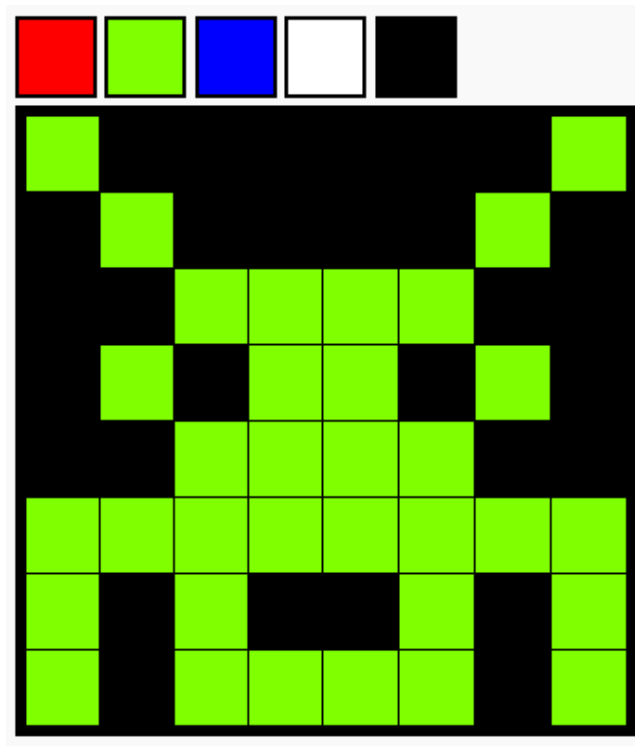


All Code Clubs must be registered. By registering your club we can measure our impact, and we can continue to provide free resources that help children learn to code. You can register your club at [codeclubworld.org](https://codeclubworld.org).

## Передмова

У цьому проекті ви створите редактор піксельної графіки. А також, застосовуючи HTML та CSS, ви дізнаєтесь як використовувати мову програмування JavaScript для створення інтерактивних веб-сторінок.



Як використовувати редактор: Натисніть на колір з палітри, для того щоб вибрати колір вашого пера, після цього натисніть на пікселі для зміни їхнього кольору.



**Список завдань**

Follow these **INSTRUCTIONS** one by one



**Тестувати проект**

Click on the green flag to **TEST** your code



**Зберегти проект**

Make sure to **SAVE** your work now

# Крок 1: Створення сітки пікселів

Зробімо сітку пікселів, яку ви можете використовувати для створення піксельної графіки. CSS надає стилі таблиці для сітки та макетів таблиці.

Таблиці містять рядки, які містять клітинки. Ви збираєтесь створити таблицю з чорним фоном і потім помістити в неї білі пікселі.

## ✓ Список дій

- Відкрийте цю чернетку: [jump.to/cc/web-pixel](http://jump.to/cc/web-pixel). Якщо ви читаете це онлайн, то також можете використати вкладену чернетку нижче.



- Додайте цей html код до `<body>` файлу `index.html`, для того щоб створити `<div>`, який буде містити вашу піксельну графіку. Додатково, надайте своєсо html коду, ідентифікатор `мистецтво`, щоб можна було його стилізувати:



```
</head>
<body>
<div id="art"></div>
</body>
</html>
```

Тепер перейдіть до вашого `style.css` файлу і додайте стиль таблиці для графіки `<div>`. Зверніть увагу, що три рядки пікселів однакові. Введіть перший рядок і потім скопіюйте інші.

```
index.html style.css script.js
#art {
  display: table;
  border-spacing: 1px;
  background-color: black;
  border: 5px solid black;
}
```

Це створює таблицю з межею і встановлює відступи всередині сітки.

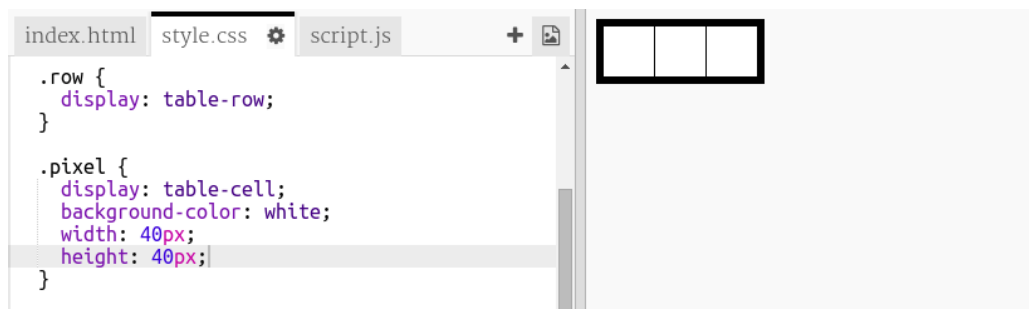
Це не виглядає дуже цікаво поки що, але вам потрібно вставити рядки пікселів всередині таблиці.

- Тепер повернімось до вашого `index.html` файлу і додамо рядок з 3 пікселів всередину графіки `<div>`:

```
<div id="art">
  <div class="row">
    <div class="pixel"></div>
    <div class="pixel"></div>
    <div class="pixel"></div>
  </div>
</div>
```

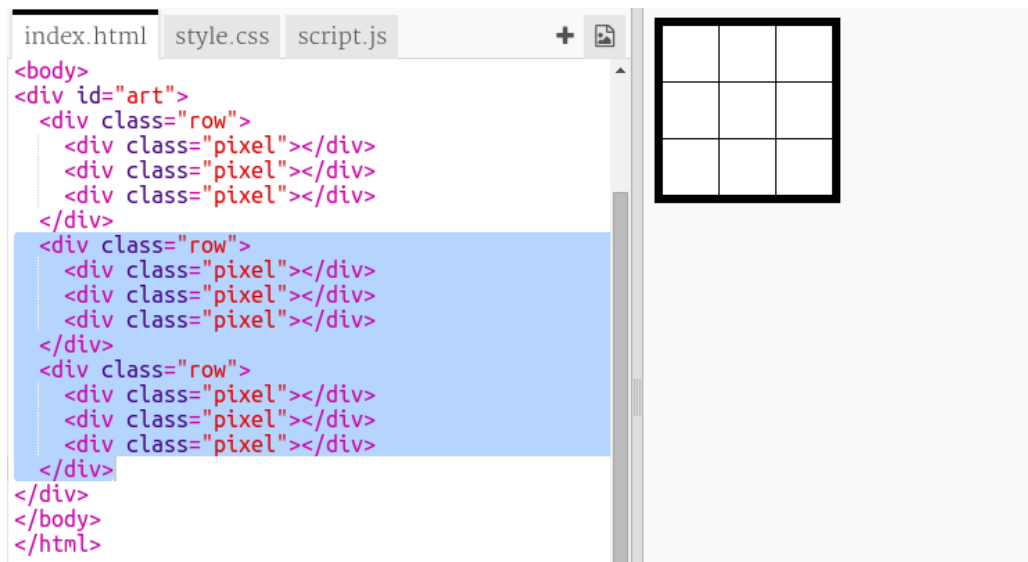
Цього разу ви використає класи, щоб стилізувати теги `div`, так як вони будуть зустрічатися часто.

Додайте наступний стиль для рядків та клітинок:



Тепер ваші пікселі будуть вибудовуватися в сітці з чорними лініями навколо них.

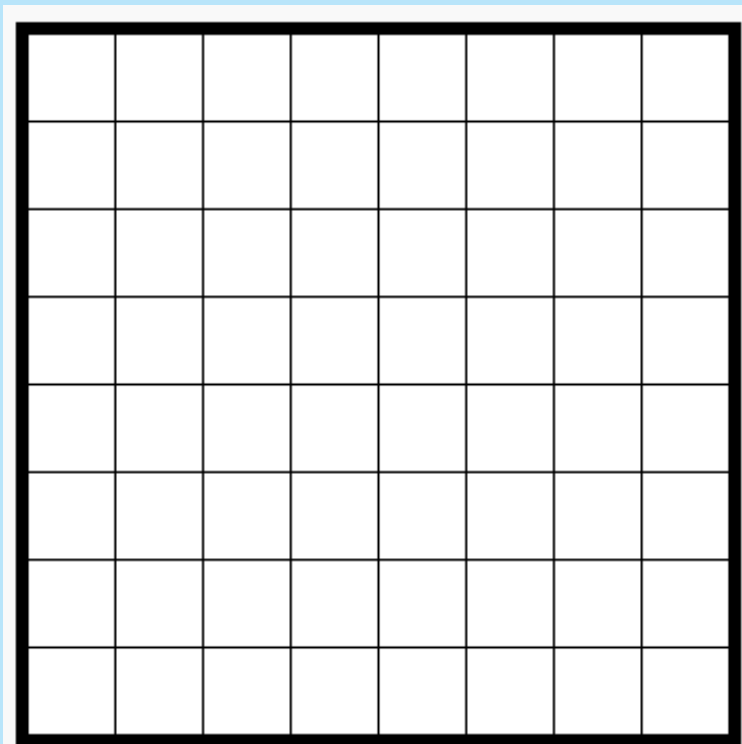
- Тепер додайте ще два рядки пікселів для створення сітки 3 x 3. Пам'ятайте, щоб заощадити час використовуйте копіювання та вставку.



## Завдання: Змініть розмір сітки

3x3 це мала сітка піксельної графіки. Чи можете ви зробити її більшою, розміро наприклад 8x8.

Спробуйте використовувати вирізання і вставку замість введення всього вручну.





## Крок 2: Розмальовуємо пікселі

HTML використовується для організації коду, а CSS - для його стилізування. JavaScript є мовою програмування, що використовується для зміни веб-сторінки, під час взаємодії з нею.

Ви можете використати HTML та CSS для встановлення фонового кольору окремих пікселів, але це буде дуже повільно! Замість цього ви додаєте код JavaScript, щоб розмальовувати пікселі автоматично при натисканні на них.

- ☐ В мові програмування JavaScript код розміщується всередині функції `function`, яку можна викликати, коли ми хочемо запустити код.

Ви створите функцію під назвою `setPixelColour`

Функція `setPixelColour` повинна знати колір якого пікселя потрібно змінити, це є `input` (вхідні дані).

Додайте наступний код до файлу `script.js` для встановлення фонового кольору пікселя:

```
function setPixelColour(pixel)
{
    pixel.style.backgroundColor = 'black';
}
```

Зверніть увагу, що `backgroundColor` використовує американське написання кольору.

- ☐ Тепер нам потрібно викликати функцію, коли піксель натиснутий.

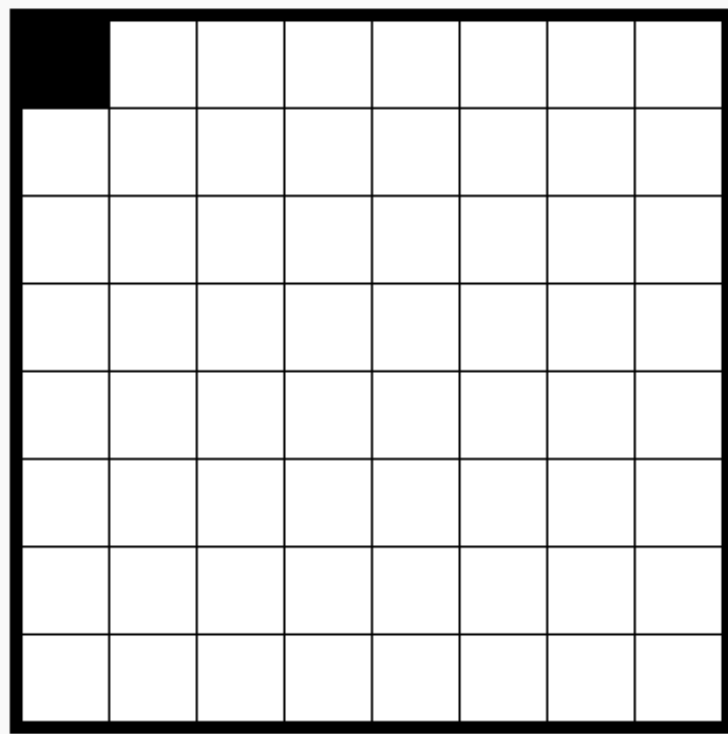
Щоб викликати функцію, коли натиснуто на елемент, HTML

використовує атрибут `onclick`. Потрібно буде передати “цей” як у вхідні дані, так що ваша функція знала колір якого пікселя потрібно змінити.

Перейдіть до файлу `index.html` та додайте наступний код до першого пікселя:

```
<div id="art">
  <div class="row">
    <div class="pixel" onclick="setPixelColour(this)"></div>
    <div class="pixel"></div>
    <div class="pixel"></div>
```

- ☐ Перевірте свій код, натиснувши на перший піксель. Він повинен стати чорним:

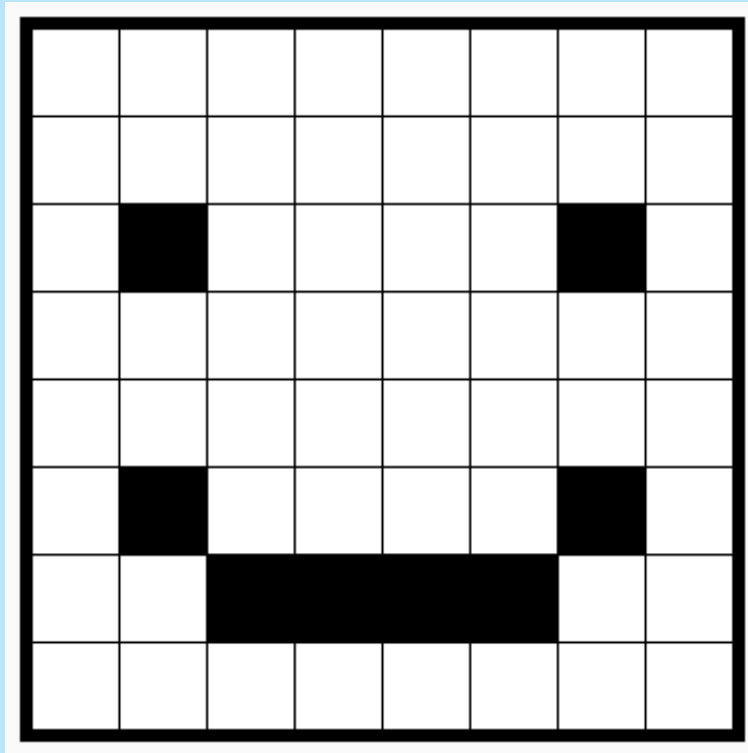


Ви тільки додали `onclick` код до першого пікселя, тому це не буде поки працювати для інших пікселів.

## Завдання: Зробіть всі пікселі клікабельними

Чи можете ви зробити так, щоб на всі пікселі можна було натиснути? Використовуйте вирізання та вставку для того, щоб зробити це швидше.

Створіть швидко частинку піксельної графіки.



Порада: Ви можете клікнути Запустити автоматично, щоб очистити всі пікселі.

## Крок 3: Додати палітру кольорів

Вас також дратує те, що ви не можете змінити колір пікселя назад до білого, якщо зробили помилку? Давайте виправимо це, створивши палітру кольорів так, щоб ви змогли натиснути на колір для зміни пера.

☐ Спочатку створіть стиль пера.



Додайте наступний код у нижню частину вашого файлу `style.css`:

```
.pen {
  display: inline-block;
  width: 40px;
  height: 40px;
  border: 2px solid black;
}
```

- ☐ Тепер створіть чорне і біле кольорове перо, яке використовує цей стиль.

Додайте наступний код до файлу `index.html` після `<body>`:



`style=` дозволяє додавати CSS всередині HTML, що досить зручно.

- ☐ Було б добре змінювати колір пера, після натискання на колір палітри.

Змінні використовують для зберігання інформації. Давайте створимо змінну `penColour` в `script.js`.

Додайте наступний код у верхню частину файлу:

```
var penColour = 'black';
```

Потім додайте функцію, щоб змінити `penColour`:

```
var penColour = 'black';
```

```
function setPenColour(pen)
{
    penColour = pen;
}
```

- ☐ Вам також буде потрібно використати колір пера щоб змінити колір пікселю.

Змініть функцію `setPixelColour` для того, що використати змінну `penColour` замість `чорного`:

```
function setPixelColour(pixel)
{
    pixel.style.backgroundColor = penColour;
}
```

- ☐ Тепер вам потрібно викликати функцію `setPenColour`, коли вибрано колір пера.

Додайте виділений код `onclick` до ваших кольорів пера:

```
<body>
<div id="palette">
  <div class="pen" style="background-color:white;" onclick="setPenColour('white')"></div>
  <div class="pen" style="background-color:black;" onclick="setPenColour('black')"></div>
</div>
<div id="art">
```

- ☐ Тепер перевірте, чи ви можете перемикатися між чорним і білим кольором пера, щоб замальовувати і видаляти пікселі.



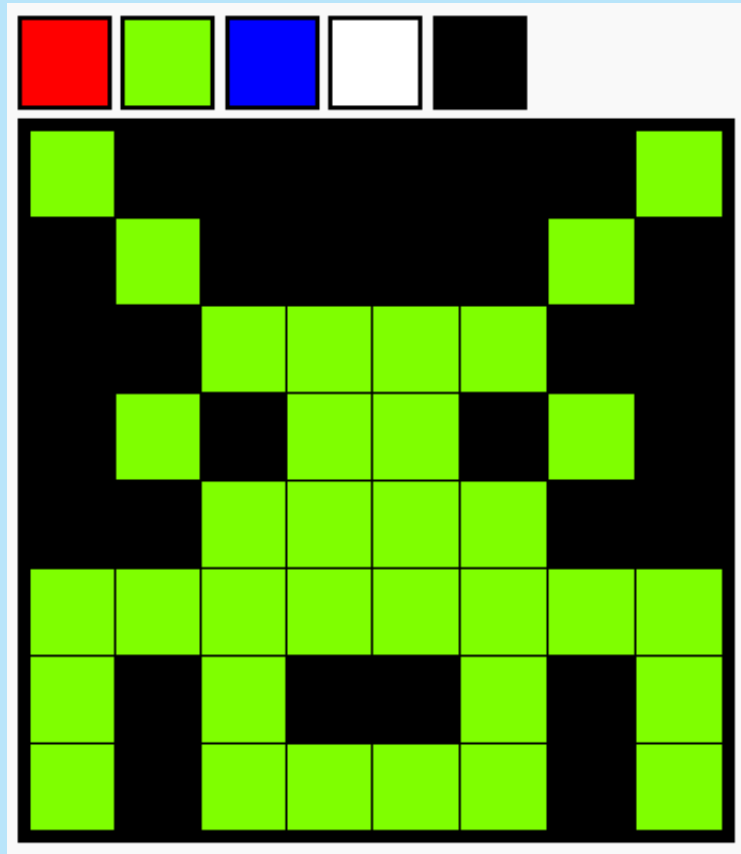
**Збережіть свій проект**

## Завдання: Додайте більше кольорів до палітри

Чи можете ви додати більше кольорів до палітри? Виберіть кольори, з якими ви хочете створити піксельну графіку.

Потім створіть якусь класну піксельну графіку.

Підказка: яскраво-зелений колір називається `chartreuse`.



Запитайте вашого лідера клубу чи ви можете використати програму Windows Snipping Tool або альтернативну програму, щоб зберегти копію вашої графіки як зображення.



Збережіть свій проект