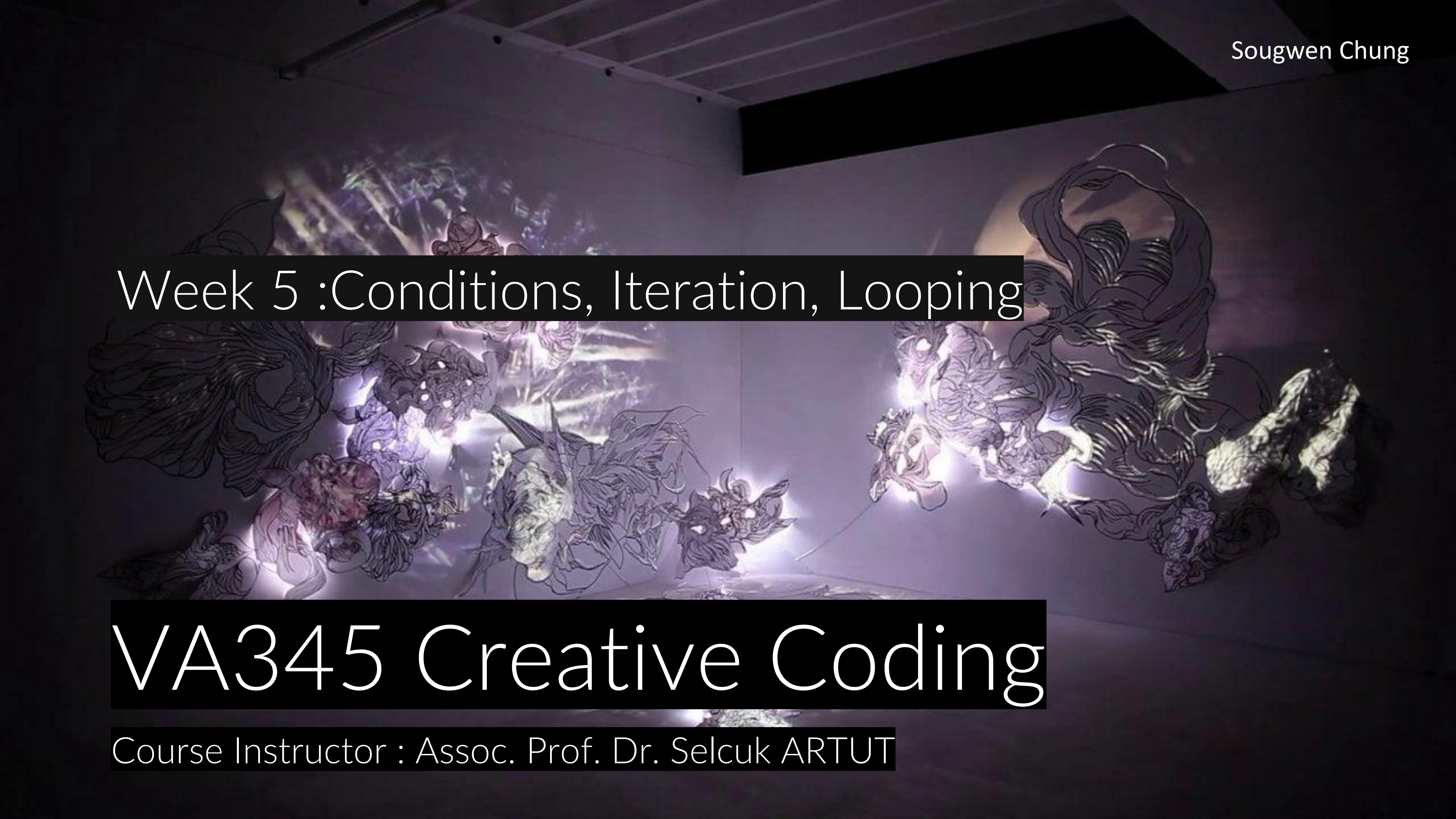


Sougwen Chung

Week 5 :Conditions, Iteration, Looping

VA345 Creative Coding

Course Instructor : Assoc. Prof. Dr. Selcuk ARTUT

Sougwen Chung

Sougwen Chung (愫君) is a Chinese-born, Canadian-raised artist based in New York

http://sougwen.com/

Reminder: Whitespace

In many programming languages, including p5.js, there can be an arbitrary amount of space between the elements of a program. Unlike the rigorous syntax of statement terminators, spacing does not matter. The following two lines of code are a standard way of writing a program:

```
createCanvas(200, 200);
background(102);
```

However, the whitespace between the code elements can be set to any arbitrary amount and the program will run exactly the same way:

```
createCanvas
( 200,
200) ;
background ( 102)
;
```

Tool: Console

When software runs, the computer performs operations at a rate too fast to perceive with human eyes.

Because it is important to understand what is happening inside the machine, the functions console.log() and print() can be used to display data while a program is running.

The console can be used to display a variable, confirm an event, or check incoming data from an external device.

```
console.log("Hello World");
print("Hello World");
```

Variable Scope

Variables have a global or function "scope". For example, variables declared within either the setup() or draw() functions may be only used in these functions.

Global variables, variables declared outside of setup() and draw(), may be used anywhere within the program. If a function variable is declared with the same name as a global variable, the program will use the function variable to make its calculations within the current scope.

https://p5js.org/examples/data-variable-scope.html

Week 5 :Conditions, Iteration

Conditionals

Conditionals allow a program to make decisions about which lines of code run and which do not. They let actions take place only when a specific condition is met.

Conditionals allow a program to behave differently depending on the values of their variables. For example, the program may draw a line or an ellipse depending on the value of a variable. The if structure is used in Processing to make these decisions:

```
if (test) {
statements
}
```

Week 5 :Conditions, Iteration, Looping

Conditions

Example:

```
if(aNumber == 3){
        fill(255,0,0);
        ellipse(50,50,80,80);
}
```

The two lines of code inside the curly brackets are executed only when the condition is met –i.e. when the value of the variable **aNumber** is **3**.

Conditions

Example:

```
if(aNumber == 3){
        fill(255,0,0);
        ellipse(50,50,80,80);
}else{
        fill(0,255,0);
        ellipse(50,50,80,80);
}
```

With else, the if condition is expanded by one code snippet that is executed when the condition is not met.

Conditions

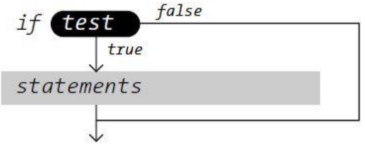
Example:

```
if(aNumber == 3) fill(255,0,0);
else fill(0,255,0);
```

If a code snippet consists of only one line, the curly brackets can be eliminated

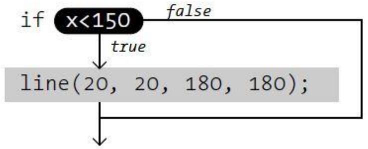
General case if structure

```
if (test) {
  statements
}
```



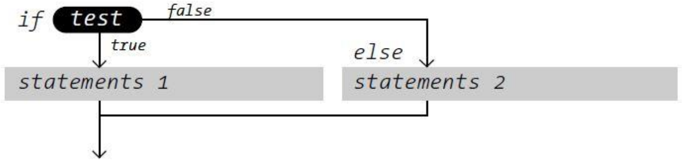
A specific if structure

```
if (x < 150) {
  line(20, 20, 180, 180);
}
```



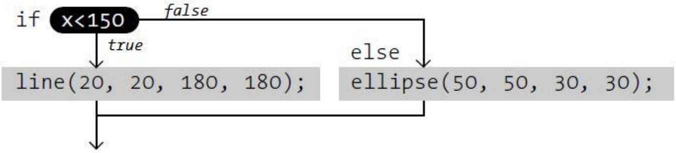
General case if/else structure

```
if (test) {
  statements 1
} else {
  statements 2
}
```



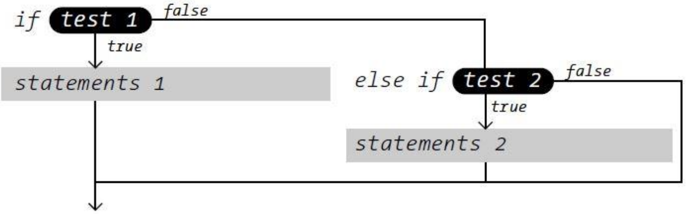
A specific if/else structure

```
if (x < 150) {
  line(20, 20, 180, 180);
} else {
  ellipse(50, 50, 30, 30);
}
```



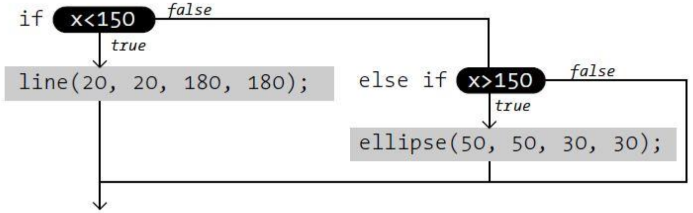
General case if/else if structure

```
if (test 1) {
  statements 1
} else if (test 2) {
  statements 2
}
```



A specific if/else if structure

```
if (x < 150) {
  line(20, 20, 180, 180);
} else if (x > 150) {
  ellipse(50, 50, 30, 30);
}
```



Logical operators

Logical operators are used to combine two or more relational expressions and to invert logical values. They allow for more than one condition to be considered simultaneously.

The logical operators are symbols for the logical concepts of AND, OR, and NOT:

Operator	Meaning
&&	AND
||	OR
!	NOT

The following table outlines all possible combinations and the results.

Expression	Evaluation
true && true	true
true && false	false
false && false	false
true || true	true
true || false	true
false || false	false
!true	false
!false	true

Let’s move things here and there

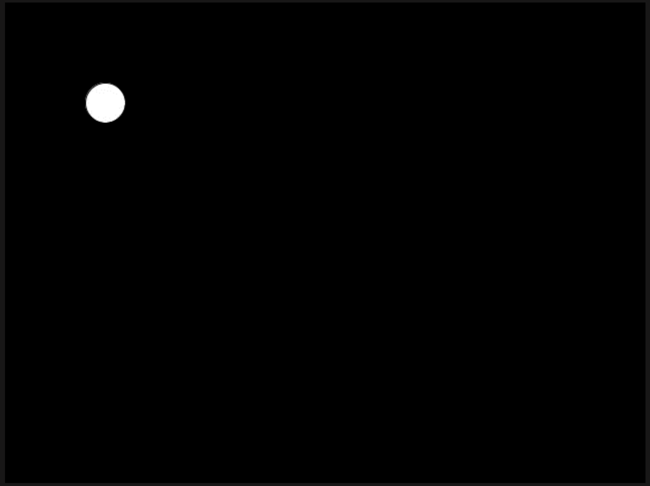
To put a shape into motion, use a variable to change its attributes.

```
let x = 100;
function setup(){
        createCanvas(640,480);
}
function draw(){
        ellipse(x,100,40,40);
        x = x + 1;
}
```

Coding Challenge (30 minutes)

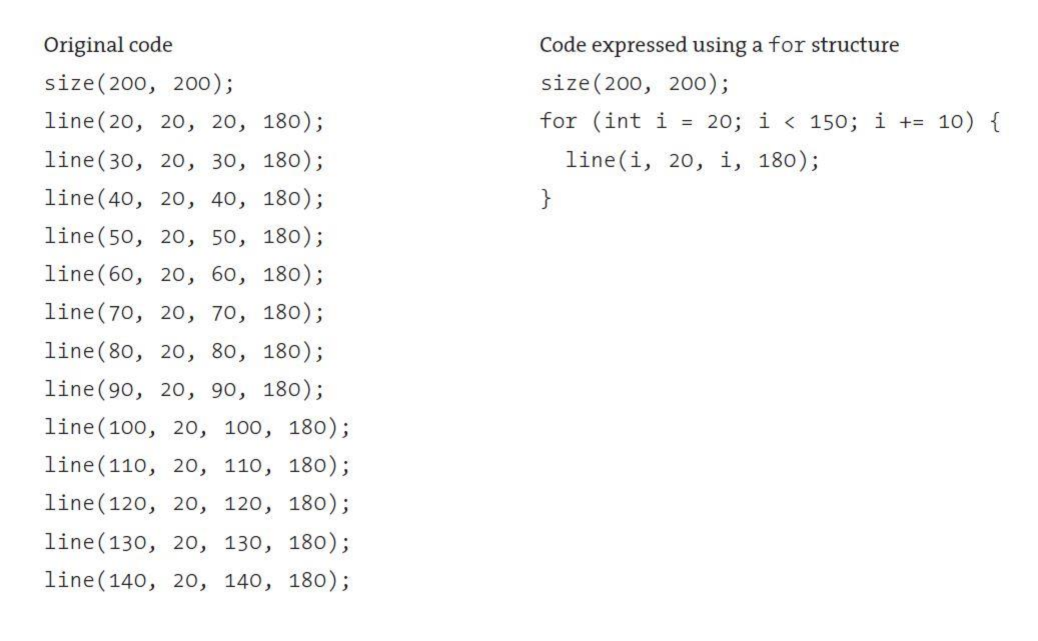
Accomplish this animaton.

Ambition Challenge : Use XY axis



Iteration

Iterative structures are used to compact lengthy lines of repetitive code. Decreasing the length of the code can make programs easier to manage and can also help to reduce errors.



Original code

```
size(200, 200);
line(20, 20, 20, 180);
line(30, 20, 30, 180);
line(40, 20, 40, 180);
line(50, 20, 50, 180);
line(60, 20, 60, 180);
line(70, 20, 70, 180);
line(80, 20, 80, 180);
line(90, 20, 90, 180);
line(100, 20, 100, 180);
line(110, 20, 110, 180);
line(120, 20, 120, 180);
line(130, 20, 130, 180);
line(140, 20, 140, 180);
```

Code expressed using a `for` structure

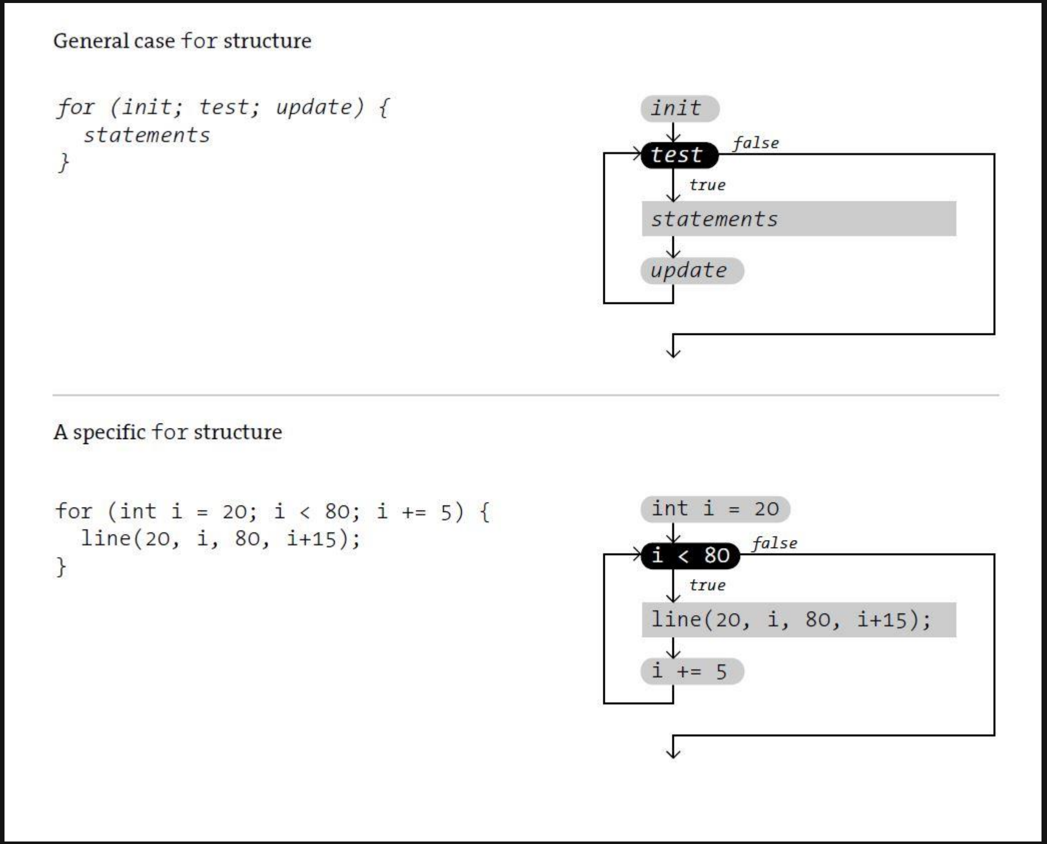
```
size(200, 200);
for (int i = 20; i < 150; i += 10) {
  line(i, 20, i, 180);
}
```

Loops : for loop

Example:

```
for (int i = 0; i<=5; i++){
          line(0,0,i*20,100);
          line(100,0,i*20,100);
}
```

The two lines of code inside the curly brackets are executed exactly six times. First the variable i is set to the value 0, then increased by 1 (i++) after each cycle, as long as the value is 5 or less.



General case for structure

```
for (init; test; update) {
  statements
}
```

A specific for structure

```
for (int i = 20; i < 80; i += 5) {
  line(20, i, 80, i+15);
}
```

1. The *init* statement is run
2. The *test* is evaluated to *true* or *false*
3. If the *test* is *true*, continue to step 4. If the *test* is *false*, jump to step 6
4. Run the statements within the block
5. Run the *update* statement and jump to step 2
6. Exit the structure and continue running the program

Loops : while loop

Example:

```
float myValue = 0;
while (myValue < 100){
        myValue = myValue + random(5)
}
```

The two lines of code inside the curly brackets are executed exactly six times. First the variable i is set to the value 0, then increased by 1 (i++) after each cycle, as long as the value is 5 or less.

Let’s make Islamic Patterns

1. Draw a circle 50 px in diameter
2. Draw a square around the circle
3. Draw a circle half the size of the original circle around the bottom corner of the square
4. Repeat steps 1-3 to the immediate right of their original position
5. Repeat step 4 six times
6. Repeat steps 1-3 immediately below their original position
7. Repeat steps 4-5
8. Repeat steps 6-7 six times

Mouse events

The mouse event functions are mousePressed(), mouseReleased(), mouseMoved(), and mouseDragged():

mousePressed() Code inside this block is run one time when a mouse button is pressed

mouseReleased() Code inside this block is run one time when a mouse button is released

mouseMoved() Code inside this block is run one time when the mouse is moved

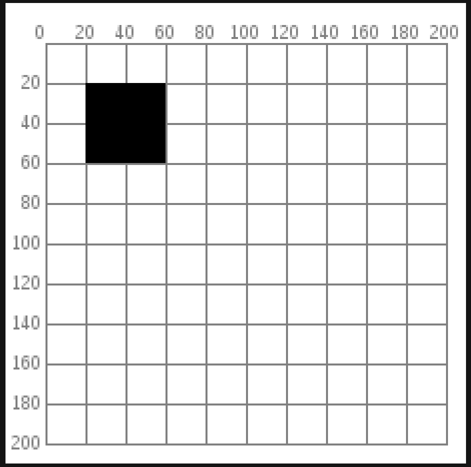
mouseDragged() Code inside this block is run one time when the mouse is moved while a mouse button is pressed

mouseX : https://p5js.org/reference/#/p5/mouseX

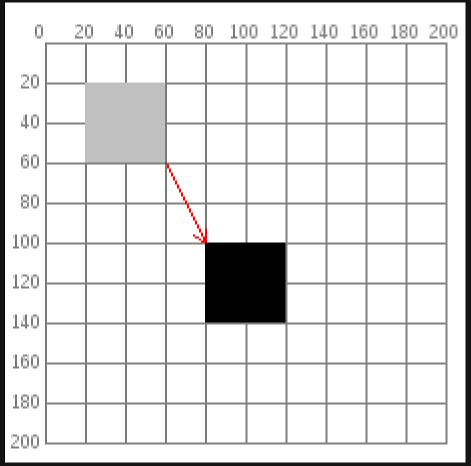
mouseY : https://p5js.org/reference/#/p5/mouseY

Introduction to The Transformation Matrix

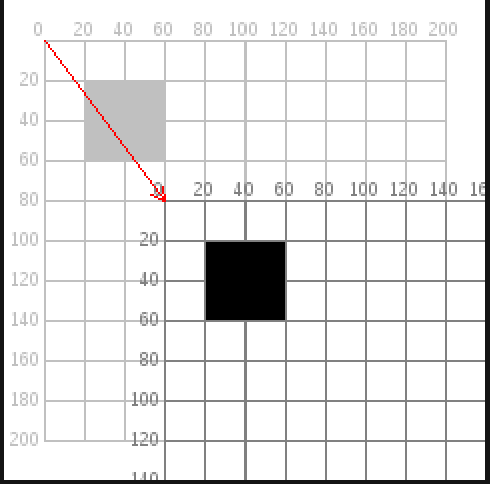
As you know, your p5.js window works like a piece of graph paper. When you want to draw something, you specify its coordinates on the graph. Here is a simple rectangle drawn with the code rect(20, 20, 40, 40). The coordinate system (a fancy word for "graph paper") is shown in gray.



If you want to move the rectangle 60 units right and 80 units down, you can just change the coordinates by adding to the x and y starting point: rect(20 + 60, 20 + 80, 40, 40) and the rectangle will appear in a different place. (We put the arrow in there for dramatic effect.)



But there is a more interesting way to do it: move the graph paper instead. If you move the graph paper 60 units right and 80 units down, you will get exactly the same visual result. Moving the coordinate system is called translation.

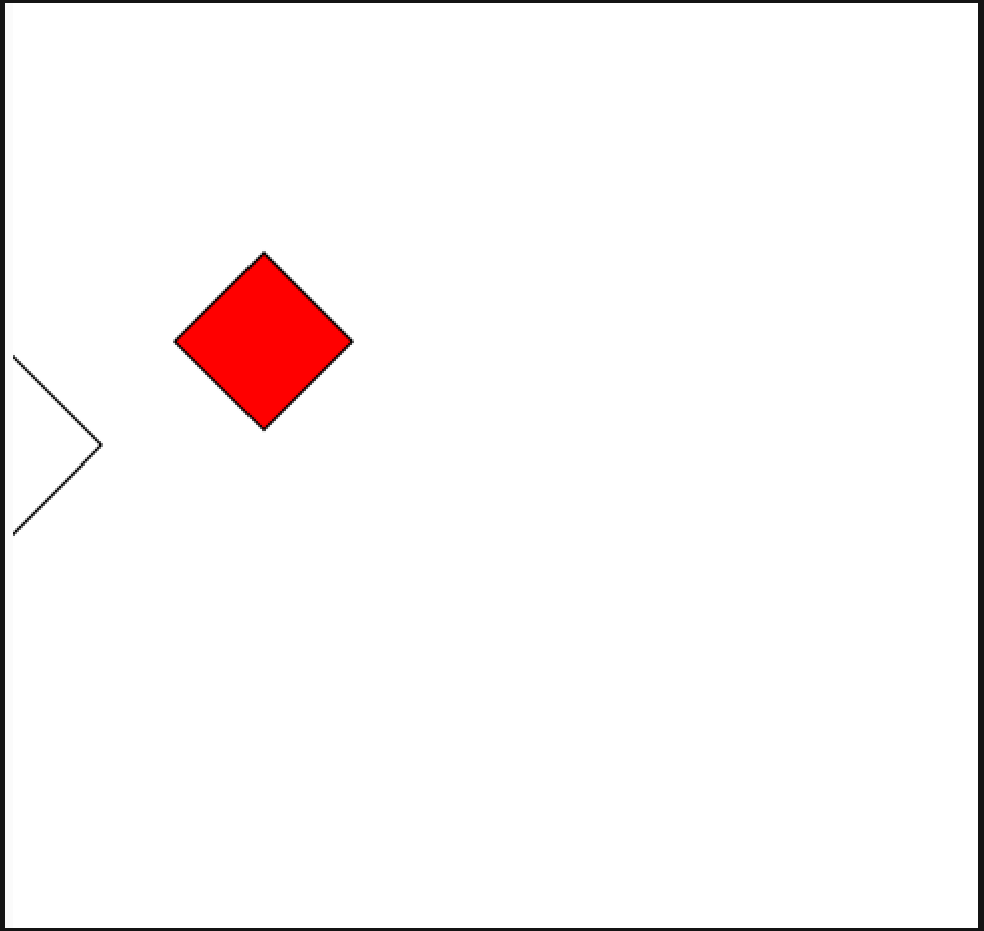


The Transformation Matrix

Every time you do a rotation, translation, or scaling, the information required to do the transformation is accumulated into a table of numbers. This table, or matrix has only a few rows and columns, yet, through the miracle of mathematics, it contains all the information needed to do any series of transformations. And that’s why the push() and pop() have that word in their name.

The Transformation Matrix

```
function setup() {
        createCanvas(400, 400);
        angleMode(DEGREES);
}
function draw() {
        background(255);
        noFill();
        push();
        rotate(45);
        translate(100, 100);
        rect(0, 0, 50, 50);
        pop();
        fill(color(255, 0, 0));
        push();
        translate(100, 100);
        rotate(45);
        rect(0, 0, 50, 50);
        pop();
}
```

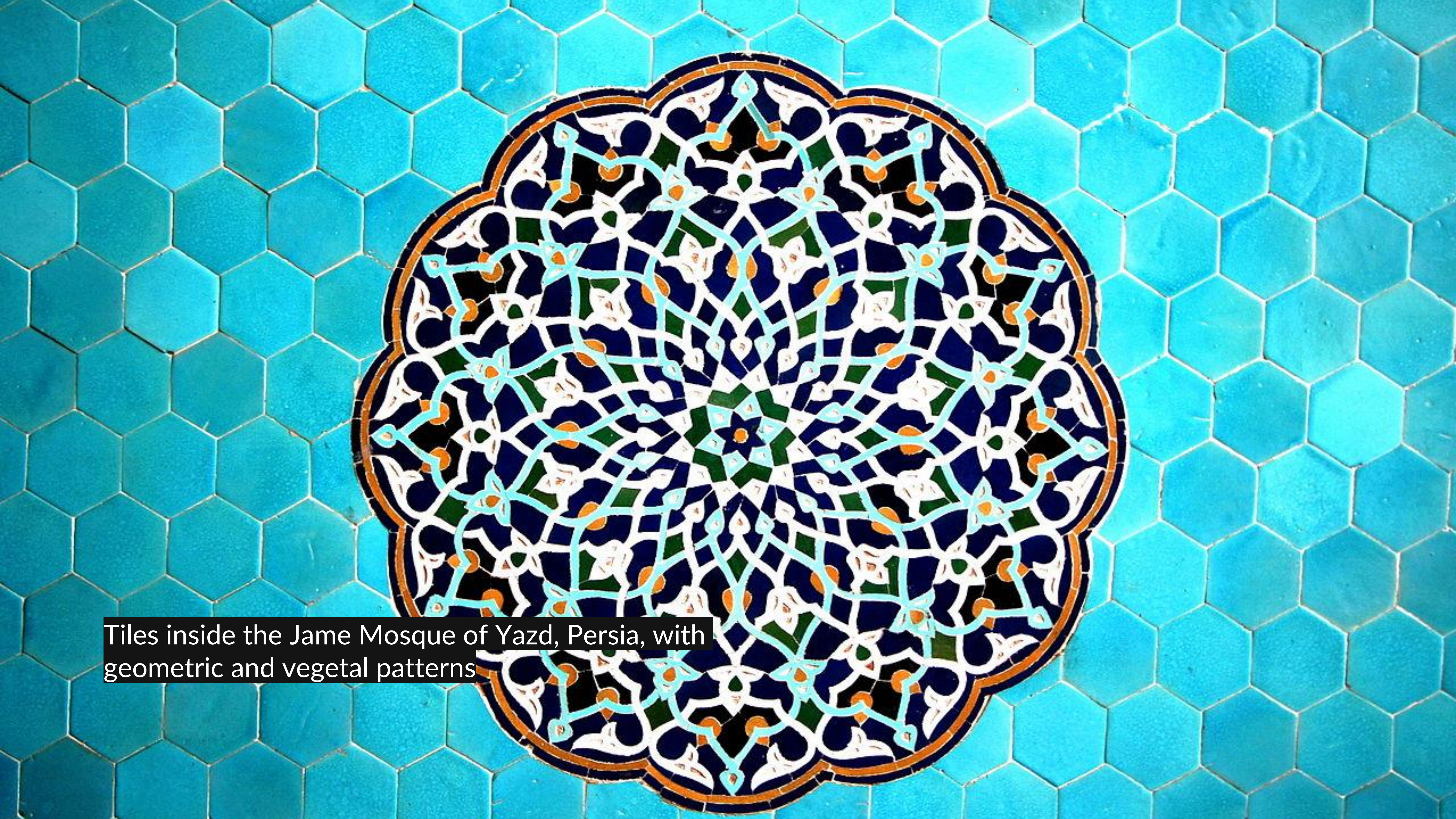


Islamic Geometric Patterns

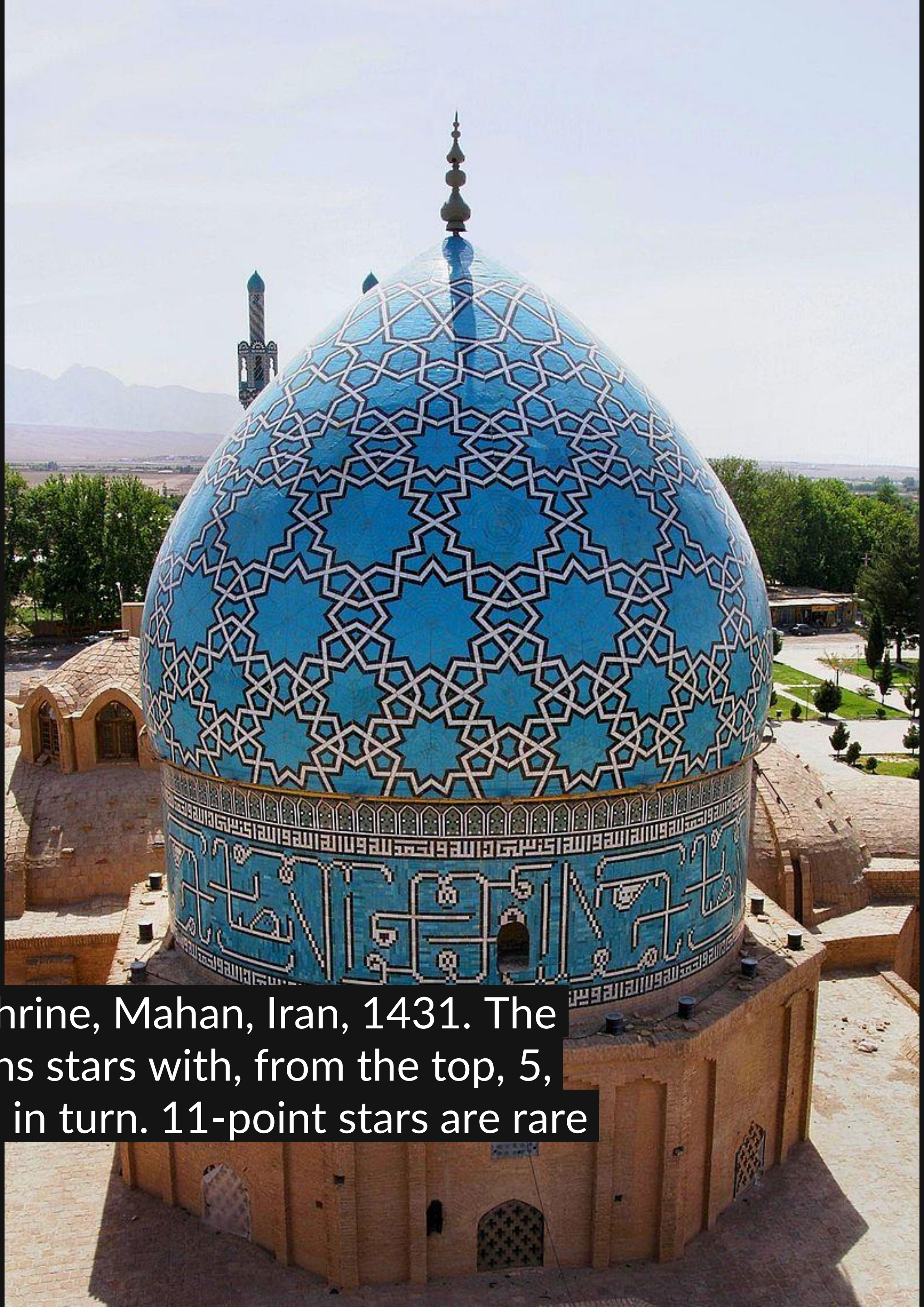
The geometric designs in Islamic art are often built on combinations of repeated squares and circles, which may be overlapped and interlaced, as can arabesques (with which they are often combined), to form intricate and complex patterns, including a wide variety of tessellations.

Islamic art mostly avoids figurative images to avoid becoming objects of worship. This aniconism in Islamic culture caused artists to explore non-figural art and created a general aesthetic shift toward mathematically-based decoration.

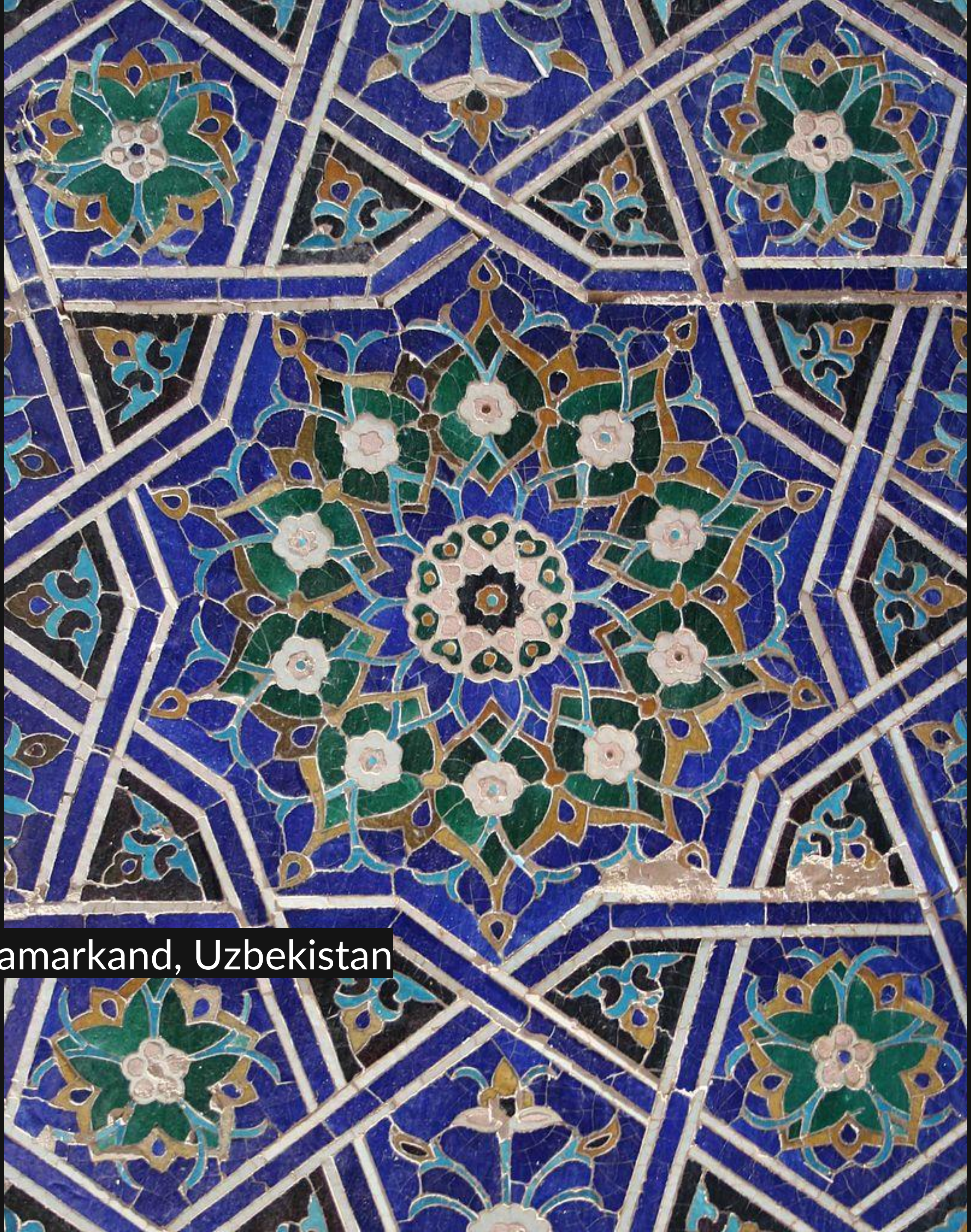
In Islamic culture, the patterns are believed to be the bridge to the spiritual realm, the instrument to purify the mind and the soul.



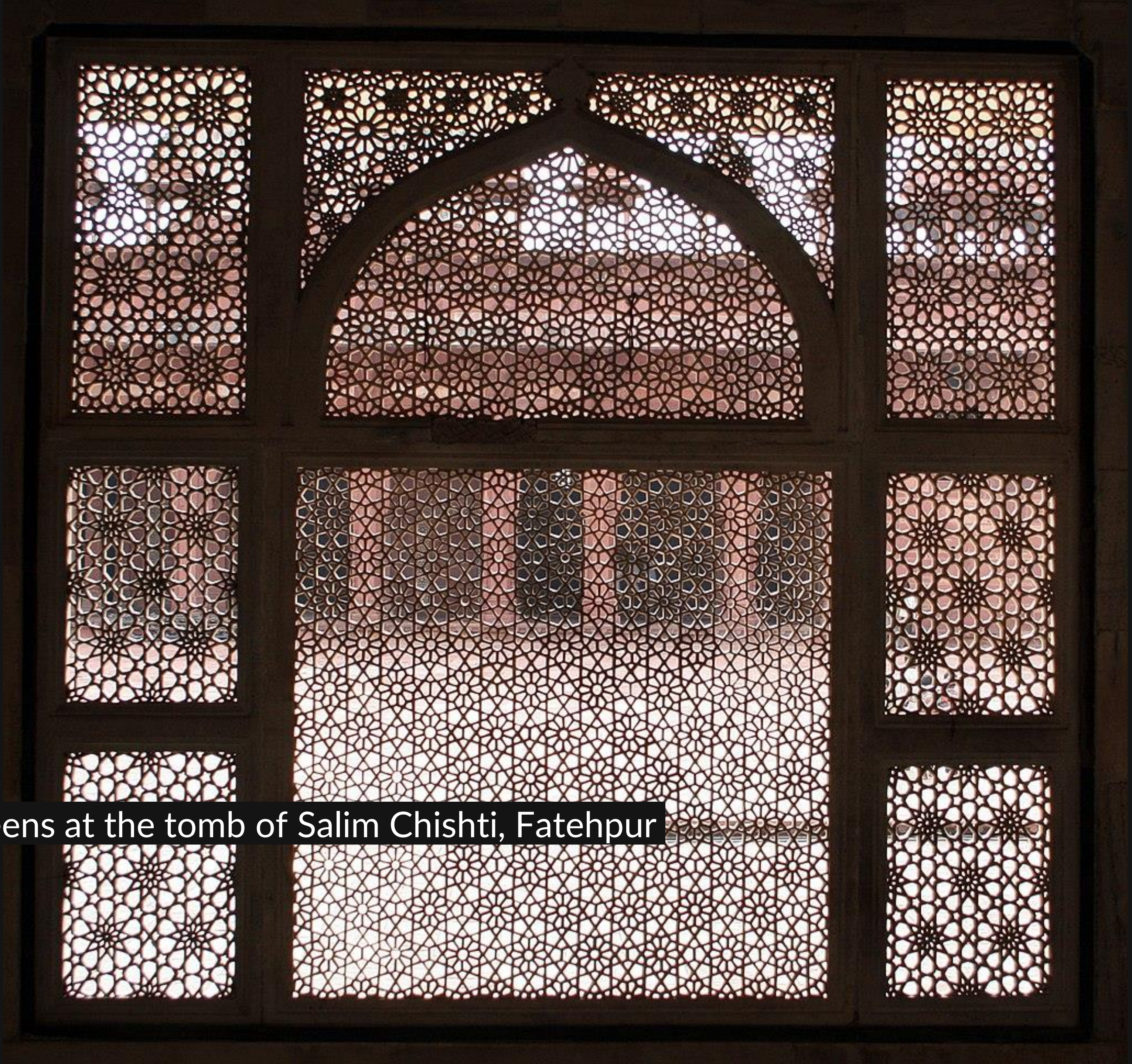
Tiles inside the Jame Mosque of Yazd, Persia, with geometric and vegetal patterns



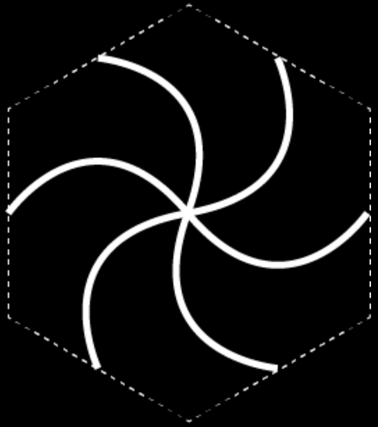
The Shah Nematollah Vali Shrine, Mahan, Iran, 1431. The blue girih-tiled dome contains stars with, from the top, 5, 7, 9, 12, 11, 9 and 10 points in turn. 11-point stars are rare in Islamic art.[11]



Girih at Shah-i-Zinda in Samarkand, Uzbekistan

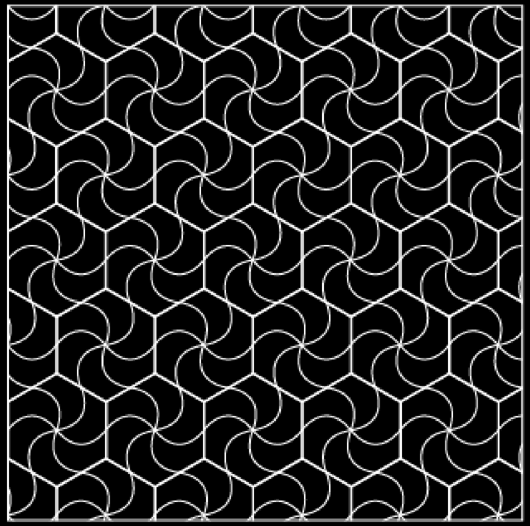


Jali pierced screens at the tomb of Salim Chishti, Fatehpur Sikri, India



Hands on tutorial

Drawing an Islamic Geometric Pattern / Elhambra, Granda Spain (1302-1391)



Assignment 003

Draw this exercise below with Illustrator and p5.js

Drawing an Islamic Geometric Pattern / Elhambra, Granda Spain (1302-1391)