

September 19, 2025

<https://github.com/Setasign/fpdf> から fpdf.php を同じディレクトリ内に移動

```
mkdir -p /var/www/html/pic/uploads/output sudo chown -R www-data:www-data /var/www/html/pic/uploads sudo  
chmod -R 775 /var/www/html/pic/uploads
```

```
/your_project/  
├── main.php  
└── uploads/  
    └── output/ ← ここに変換後のファイルが保存される
```

```
<?php  
// 依存: PHP GD, imagick(推奨), FPDF (https://github.com/Setasign/fpdf)  
  
require('fpdf.php'); // FPDFのファイルが同じフォルダにある前提  
  
function convertImage($srcPath, $destPath, $destExt) {  
    $info = getimagesize($srcPath);  
    if (!$info) return false;  
    $mime = $info['mime'];  
  
    switch ($mime) {  
        case 'image/jpeg':  
            $img = imagecreatefromjpeg($srcPath);  
            break;  
        case 'image/png':  
            $img = imagecreatefrompng($srcPath);  
            break;  
        case 'image/gif':  
            $img = imagecreatefromgif($srcPath);  
            break;  
        default:  
            return false;  
    }  
  
    switch (strtolower($destExt)) {  
        case 'jpg':  
        case 'jpeg':  
            imagejpeg($img, $destPath, 90);  
            break;  
        case 'png':  
            imagepng($img, $destPath);  
            break;  
        case 'gif':  
            imagegif($img, $destPath);  
            break;  
        default:  
            imagedestroy($img);  
            return false;  
    }  
  
    imagedestroy($img);  
    return true;  
}  
  
function pdfToImages($pdfPath, $outputDir) {  
    // Imagickが必要  
    if (!class_exists('Imagick')) return false;  
  
    $imagick = new Imagick();  
    $imagick->setResolution(150, 150); // 解像度調整  
    $imagick->readImage($pdfPath);  
    $count = 0;  
    $files = [];
```

```

foreach ($imagick as $page) {
    $page->setImageFormat('png');
    $filename = $outputDir . '/page_' . $count . '.png';
    $page->writeImage($filename);
    $files[] = $filename;
    $count++;
}
$imagick->clear();
$imagick->destroy();

return $files; // 画像ファイルパス配列
}

function imageToPdf($imagePath, $pdfPath) {
    $pdf = new FPDF();
    $pdf->AddPage();

    list($width, $height) = getimagesize($imagePath);
    $w = $width > 190 ? 190 : $width * 0.75; // ページ幅に合わせて縮小
    $h = $height > 277 ? 277 : $height * 0.75;

    $pdf->Image($imagePath, 10, 10, $w, $h);
    $pdf->Output('F', $pdfPath);
    return file_exists($pdfPath);
}

function sanitizeFileName($filename) {
    return preg_replace('/[^a-zA-Z0-9_\.-]/', '_', $filename);
}

// --- アップロード処理 ---

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    if (!isset($_FILES['file']) || $_FILES['file']['error'] !== UPLOAD_ERR_OK) {
        echo "ファイルアップロードエラー";
        exit;
    }

    $uploadDir = __DIR__ . '/uploads';
    if (!is_dir($uploadDir)) mkdir($uploadDir);

    $tmpName = $_FILES['file']['tmp_name'];
    $origName = sanitizeFileName($_FILES['file']['name']);
    $ext = strtolower(pathinfo($origName, PATHINFO_EXTENSION));

    $targetExt = strtolower($_POST['target_format'] ?? '');

    if (!$targetExt) {
        echo "変換先フォーマットを選択してください。";
        exit;
    }

    $uploadedPath = $uploadDir . '/' . $origName;
    move_uploaded_file($tmpName, $uploadedPath);

    $outputDir = $uploadDir . '/output';
    if (!is_dir($outputDir)) mkdir($outputDir);

    if (in_array($ext, ['jpg', 'jpeg', 'png', 'gif'])) {
        if (in_array($targetExt, ['jpg', 'jpeg', 'png', 'gif'])) {
            // 画像→画像変換
            $outputFile = $outputDir . '/converted.' . $targetExt;
            if (convertImage($uploadedPath, $outputFile, $targetExt)) {
                echo "変換成功: <a href='uploads/output/converted.$targetExt' download>ダウンロード</a>";
            } else {
                echo "画像変換に失敗しました。";
            }
        } elseif ($targetExt === 'pdf') {
            // 画像→PDF

```

```

$outputFile = $outputDir . '/converted.pdf';
if (imageToPdf($uploadedPath, $outputFile)) {
    echo "PDF生成成功: <a href='uploads/output/converted.pdf' download>ダウンロード</a>";
} else {
    echo "PDF生成に失敗しました。";
}
} else {
    echo "サポートされていない変換です。";
}
} elseif ($ext === 'pdf') {
    if (in_array($targetExt, ['jpg', 'jpeg', 'png', 'gif'])) {
        // PDF→画像変換 (Imagick必要)
        $files = pdfToImages($uploadedPath, $outputDir);
        if ($files === false) {
            echo "PDF→画像変換にはImagickが必要です。";
            exit;
        }
        echo "PDFから画像変換成功:<br>";
        foreach ($files as $f) {
            $fileUrl = str_replace(__DIR__, '', $f);
            echo "<a href='uploads/output/" . basename($f) . "' download>" . basename($f) . "</a><br>";
        }
    } else {
        echo "サポートされていない変換です。";
    }
} else {
    echo "対応していないファイル形式です。";
}

exit;
}
?>

<!DOCTYPE html>
<html lang="ja">
<head>
<meta charset="UTF-8" />
<title>画像・PDF変換ツール</title>
</head>
<body>

<h1>画像・PDF変換ツール</h1>
<form action="" method="POST" enctype="multipart/form-data">
    <label>変換したいファイルを選択: <input type="file" name="file" required></label><br><br>
    <label>変換先フォーマット:
        <select name="target_format" required>
            <option value="">選択してください</option>
            <option value="jpg">JPEG</option>
            <option value="png">PNG</option>
            <option value="gif">GIF</option>
            <option value="pdf">PDF</option>
        </select>
    </label><br><br>
    <button type="submit">変換スタート</button>
</form>

</body>
</html>

```