

Snake Game part 4

Step 4: Add Tail

Great! Now we can control the snake's movements. Now let's add the ability for our little snake head to grow a tail.

To make the tail, we will need a list of previous positions (like in the Mouse Trail code from a previous lesson). Instead of a regular `array`, we will use an `ArrayList`, which gives us extra features, like the ability to change the size of the list.

```
ArrayList<PVector> tail = new ArrayList();
```

We will also need a variable to keep track of how long the tail should be.

```
int tailLength;
```

Then, we will create two new methods, one to move the tail, and one to show the tail.

```
void moveTail() {
  if (tailLength > 0) {
    if (tailLength == tail.size() && !tail.isEmpty()) {
      tail.remove(0);
    }
    tail.add(new PVector(snake.x, snake.y));
  }
}

void showTail() {
  for (PVector section : tail) {
    square(section.x, section.y, gridSize);
  }
}
```

Later, the tail will grow when it eats food. For now, we'll just have the tail grow when the mouse is clicked.

```
void mousePressed() {
  tailLength++;
}
```

(While we're at it, lets also refactor our `newGame()` code to use the `direction` method. This is optional, but its nice to be consistent.)

```
void newGame() {
  snake = new PVector(width/2, height/2);
  snakeSize = gridSize;
  direction(1, 0);
}
```

Your code will now look like this:

```
int gridSize = 20;

PVector snake;
int snakeSize;
int xSpeed;
int ySpeed;
int tailLength;
ArrayList<PVector> tail = new ArrayList();

void setup() {
  size(600, 600);
  frameRate(10);
  newGame();
}

void draw() {
  background(0);
  moveSnake();
  moveTail();
  showSnake();
  showTail();
}

void newGame() {
  snake = new PVector(width/2, height/2);
  snakeSize = gridSize;
  direction(1, 0);
}

void showSnake() {
  fill(100, 215, 0);
  square(snake.x, snake.y, snakeSize);
}

void moveSnake() {
  snake.x += xSpeed;
  snake.y += ySpeed;

  // This keeps the snake on the board
  snake.x = constrain(snake.x, 0, width-gridSize);
  snake.y = constrain(snake.y, 0, height-gridSize);
}
```

```

void moveTail() {
    if (tailLength > 0) {
        if (tailLength == tail.size() && !tail.isEmpty()) {
            tail.remove(0);
        }
        tail.add(new PVector(snake.x, snake.y));
    }
}

void showTail() {
    for (PVector section : tail) {
        square(section.x, section.y, gridSize);
    }
}

void direction(int x, int y) {
    xSpeed = x * gridSize;
    ySpeed = y * gridSize;
}

void keyPressed() {
    if (keyCode == UP) {
        direction(0, -1);
    } else if (keyCode == DOWN) {
        direction(0, 1);
    } else if (keyCode == RIGHT) {
        direction(1, 0);
    } else if (keyCode == LEFT) {
        direction(-1, 0);
    }
}

void mousePressed() {
    tailLength++;
}

```

Challenges

1. Make the segments of the tail different colors. Perhaps, have each segment get lighter, or darker, or have them follow a pattern of alternating colors.