```
import turtle # this is a libery
import random # for increasing size of snake
import time
delay =  0.2 # for increasing & decreasing the speed of snake
sc = 0 # score card
hs = 0 # highest score
bodies = [] # empty list when snake goes to food then end the food

# creating the screen
s1=turtle.Screen() # creating object
s1.title("Snake game")
s1.bgcolor("Sky blue")
s1.setup(width=600,height=600)
s1.tracer(0)  # to manually control screen updates

# creating a head
h1=turtle.Turtle()
h1.shape("circle")
h1.speed(0)
h1.fillcolor("blue") # inner side color of circle
h1.penup() # deu to this face not drwaing  any line
h1.goto(0,0) # for change position of circle
h1.direction="stop"

# creation a food
f1=turtle.Turtle()
f1.speed(0) # food movinf speed
f1.shape("square")
f1.color("green")
f1.penup()# dont show line
f1.ht() # when food want to change position then he hide
f1.goto(150,240) # position of food by using x & y axis
f1.st()# function for food show after change the position
f1.direction="stop"

# creating scroreboard
sb=turtle.Turtle()
sb.hideturtle()  # tide the scoreboard
sb.penup()
sb.goto(-270,270) # position of score board using graph
sb.write("Score: 0 | highest score: 0 ", font=("Arial", 14, "bold")) #
for writting a score

def moveup():    # creating function
    if h1.direction!="down" : # if snake want to go up then his direction
should be not down
        h1.direction="up"

def moveleft():
    if h1.direction!="right":
        h1.direction="left"

def moveright():
    if h1.direction!="left":
        h1.direction="right"

def movedown():
    if h1.direction!="up":
```

```
        h1.direction="down"

def movestop():
    h1.direction="stop" # function for stop the game

def move():
    if h1.direction=="up": # snake going to up side & according to graph
up side have y axis us there using y
        y=h1.ycor() #  for giving the tavlue for change the position of
snake
        h1.sety(y+20)# snake value will change while trevelling
    if h1.direction =="left":
        x=h1.xcor()
        h1.setx(x-20) # - for snake going to left side
    if h1.direction =="down":
        y=h1.ycor()
        h1.sety(y-20)
    if h1.direction =="right":
        x=h1.xcor()
        h1.setx(x+20)

# event handling
s1.listen() # for giving order using keyboard
s1.onkey(moveup,"Up") # capital U for up
s1.onkey(movedown,"Down")
s1.onkey(moveleft,"Left")
s1.onkey(moveright,"Right")
s1.onkey(movestop,"space")


while True:
    s1.update() # to update screen
    if h1.xcor()>290:     #check collison with border
        h1.setx(-290) # of snake scross the border then he come from
apposite side border this number is used according to x & y axis

    if h1.xcor()<-290:
        h1.setx(290)

    if h1.ycor()>290:
        h1.sety(-290)

    if h1.ycor()<-290:
        h1.sety(290)

    if h1.distance(f1)<20:   # check collison with food
        x=random.randint(-200,200) # take any random number between -
200,200 for food position changing
        y=random.randint(-200,200)
        f1.goto(x,y)

        # increasing the body of snake
        b1=turtle.Turtle() # mkaing the body of snake
        b1.speed(0)
        b1.penup()
        b1.shape("square")
        b1.color("red")
        bodies.append(b1) # bodies means empty list which maked up side
```

```
        # increasing the score
        sc=sc+10
        if sc>hs:
            hs=sc
        sb.clear() # score board
        sb.write("Score:{} | Highest Score: {}".format(sc,hs),
font=("Arial", 14, "bold"))
        if sc % 50 ==0:  # increasing the speed after every 50 points
            delay = delay-0.001

    # moving the snake body
    for i in range(len(bodies)-1,0,-1):
        x=bodies[i-1].xcor() # means fond the position of  x cordinate
        y=bodies[i-1].ycor()
        bodies[i].goto(x,y)

    if len(bodies)>0:
        x=h1.xcor()
        y=h1.ycor()
        bodies[0].goto(x,y)

    move()

    # if check is snake whith his body then staop the game
    for b in bodies:
        if b.distance(h1)<20:
            time.sleep(1)
            h1.goto(0,0)
            h1.direction="stop"
            for body in bodies: # hide the face of body
                body.ht()
            bodies.clear()
            sc=0
            delay=0.2 # when game is end then again speed put remain.
            sb.clear()
            sb.write("Score:{} | Highest score:{}".format(sc,hs),
font=("Arial", 14, "bold"))
    time.sleep(delay)
```