Snake Game part 2

Step 2: Move Snake

Now, let's make a method that will move our snake, one grid block at a time. For this we will need to create variables for the x and y speed.

We will also use a new Processing method, called `constrain()`, which allows us to keep the value of a variable between a minimum and a maximum value. We will use this to keep our snake on the screen.

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

  snake.x = constrain(snake.x, 0, width-gridSize);
  snake.y = constrain(snake.y, 0, height-gridSize);
}
```

To give the old fashioned movement affect, we will slow down the frame rate, to 10 (by default it is 60). This means that the draw loop will only update 10 times a second, instead of 60 times a second.

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

The code will now look like this:

```
int gridSize = 20;

PVector snake;
int snakeSize;
int xSpeed;
int ySpeed;

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

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

void newGame() {
  //snake.x = width/2;
  //snake.y = height/2;
  snake = new PVector(width/2, height/2);
  snakeSize = gridSize;
  xSpeed = gridSize;
}

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

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

  snake.x = constrain(snake.x, 0, width-gridSize);
  snake.y = constrain(snake.y, 0, height-gridSize);
}
```

Challenges

1. Try playing the game with different frame rates. What seems to work best?
2. Set up a variable for the frame rate. This way we can make the speed get faster as the snake gets longer.