```
// src/demo/SignalPreview.tsx

import React, { useEffect, useState } from "react";
import type { WnspFrame } from "../protocol/frameTypes";

interface SignalPreviewProps {
  frames: WnspFrame[];
  frameDurationMs?: number;
}

/**
 * Simple animated preview that cycles through frame colors,
 * simulating a flashing signal.
 *
 * NOTE: Uses wavelength as a label only; you can derive an
approximate
 * color for UI, or later map back to your LETTER_COLORS.
 */
export const SignalPreview: React.FC<SignalPreviewProps>
= ({
  frames,
  frameDurationMs = 200,
}) => {
  const [index, setIndex] = useState(0);

  useEffect(() => {
    if (!frames.length) return;
    setIndex(0);
    const interval = setInterval(() => {
      setIndex((prev) => (prev + 1) % frames.length);
```

```
    }, frameDurationMs);
    return () => clearInterval(interval);
  }, [frames, frameDurationMs]);

  if (!frames.length) {
    return (
      <div style={{ padding: "1rem" }}>
        <h3>Signal Preview</h3>
        <p>(No frames to preview)</p>
      </div>
    );
  }

  const current = frames[index];

  // Map wavelength to a rough gray-scale / hue
demonstration (simple placeholder).
  const normalized =
    (current.wavelengthNm - 380) / (740 - 380); // 0–1
  const brightness = 50 + normalized * 50; // 50–100

  return (
    <div style={{ padding: "1rem", border: "1px solid #444",
borderRadius: 8 }}>
      <h3>Signal Preview</h3>
      <div
        style={{
          width: "100%",
          height: 80,
          borderRadius: 8,
```

```
          border: "1px solid #222",
          marginBottom: "0.5rem",
          background: `hsl(${normalized * 300}, 80%, $
{brightness}%)`,
          transition: "background 0.15s linear",
        }}
      />
      <div style={{ fontSize: 12 }}>
        Frame {index + 1}/{frames.length} – wavelength:
{current.wavelengthNm} nm,
        intensity: {current.intensityLevel}, payloadBit:
{current.payloadBit}
      </div>
    </div>
  );
};
```