


```

// 保存先のフォルダ名を抽出
var folderName = new Array();
folderName = saveObj2.split("/");

// alert(folderName[5]);

var sName = new Array();
sName = folderName[7].split(".");

saveJPEG(saveFolder+colorNum+"/"+sName[0]+"m-pv.jpg");
activeDocument.close(SaveOptions.DONOTSAVECHANGES);

}

/*////////////////////////////////////
    元画像を縦 160 にして remane
    //////////////////////////////////*/

//フォルダを選択
var folderObj2 = Folder("D:/_tori/Set_png/sd12c/pn01/"+colorNum+"/");

//保存先を選択
var saveObj2 = "D:/_tori/Set_jpeg/sd12c/pn01/s";
var saveFolder2 = saveObj2 + "/" ; //末尾に区切りのスラッシュが抜けているので足しておく

//indd ファイルを配列化（小文字・大文字無視）
var fileListJPEG2 = folderObj2.GetFiles("*.png") ;

//ファイル数分繰り返す
for (i = 0; i < fileListJPEG2.length ; i ++ )
{
    app.open(fileListJPEG2[i]);

// アクション実行
app.doAction("s160 縦 JPG キャンバスサイズ", "画像調整アクション 20200507");

var foldername = (new Folder(fileListJPEG2[i])); //作業フォルダ
//フォルダ名用に変数を用意する
var saveObjNameFolder2 = foldername;

```

```

//末尾に区切りのスラッシュが抜けているので足しておく
var saveObj2 = saveObjNameFolder2 + "/";

// 保存先のフォルダ名を抽出
var folderName = new Array();
folderName = saveObj2.split("/");

// alert(folderName[5]);

var sName = new Array();
sName = folderName[7].split(".");

// alert(sName[0]);

saveJPEG(saveFolder2+colorNum+"/"+sName[0]+"s-pv.jpg");
activeDocument.close(SaveOptions.DONOTSAVECHANGES);

}

////////////////////////////////////
function saveJPEG(jpegFileObj)
{
saveFileObj = new File(jpegFileObj);
jpegOpt = new JPEGSaveOptions();
jpegOpt.embedColorProfile = true;
jpegOpt.quality = 12; // クオリティ
jpegOpt.formatOptions = FormatOptions.OPTIMIZEDBASELINE; // ベースライン最適化
jpegOpt.matte = MatteType.NONE;
activeDocument.saveAs(saveFileObj,jpegOpt,true,Extension.LOWERCASE);
}

function setUnit(unt){ preferences.rulerUnits = unt; }
function getUnit(){ return preferences.rulerUnits; }

```