```
/**
 * Screaming Frog JavaScript Extraction:
 * Generate JSON-LD Article Schema
 *
 * How to Use:
 * 1. In Screaming Frog, go to Configuration > Custom >
Extraction.
 * 2. Add a new extraction row, set "Extraction Type" to
"JavaScript".
 * 3. Paste this entire script into the code box.
 * 4. Adjust the selectors & values for your site.
 * 5. Crawl the pages. The "Custom" tab in Screaming Frog will
show the JSON-LD result.
 */




  //
  // --- 1. PAGE URL & CANONICAL ---
  //
  const url = window.location.href;
  const canonicalLinkEl =
document.querySelector('link[rel="canonical"]');
  const canonicalURL = canonicalLinkEl ? canonicalLinkEl.href :
url;

  //
  // --- 2. HEADLINE ---
  //
  // Adjust the selector to match your site’s main headline
element
  // e.g., <h1 class="post-heading">
  const headlineEl = document.querySelector('h1.post-heading');
  const headline = headlineEl
    ? headlineEl.textContent.trim()
    : 'NO HEADLINE FOUND';

  // We'll set 'alternativeHeadline' the same as 'headline' by
default
  const alternativeHeadline = headline;

  //
  // --- 3. META DESCRIPTION ---
  //
  // Uses the standard <meta name="description">
  const metaDescriptionEl =
document.querySelector('meta[name="description"]');
  const metaDescription = metaDescriptionEl
    ? metaDescriptionEl.content.trim()
    : 'NO META DESCRIPTION FOUND';

  //
  // --- 4. PUBLISHED DATE ---
  //
  // Example: a date stored in .text-block-4.date-and-time
  let rawPublished = '';
  const datePublishedEl =
document.querySelector('.text-block-4.date-and-time');
  if (datePublishedEl) {
    rawPublished = datePublishedEl.textContent.trim();
  }
  // Attempt ISO conversion
  let datePublished = rawPublished;
  try {
    datePublished = new Date(rawPublished).toISOString();
  } catch (e) {
    // If parse fails, we'll keep the raw string
  }

  //
  // --- 5. LAST PUBLISHED / MODIFIED DATE ---
  //
  // Example: found in an HTML comment like <!-- Last Published:
Fri Feb 14 2025 ... -->
  // If none found, we'll fallback to 'datePublished'
  let dateModified = datePublished;
  const htmlString = document.documentElement.innerHTML;
  const lastPubMatch = htmlString.match(/Last
Published:\s+(.*?)\s+-->/);
  if (lastPubMatch && lastPubMatch[1]) {
    try {
      dateModified = new Date(lastPubMatch[1]).toISOString();
    } catch (e) {
      dateModified = datePublished;
    }
  }

  //
  // --- 6. AUTHOR ---
  //
  // Example: author is an <a> immediately following an element
with .text-block-2
```

```
  const authorAnchor = document.querySelector('.text-block-2 +
a');
  const authorName = authorAnchor ?
authorAnchor.textContent.trim() : 'Unknown';
  const authorUrl = authorAnchor ? authorAnchor.href : '';
  // If you need sameAs: e.g. const sameAsLinks = [
'https://twitter.com/someone' ];

  //
  // --- 7. ARTICLE IMAGE (HERO) ---
  //
  // Example: main article image has class="article-photo"
  let heroImageUrl = '';
  let heroImageWidth = '';
  let heroImageHeight = '';
  const heroImageEl = document.querySelector('.article-photo');
  if (heroImageEl) {
    // Basic attributes
    heroImageUrl = heroImageEl.src || '';
    heroImageWidth = heroImageEl.getAttribute('width') || '';
    heroImageHeight = heroImageEl.getAttribute('height') || '';

    // If <img> has a 'srcset', parse the largest image
    if ((!heroImageWidth || !heroImageHeight) &&
heroImageEl.hasAttribute('srcset')) {
      const srcset = heroImageEl.getAttribute('srcset') || '';
      const entries = srcset.split(',').map(part =>
part.trim().split(' '));
      const sortedBySize = entries.sort((a, b) => parseInt(b[1],
10) - parseInt(a[1], 10));
      const largest = sortedBySize[0];
      if (largest) {
        heroImageUrl = largest[0];
        // Example ratio-based height, adjust as needed:
        heroImageWidth = largest[1];
        heroImageHeight = Math.round((heroImageWidth / 1600) *
840);
      }
    }
  }

  //
  // --- 8. ARTICLE SECTION & KEYWORDS ---
  //
  // E.g., your site might store "Article" in an <a
class="badge">Article</a>
  const badgeEl = document.querySelector('.badge');
  const articleSection = badgeEl ? badgeEl.textContent.trim() :
'';

  // For keywords, assume .sidebar-tags .tag
  const tags = document.querySelectorAll('.sidebar-tags .tag');
  const keywords = Array.from(tags).map(tag =>
tag.textContent.trim()).join(', ');

  //
  // --- 9. WORD COUNT & TIME REQUIRED ---
  //
  const wordCount = document.body.innerText.split(/\s+/).length;
  const timeRequired = ''; // or "PT5M" if you have logic to
estimate reading time

  //
  // --- 10. BUILD JSON-LD SCHEMA ---
  //
  const schema = {
    "@context": "https://schema.org",
    "@type": "Article",

    "headline": headline,
    "alternativeHeadline": alternativeHeadline,
    "description": metaDescription,
    "articleSection": articleSection,
    "keywords": keywords,

    "datePublished": datePublished,
    "dateModified": dateModified,
    "isAccessibleForFree": true,

    // The main article image
    "image": {
      "@type": "ImageObject",
      "url": heroImageUrl,
      "width": heroImageWidth,
      "height": heroImageHeight
    },

    // The page itself
    "mainEntityOfPage": {
      "@type": "WebPage",
      "@id": canonicalURL
    },
    "url": url,

    // Author
    "author": {
      "@type": "Person",
      "name": authorName,
      "url": authorUrl
    },

    // Publisher (Adjust name, logo URL, etc. to match your site)
    "publisher": {
      "@type": "Organization",
      "name": "Grocery Doppio",
      "logo": {
        "@type": "ImageObject",
        "url":
"https://cdn.prod.website-files.com/621ed1db4fa2cb50f54c2b64/62d58
1d10c442a16535b1ca0_grocery%20doppio%20PW%20Incisiv.png",
        "width": 758,
        "height": 136
      }
    },

    // Additional optional fields
    "wordCount": wordCount,
    "timeRequired": timeRequired,
    "speakable": {
      "@type": "SpeakableSpecification",
      "cssSelector": ["h1", ".post-body"]
    },

    // Potential Action (ReadAction)
    "potentialAction": {
      "@type": "ReadAction",
      "target": [
        {
          "@type": "EntryPoint",
          "urlTemplate": url
        }
      ]
    }
  };

  //
  // --- 11. RETURN SCHEMA (JSON-LD) TO SCREAMING FROG ---
  //
  const outputJson = JSON.stringify(schema, null, 2);
  return seoSpider.data(`<script
type="application/ld+json">\n${outputJson}\n</script>`);
```