

```
document.getElementById('generateBtn').addEventListener('click', generateID);
document.getElementById('toggleSide').addEventListener('click', toggleSide);
document.getElementById('downloadBtn').addEventListener('click', downloadPNG);
document.getElementById('realIDCheck').addEventListener('change', (e) => {
    isRealID = e.target.checked;
    generateID();
});

let isFront = true;
let isRealID = true;
let uploadedPhoto = null; // Store the uploaded photo Data URL

// Handle photo upload
document.getElementById('photoUpload').addEventListener('change', (e) => {
    const file = e.target.files[0];
    if (file) {
        const reader = new FileReader();
        reader.onload = (event) => {
            uploadedPhoto = event.target.result; // Data URL
        };
        reader.readAsDataURL(file);
    }
});

function generateID() {
    const canvas = document.getElementById('idCanvas');
    const ctx = canvas.getContext('2d');
    ctx.clearRect(0, 0, canvas.width, canvas.height);

    const data = {
        fullName: document.getElementById('fullName').value,
        street: document.getElementById('street').value,
        cityStateZip: document.getElementById('cityStateZip').value,
        dlNumber: document.getElementById('dlNumber').value,
        dlClass: document.getElementById('dlClass').value,
        restrictions: document.getElementById('restrictions').value,
        endorsements: document.getElementById('endorsements').value,
        sex: document.getElementById('sex').value,
        height: document.getElementById('height').value,
        eyes: document.getElementById('eyes').value,
        hair: document.getElementById('hair').value,
        weight: document.getElementById('weight').value,
        dob: document.getElementById('dob').value,
    }
}
```

```
        issueDate: document.getElementById('issueDate').value,
        expDate: document.getElementById('expDate').value,
        donor: document.getElementById('donor').value
    };

    if (isFront) {
        drawFront(ctx, data);
    } else {
        drawBack(ctx, data);
    }
}

function drawFront(ctx, data) {
    ctx.fillStyle = '#FFFFFF';
    ctx.fillRect(0, 0, 1012, 637);
    ctx.strokeStyle = '#003366';
    ctx.lineWidth = 6;
    ctx.strokeRect(10, 10, 992, 617);

    ctx.globalAlpha = 0.1;
    ctx.fillStyle = '#000000';
    ctx.font = 'bold 100px Arial';
    ctx.fillText('ROCKY MTS', 200, 300);
    ctx.globalAlpha = 1;

    ctx.font = 'bold 16px Arial';
    ctx.fillStyle = '#000000';
    ctx.fillText('COLORADO', 20, 40);
    ctx.font = '12px Arial';
    ctx.fillText('DEPARTMENT OF REVENUE', 20, 60);

    if (isRealID) {
        ctx.fillStyle = '#FFD700';
        ctx.beginPath();
        ctx.arc(980, 20, 15, 0, Math.PI * 2);
        ctx.fill();
        ctx.fillStyle = '#000000';
        ctx.font = 'bold 10px Arial';
        ctx.fillText('* REAL ID', 950, 25);
    }

    // Draw uploaded photo if available
    if (uploadedPhoto) {
```

```

const img = new Image();
img.onload = () => {
    ctx.drawImage(img, 400, 50, 150, 200);
};
img.src = uploadedPhoto;
} else {
    // Fallback placeholder
    ctx.fillStyle = '#CCCCCC';
    ctx.fillRect(400, 50, 150, 200);
    ctx.fillStyle = '#000000';
    ctx.font = '12px Arial';
    ctx.fillText('No Photo', 450, 150);
}

ctx.font = 'bold 18px Arial';
ctx.fillText(`FULL NAME: ${data.fullName}`, 20, 100);
ctx.font = '14px Arial';
ctx.fillText(`ADDRESS: ${data.street}`, 20, 120);
ctx.fillText(`${data.cityStateZip}`, 20, 140);
ctx.fillText(`DL NO: ${data.dlNumber} CLASS: ${data.dlClass} REST: ${data.restrictions}`, 20,
180);
ctx.fillText(`END: ${data.endorsements} SEX: ${data.sex} HEIGHT: ${data.height}`, 20, 200);
ctx.fillText(`EYES: ${data.eyes} HAIR: ${data.hair} WT: ${data.weight}`, 20, 220);
ctx.font = 'bold 14px Arial';
ctx.fillStyle = '#FF0000';
ctx.fillText(`EXP: ${data.expDate}`, 20, 260);
ctx.fillStyle = '#000000';
ctx.fillText(`DATE OF BIRTH: ${data.dob} ISSUED: ${data.issueDate}`, 20, 240);

ctx.lineWidth = 2;
ctx.beginPath();
ctx.moveTo(400, 280);
ctx.lineTo(550, 280);
ctx.stroke();

ctx.font = '4px Arial';
ctx.globalAlpha = 0.5;
for (let i = 0; i < 100; i++) {
    ctx.fillText('COLORADO', i * 10, 10);
    ctx.fillText('COLORADO', i * 10, 627);
}
ctx.globalAlpha = 1;

```

```

ctx.font = '8px monospace';
const mrz = `@\\n\\x1e\\rANSI
636000080002DLDAQ${data.dlNumber}\\nDCS${data.fullName}\\nDDEN\\nDAC${data.fullName.
split(' ')[0]}\\nDDFN\\nDAD${data.fullName.split(' ').slice(1).join(
')}\\nDDGN\\nDC${data.dlClass}M\\nDCBNONE\\nDCDNONE\\nDBD${data.expDate.replace(/\\//g,
"")}\\nDBB${data.dob.replace(/\\//g, ")}\\nDBH${data.height.replace(/[^0-9]/g,
"')}\\nDAG${data.street}\\nDAI${data.cityStateZip.split(',')[0].trim()}\\nDAJCO\\nDAK${data.cityState
Zip.split(
')[2]}0000\\nDCFNONE\\nDCGUSA\\nDDC${data.eyes}\\nDDDNONE\\nDDGUSA\\nDDEU\\nDDJCH
\\nDDK1\\nDDL1\\nDDMNONE\\nDDN0\\nDDP0\\nDDQNONE\\nDDRNONE\\nDDSNONE\\nDDT0\\n
DDU0\\x1e\\r`;

const mrzLines = mrz.split('\\n');
mrzLines.forEach((line, index) => {
  ctx.fillText(line, 20, 580 + index * 10);
});
}

function drawBack(ctx, data) {
  ctx.fillStyle = '#FFFFFF';
  ctx.fillRect(0, 0, 1012, 637);
  ctx.strokeStyle = '#003366';
  ctx.lineWidth = 6;
  ctx.strokeRect(10, 10, 992, 617);

  ctx.font = 'bold 16px Arial';
  ctx.fillStyle = '#000000';
  ctx.fillText('COLORADO FLAG', 20, 40);

  ctx.font = 'bold 18px Arial';
  ctx.fillStyle = '#FF0000';
  ctx.fillText(`DONOR: ${data.donor}`, 20, 80);
  ctx.fillStyle = '#000000';
  ctx.font = '14px Arial';
  ctx.fillText(`CLASS: ${data.dlClass} REST: ${data.restrictions} END: ${data.endorsements}`,
20, 120);

  ctx.font = '10px Arial';
  ctx.fillText('THIS DOCUMENT IS COMPLIANT WITH THE REAL ID ACT', 20, 160);
  ctx.fillText('FOR DOMESTIC AIR TRAVEL AND FEDERAL FACILITIES', 20, 180);
  ctx.fillText('VISIT WWW.DPS.STATE.CO.US FOR MORE INFO', 20, 200);

  const barcodeData = generateBarcodeData(data);
  generateBarcode(barcodeData, (barcodeCanvas) => {

```

```

    ctx.drawImage(barcodeCanvas, 400, 250, 200, 100);
  });

  ctx.font = '6px Arial';
  ctx.fillStyle = '#888888';
  ctx.fillText('This card is not valid for identification if expired. Report lost/stolen to  
1-800-284-4062 (CDOR Hotline)', 20, 610);

  ctx.globalAlpha = 0.2;
  ctx.fillStyle = '#00FFFF';
  ctx.fillText('COLORADO', 800, 100);
  ctx.globalAlpha = 1;
}

function generateBarcodeData(data) {
  return `@\\n\\x1e\\rANSI
636000080002DLDAQ${data.dlNumber}\\nDCS${data.fullName}\\nDDEN\\nDAC${data.fullName.
split(' ')[0]}\\nDDFN\\nDAD${data.fullName.split(' ').slice(1).join(
')}\\nDDGN\\nDC${data.dlClass}M\\nDCBNONE\\nDCDNONE\\nDBD${data.expDate.replace(/\\//g,
"")}\\nDBB${data.dob.replace(/\\//g, " ")}\\nDBH${data.height.replace(/[\\^0-9]/g,
"")}\\nDAG${data.street}\\nDAI${data.cityStateZip.split('.')[0].trim()}\\nDAJCO\\nDAK${data.cityState
Zip.split(
')[2]}0000\\nDCFNONE\\nDCGUSA\\nDDC${data.eyes}\\nDDDNONE\\nDDGUSA\\nDDEU\\nDDJCH
\\nDDK1\\nDDL1\\nDDMNONE\\nDDN0\\nDDP0\\nDDQNONE\\nDDRNONE\\nDDSNONE\\nDDT0\\n
DDU0\\x1e\\r`;
}

function generateBarcode(text, callback) {
  const canvas = document.createElement('canvas');
  bwipjs.toCanvas(canvas, {
    bcid: 'pdf417',
    text: text,
    scale: 3,
    height: 10,
    width: 30,
    includetext: false,
    textalign: 'center',
  }, (err) => {
    if (err) {
      console.error(err);
    } else {
      callback(canvas);
    }
  })
}

```

```
});  
}  
  
function toggleSide() {  
    isFront = !isFront;  
    generateID();  
}  
  
function downloadPNG() {  
    const canvas = document.getElementById('idCanvas');  
    const link = document.createElement('a');  
    link.download = 'Colorado_REAL_ID.png';  
    link.href = canvas.toDataURL();  
    link.click();  
}
```