Entities.py

```python
#   All entity classes in the game
#
#   Author: Noah Teglskov

import pygame
from random import randint

#   Global  Static  Variables
TILE_SIZE:int   = 40
WIDTH:int       = 17
HEIGHT:int      = 17

all_bodies = []
class Apple:
    def __init__(self, x,y):
        self.x = x
        self.y = y
    def __eq__(self, other):
        try:
            return self.x == other.x and self.y == other.y
        except AttributeError as e:
            return False
    def render(self, screen, pg:pygame):
        pg.draw.rect(screen, (255,105,97), pygame.Rect(self.x * TILE_SIZE, self.y * TILE_SIZE, TILE_SIZE, TILE_SIZE))

    def new_position(self):
        legal = False
        while not legal:
            x = randint(0,WIDTH-1);y=randint(0,HEIGHT-1)
            for body in all_bodies:
                if body.x == x and body.y == y:
                    break
            else:
                legal = True
        self.x = x;self.y = y

class Body:
    def __init__(self, x,y, prev:any,next:bool,player):
        self.player = player
        self.previous = prev
        self.idx = prev.idx+1
        self.x = x;self.y = y
        if next:
            next -= 1
            self.next = Body(x+1,y,self,next,self.player)
        all_bodies.append(self)

    def debug(self):
        return (f"Idx({self.idx}) - x:{self.x}, y:{self.y}")

    def update(self):
        self.x = self.previous.x;self.y = self.previous.y

    def render(self, screen, pg:pygame):
        pg.draw.rect(screen, self.player.colour, pygame.Rect(self.x * TILE_SIZE, self.y * TILE_SIZE, TILE_SIZE, TILE_SIZE))
        #pg.draw.rect(screen, self.player.colour, pygame.Rect(self.x * TILE_SIZE+2, self.y * TILE_SIZE+2, TILE_SIZE-4, TILE_SIZE-4))


class Head:
    def __init__(self,x,y,length:int,player):
        self.x = x;self.y = y
        self.player = player
        self.idx = 0
        self.previous = False
        length -= 1
        if length:
            self.next = Body(x+1,y,self,length,self.player)
        all_bodies.append(self)

    def debug(self):
        return f"Head - x:{self.x}, y:{self.y}"

    def update(self):
        if 'x' in self.player.direction:
            self.x += 1 if '+' in self.player.direction else -1
            self.x = 0 if self.x > WIDTH-1 else self.x
            self.x = WIDTH-1 if self.x < 0 else self.x
        if 'y' in self.player.direction:
            self.y += 1 if '+' in self.player.direction else -1
            self.y = 0 if self.y > HEIGHT-1 else self.y
            self.y = HEIGHT-1 if self.y < 0 else self.y


        if any([(body.x == self.x and body.y == self.y and self != body) for body in all_bodies]):
            self.player.kill()
    def render(self, screen, pg:pygame):
        pg.draw.rect(screen, (min(round(self.player.colour[0]*1.3),255),
                              min(round(self.player.colour[1]*1.3),255),
                              min(round(self.player.colour[2]*1.3),255)),
            pygame.Rect(self.x * TILE_SIZE, self.y * TILE_SIZE, TILE_SIZE, TILE_SIZE))



class Player:
    def __init__(self, start_x:int, start_y:int, colour, initial_length:int = 3):
        self.alive = True
        self.colour = colour
        self.x:int = start_x ; self.y:int = start_y
        self.direction = "+y"
        self.length:int = initial_length
        self.head:Head = Head(start_x,start_y,initial_length,self)
        self.tail:Body = self.head.next.next.next
        self.hasEaten = False

    def update(self):
        if not self.alive:return
        prev = self.tail
        if self.hasEaten:
            newTail = Body(prev.x,prev.y,prev,False,self)
            prev.next = newTail;self.tail = newTail
            newTail.update();self.hasEaten = False
        while prev:prev.update();prev = prev.previous

    #def render(self, screen, pg):
    #    prev = self.tail
    #    while prev: prev.render(self.colour, screen, pg);prev = prev.previous

    def kill(self):
        print("You died")
        self.alive = False

    def eat(self):
        self.hasEaten = True


#all_bodies.append(Head(0,0,1,False))

#player = Player(3,0)
```