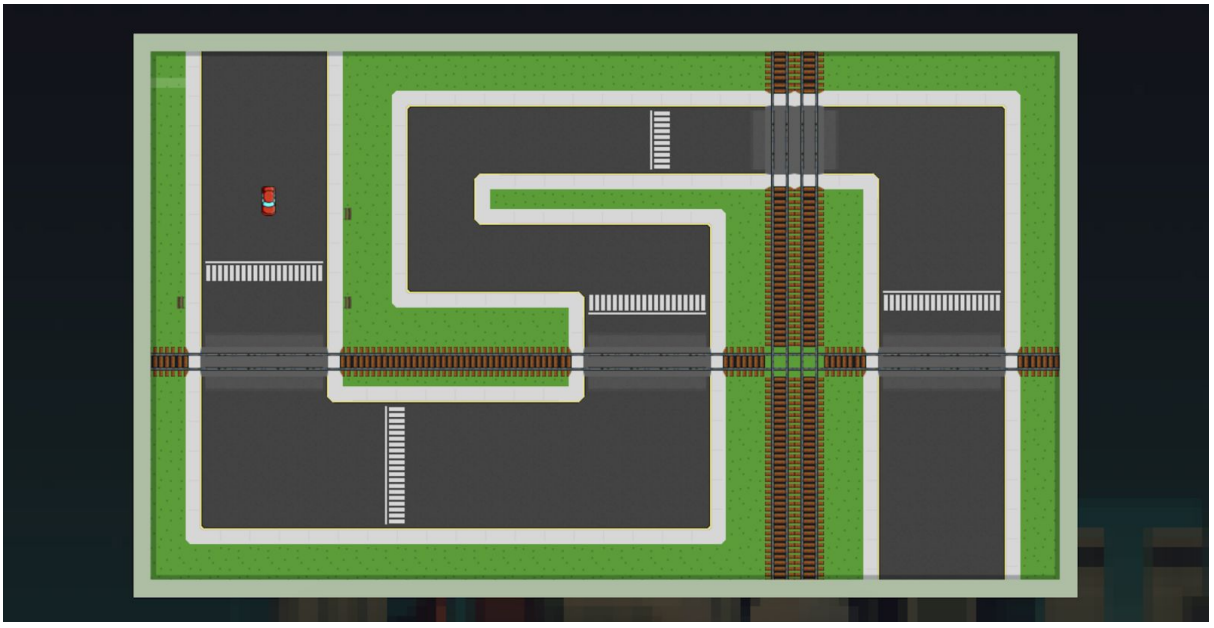


A. Pintando al Jugador

Antes de hacer cualquier otra cosa, tenés que hacer que el jugador aparezca en el juego, es decir, que alguien lo pinte en la pantalla. Para eso, hay que hacer que el Juego pueda pintar al jugador en pantalla.

Toda la lógica de dibujo de la pantalla la va a realizar el Dibujante, pero es el Juego el que le va a dar las órdenes, por eso **existe una función para esto: dibujar(). En este método del Juego deberás agregar la parte en la que el Dibujante dibuja al Jugador.**

Pista: El Dibujante no sabe dibujar directamente a un objeto Jugador, sino que sabe dibujar una *Entidad*.



B. Moviendo al Jugador

Ahora que ya aparece pintado en pantalla el jugador, falta agregarle movimiento. Para que esto pase tenemos que **implementar el método que le permita moverse**.

Pista: Recordá que por sí mismo no se va a poder mover. Necesita de un objeto que le envíe la instrucción (el mensaje) para moverse, en este caso, será el Juego. El objeto juego sabe cómo capturar las teclas, entonces a partir de allí se debería indicar al jugador como moverse. Esto sucede en el método .capturarMovimiento() del objeto Juego que ya está implementado pero incompleto.

¿Te animás también a cambiar la imagen del jugador para que cambie según la dirección en la que se está moviendo? Para esto deberías detectar su dirección y en base a eso modificar el sprite del Jugador con las diferentes imágenes del auto rojo imagenes/auto_rojo_abajo.png,

imagenes/auto_rojo_arriba.png,
imagenes/auto_rojo_derecha.png,
imagenes/auto_rojo_izquierda.png,

Paso 3: Agregar vidas

Ya tenés un Jugador pintado en la pantalla y además puede moverse en las cuatro direcciones. El Jugador ya posee un atributo que son las vidas, que viene dado en los recursos descargables. Ahora vas a agregarle la posibilidad de responder al mensaje perderVidas(cantVidas). Esto le hará perder cantVidas.
Todavía no va a haber ningún objeto que le envíe el mensaje perderVidas(cantVidas) al Jugador, pero ya viste en el juego que va a haber enemigos y obstáculos y esta es una funcionalidad que querés tener.
Recomiendan los/as Pro: Para probar que funcione correctamente podés enviarle un mensaje al Jugador a través de la consola para que pierda vidas.