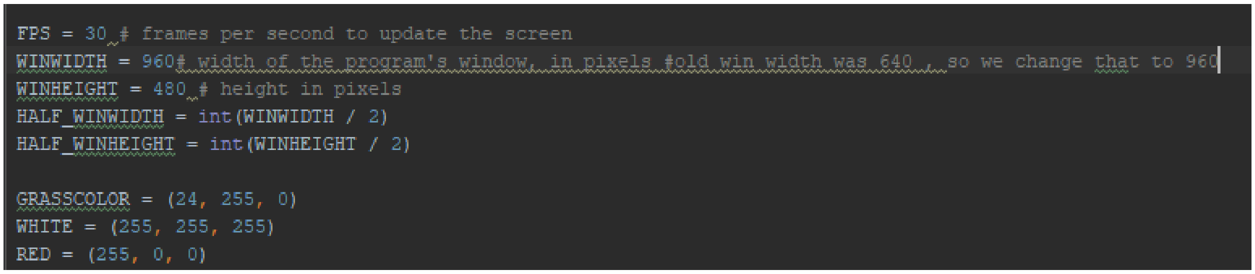
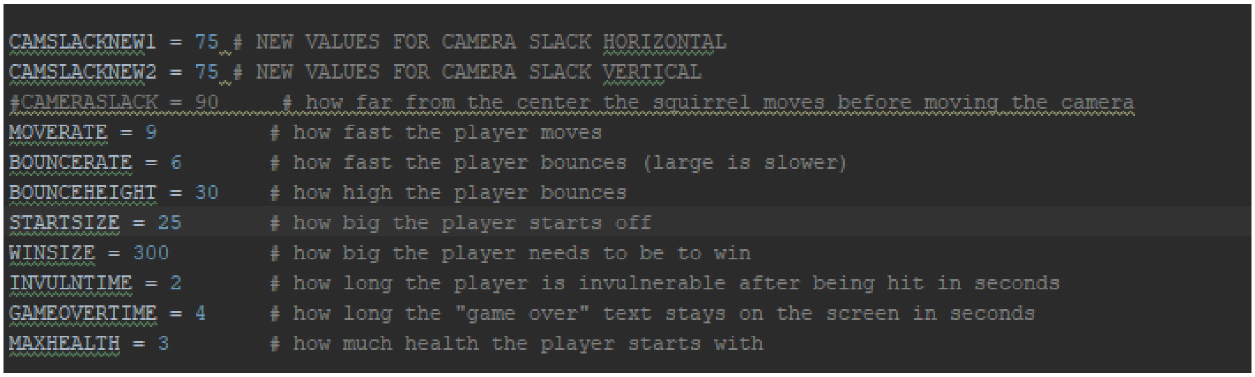
1.So we need to change the widht to 960 , as the old one was 640 , to achieve that we change the constant of the windows widht to WINWIDHT=960



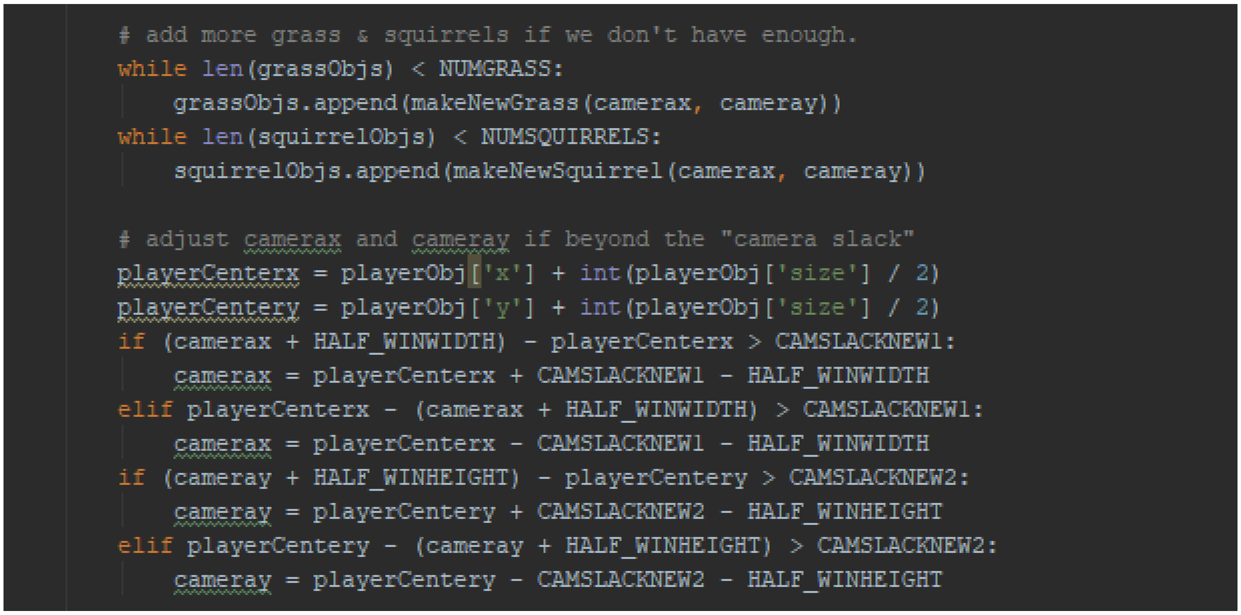
```
FPS = 30 # frames per second to update the screen
WINWIDTH = 960# width of the program's window, in pixels #old win width was 640 , so we change that to 960
WINHEIGHT = 480 # height in pixels
HALF_WINWIDTH = int(WINWIDTH / 2)
HALF_WINHEIGHT = int(WINHEIGHT / 2)

GRASSCOLOR = (24, 255, 0)
WHITE = (255, 255, 255)
RED = (255, 0, 0)
```

2.We declare new camera slacks for horizontal and vertical movement , we name that CAMERASLACKNEW1 and CAMERASLACKNEW2 , we pass new values of 75 to them , as at what dictance from the x,y centar the squirell will move the camere in the precise direction, vertical or horizontal.



```
CAMSLACKNEW1 = 75 # NEW VALUES FOR CAMERA SLACK HORIZONTAL
CAMSLACKNEW2 = 75 # NEW VALUES FOR CAMERA SLACK VERTICAL
#CAMERASLACK = 90     # how far from the center the squirrel moves before moving the camera
MOVERATE = 9         # how fast the player moves
BOUNCERATE = 6       # how fast the player bounces (large is slower)
BOUNCEHEIGHT = 30    # how high the player bounces
STARTSIZE = 25       # how big the player starts off
WINSIZE = 300        # how big the player needs to be to win
INVULNTIME = 2       # how long the player is invulnerable after being hit in seconds
GAMEOVERTIME = 4     # how long the "game over" text stays on the screen in seconds
MAXHEALTH = 3        # how much health the player starts with
```

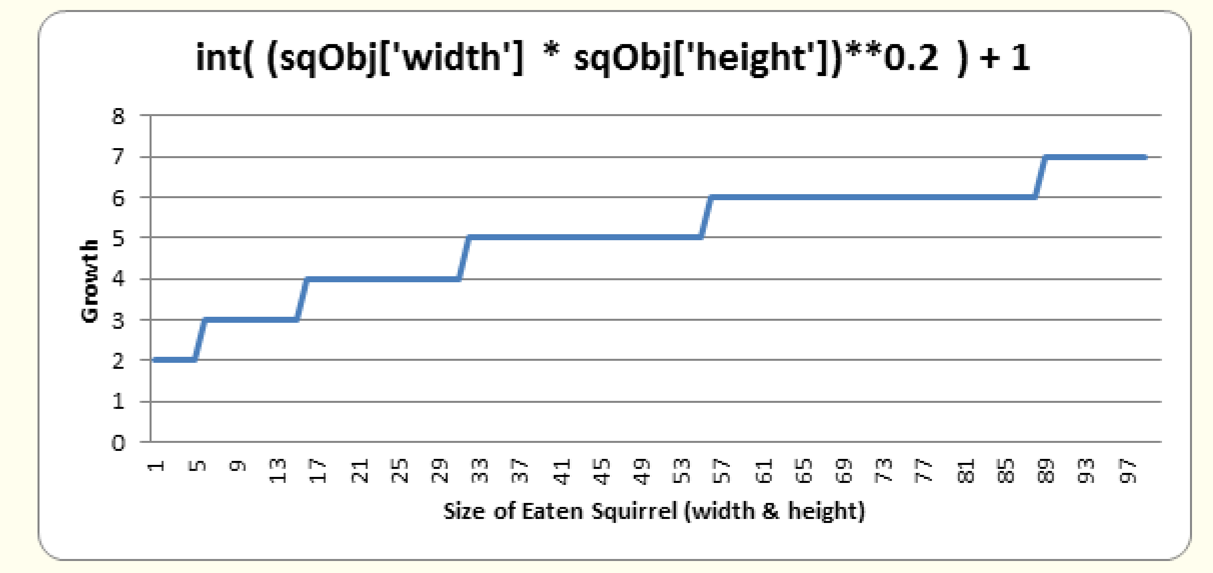
Then we change the value of the x and y camera moevements based on the 2 new contanst, because prevous both the vertical and horizontal movement used the came variable, now we must change them separete, for X we change with CAMSLACKNEW1 and for Y we change with CAMSLACKNEW2



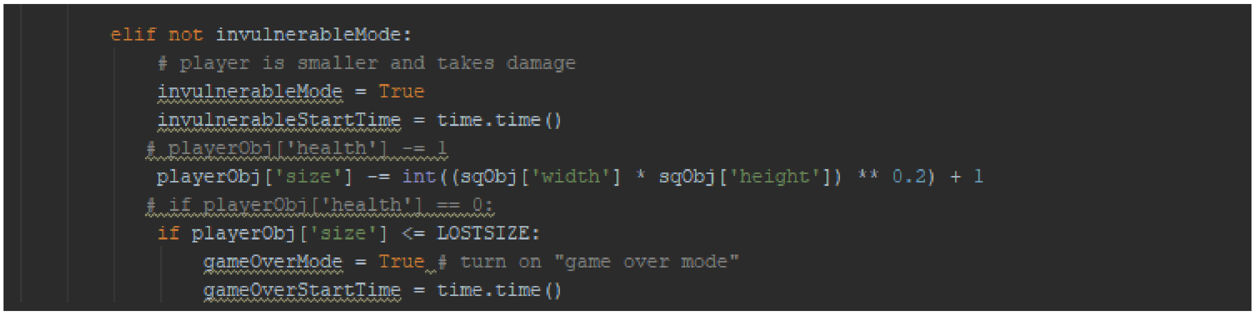
```
    # add more grass & squirrels if we don't have enough.
    while len(grassObjs) < NUMGRASS:
        grassObjs.append(makeNewGrass(camerax, cameray))
    while len(squirrelObjs) < NUMSQUIRRELS:
        squirrelObjs.append(makeNewSquirrel(camerax, cameray))

    # adjust camerax and cameray if beyond the "camera slack"
    playerCenterx = playerObj['x'] + int(playerObj['size'] / 2)
    playerCentery = playerObj['y'] + int(playerObj['size'] / 2)
    if (camerax + HALF_WINWIDTH) - playerCenterx > CAMSLACKNEW1:
        camerax = playerCenterx + CAMSLACKNEW1 - HALF_WINWIDTH
    elif playerCenterx - (camerax + HALF_WINWIDTH) > CAMSLACKNEW1:
        camerax = playerCenterx - CAMSLACKNEW1 - HALF_WINWIDTH
    if (cameray + HALF_WINHEIGHT) - playerCentery > CAMSLACKNEW2:
        cameray = playerCentery + CAMSLACKNEW2 - HALF_WINHEIGHT
    elif playerCentery - (cameray + HALF_WINHEIGHT) > CAMSLACKNEW2:
        cameray = playerCentery - CAMSLACKNEW2 - HALF_WINHEIGHT
```

3.

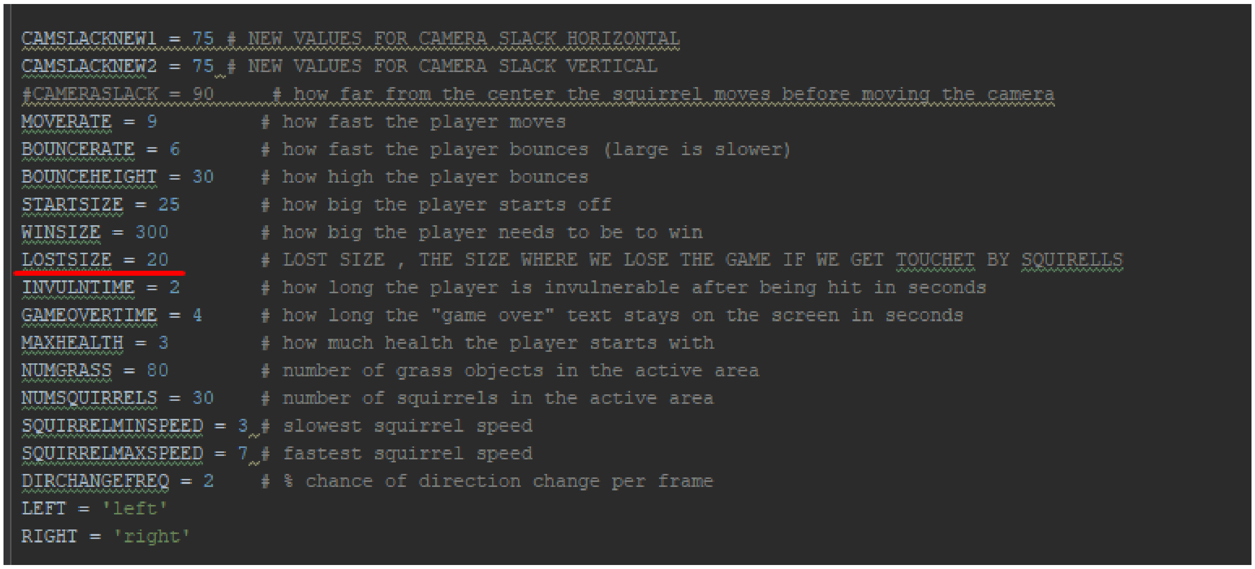
4. If we look at the graph of sizes, where the size of the squirell is calculated and their property



-we get that , the growth is calculated by the widht and height +1 of the eaten squirell , we get the same logic if you eat , just reverse it and put it with -= , that was we lose size , we define a constant of LOSTSIZE, and we are ready to use our new mechanics, we get to lose the game after 1 hit , cause the start size is 25 and the lose size is 20.

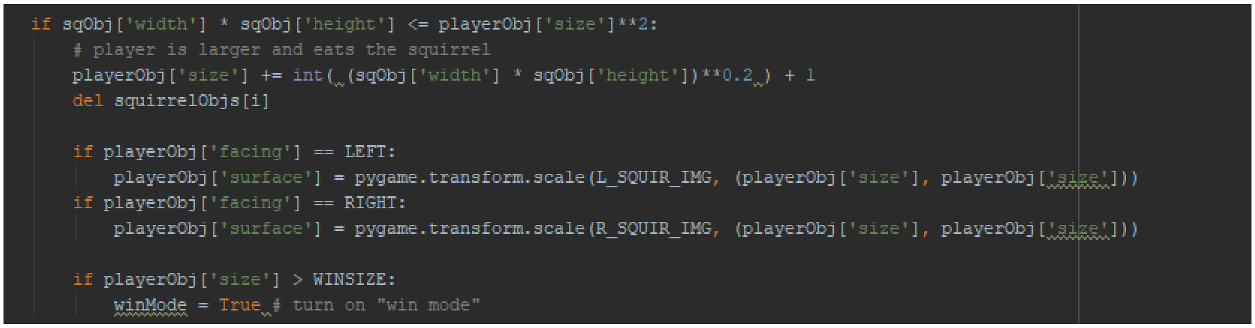


```
        elif not invulnerableMode:
            # player is smaller and takes damage
            invulnerableMode = True
            invulnerableStartTime = time.time()
           # playerObj['health'] -= 1
            playerObj['size'] -= int((sqObj['width'] * sqObj['height']) ** 0.2) + 1
           # if playerObj['health'] == 0:
            if playerObj['size'] <= LOSTSIZE:
                gameOverMode = True # turn on "game over mode"
                gameOverStartTime = time.time()
```



```
CAMSLACKNEW1 = 75 # NEW VALUES FOR CAMERA SLACK HORIZONTAL
CAMSLACKNEW2 = 75 # NEW VALUES FOR CAMERA SLACK VERTICAL
#CAMERASLACK = 90      # how far from the center the squirrel moves before moving the camera
MOVERATE = 9          # how fast the player moves
BOUNCERATE = 6        # how fast the player bounces (large is slower)
BOUNCEHEIGHT = 30     # how high the player bounces
STARTSIZE = 25        # how big the player starts off
WINSIZE = 300         # how big the player needs to be to win
LOSTSIZE = 20         # LOST SIZE , THE SIZE WHERE WE LOSE THE GAME IF WE GET TOUCHET BY SQUIRELLS
INVULNTIME = 2        # how long the player is invulnerable after being hit in seconds
GAMEOVERTIME = 4      # how long the "game over" text stays on the screen in seconds
MAXHEALTH = 3         # how much health the player starts with
NUMGRASS = 80         # number of grass objects in the active area
NUMSQUIRRELS = 30     # number of squirrels in the active area
SQUIRRELMINSPEED = 3 # slowest squirrel speed
SQUIRRELMAXSPEED = 7 # fastest squirrel speed
DIRCHANGEFREQ = 2     # % chance of direction change per frame
LEFT = 'left'
RIGHT = 'right'
```

We use the same logic as in this code



```
if sqObj['width'] * sqObj['height'] <= playerObj['size']**2:
    # player is larger and eats the squirrel
    playerObj['size'] += int( (sqObj['width'] * sqObj['height'])**0.2 ) + 1
    del squirrelObjs[i]

    if playerObj['facing'] == LEFT:
        playerObj['surface'] = pygame.transform.scale(L_SQUIR_IMG, (playerObj['size'], playerObj['size']))
    if playerObj['facing'] == RIGHT:
        playerObj['surface'] = pygame.transform.scale(R_SQUIR_IMG, (playerObj['size'], playerObj['size']))

    if playerObj['size'] > WINSIZE:
        winMode = True # turn on "win mode"
```

As when the sq object of random sq is smaller than us and we get up in size , we lose that with -= if we hit a bigger squirell and we remove the HP losing methods.