

Analog_Clock_Example__Advanced_v2_

```
//Analog Clock Example (Advanced v2)
//loop example includes "for loop" for(a=0,a<100,a++)
//uses map function to determine angle from time
//Mr. H.
```

```
int rDash = 200;
int rSecond = 180;
int rMinute = 160;
int rHour = 140;
float x;
float y;
//declares centre x as cX and centre y as cY
int cX, cY;
//declares hour to account for a.m. and p.m.
int hour;
```

```
void setup() {
  size(600, 600);
  //finds cX and cY after width and height declared
  cX = width/2;
  cY = height/2;
}
```

```
void draw() {
  background(255);
```

```
  //draws circle outline of clock
  noFill();
  stroke(0);
  strokeWeight(2);
  ellipse(cX, cY, 480, 480);
```

```
  // Draw the minute ticks
  strokeWeight(4);
  stroke(0, 0, 200);
  beginShape(POINTS);
  //360° divided by 60 minutes gives 6° per minute
  for (int a = 0; a < 360; a+=6) {
    float angle = radians(a);
    float x = cX + cos(angle) * rDash;
    float y = cY + sin(angle) * rDash;
    vertex(x, y);
  }
```

```
  // Draw the five minute dashes
  strokeWeight(5);
  stroke(0, 0, 200);
```

```

beginShape(POINTS);
//there are 12 five minute dashes, 360° divided by 12 is 30° per dash
for (int a = 0; a < 360; a+=30) {
  float angle = radians(a);
  line(cX+cos(angle)*rDash*1.05, cY+sin(angle)*rDash*1.05, cX+cos(angle)*rDash*0.95,
cY+sin(angle)*rDash*0.95);
}

//converts 24 hour time to a.m. and p.m.
if (hour() > 12) {
  hour = hour() - 12;
} else {
  hour = hour();
}
//draws hour hand of clock
x = cX+cos(map(hour, 0, 12, 0, TWO_PI) - HALF_PI)*rHour;
y = cY+sin(map(hour, 0, 12, 0, TWO_PI) - HALF_PI)*rHour;
stroke(0, 200, 0);
strokeWeight(8);
line(cX, cY, x, y);

//draws minute hand of clock
x = cX+cos(map(minute(), 0, 60, 0, TWO_PI) - HALF_PI)*rMinute;
y = cY+sin(map(minute(), 0, 60, 0, TWO_PI) - HALF_PI)*rMinute;
stroke(0, 200, 0);
strokeWeight(5);
line(cX, cY, x, y);

//draws second hand of clock
x = cX+cos(map(second(), 0, 60, 0, TWO_PI) - HALF_PI)*rSecond;
y = cY+sin(map(second(), 0, 60, 0, TWO_PI) - HALF_PI)*rSecond;
stroke(255, 0, 0);
strokeWeight(2);
line(cX, cY, x, y);

//creates a black "cap" on the clock hands
fill(0);
stroke(0);
ellipse(cX, cY, 15, 15);
}

```