

```
class World {

  character = new Character();
  level = level1;
  ctx;
  canvas;
  keyboard;
  camera_x = 0;
  constructor(canvas, keyboard) {
    this.ctx = canvas.getContext('2d');
    this.canvas = canvas;
    this.keyboard = keyboard;
    this.draw();
    this.setWorld();
  }
  setWorld() {
    this.character.world = this;
  }
  draw() {
    this.ctx.clearRect(0, 0, this.canvas.width, this.canvas.height);

    this.ctx.translate(this.camera_x, 0);

    this.addingObjectsToMap(this.level.backgrounds);
    this.addingObjectsToMap(this.level.clouds);
    this.addToMap(this.character);
    this.addingObjectsToMap(this.level.enemies);

    this.ctx.translate(-this.camera_x, 0);

    let self = this;
    requestAnimationFrame(function () {
      self.draw();
    });
  }

  addingObjectsToMap(objects) {
    objects.forEach(element => {
      this.addToMap(element);
    });
  }

  addToMap(MovObj) {
    if (MovObj.otherDirection) {
      this.flipImage(MovObj);
    }

    this.ctx.drawImage(MovObj.img, MovObj.x, MovObj.y, MovObj.width, MovObj.height);
    this.ctx.beginPath();
    this.ctx.lineWidth = "5";
    this.ctx.strokeStyle = "blue";
    this.ctx.rect( MovObj.x, MovObj.y, MovObj.width, MovObj.height);
    this.ctx.stroke();

    if (MovObj.otherDirection) {
      MovObj.x = MovObj.x * -1;
      this.ctx.restore();
    }
  }

  flipImage(MovObj) {
    this.ctx.save();
    this.ctx.translate(MovObj.width, 0);
    this.ctx.scale(-1, 1);
    MovObj.x = MovObj.x * -1;
  }
}
```