

```

from turtle import *
from datetime import *

def Skip(step):
    penup()
    forward(step)
    pendown()

def mkHand(name, length):
    #注册Turtle形状, 建立表针Turtle
    reset()
    Skip(-length*0.1)
    begin_poly()
    forward(length*1.1)
    end_poly()
    handForm = get_poly()
    register_shape(name, handForm)

def Init():
    global secHand, minHand, hurHand, printer
    mode("logo")# 重置Turtle指向北
    #建立三个表针Turtle并初始化
    mkHand("secHand", 125)
    mkHand("minHand", 130)
    mkHand("hurHand", 90)
    secHand = Turtle()
    secHand.shape("secHand")
    minHand = Turtle()
    minHand.shape("minHand")
    hurHand = Turtle()
    hurHand.shape("hurHand")
    for hand in secHand, minHand, hurHand:
        hand.shapesize(1, 1, 3)
        hand.speed(0)
    #建立输出文字Turtle
    printer = Turtle()
    printer.hideturtle()
    printer.penup()

def SetupClock(radius):
    #建立表的外框
    reset()
    pensize(7)
    for i in range(60):
        Skip(radius)
        if i % 5 == 0:
            forward(20)
            Skip(-radius-20)
        else:
            dot(5)
            Skip(-radius)
        right(6)

def Week(t):
    week = ["星期一", "星期二", "星期三",
            "星期四", "星期五", "星期六", "星期日"]
    return week[t.weekday()]

def Date(t):
    y = t.year
    m = t.month
    d = t.day
    return "%s %d %d" % (y, m, d)

def Tick():
    #绘制表针的动态显示
    t = datetime.today()
    second = t.second + t.microsecond*0.000001
    minute = t.minute + second/60.0
    hour = t.hour + minute/60.0
    secHand.setheading(6*second)
    minHand.setheading(6*minute)
    hurHand.setheading(30*hour)

    tracer(False)
    printer.forward(65)
    printer.write(Week(t), align="center",
                  font=("Courier", 14, "bold"))
    printer.back(130)
    printer.write(Date(t), align="center",
                  font=("Courier", 14, "bold"))

```

```
printer.home()
tracer(True)

ontimer(Tick, 100) #100ms后继续调用tick

def main():
    tracer(False)
    Init()
    SetupClock(160)
    tracer(True)
    Tick()
    mainloop()

if __name__ == "__main__":
    main()
```