

```
#!/bin/bash
```

```
# Enable verbose mode
```

```
set -x
```

```
# Function to print usage
```

```
print_usage() {  
    echo "Usage: $0 <output_pdf> <directory1> [<directory2> ...]"  
    echo "Example: $0 all_code.pdf /path/to/project1 /path/to/project2"  
}
```

```
# Check if at least two arguments are provided
```

```
if [ $# -lt 2 ]; then  
    print_usage  
    exit 1  
fi
```

```
# Output PDF file name
```

```
OUTPUT="$1"  
shift
```

```
# Temporary PS file
```

```
TEMP_PS="temp.ps"
```

```
# Array of file extensions to include
```

```
EXTENSIONS=("swift" "c" "cpp" "h" "hpp" "py" "java" "js" "html" "css" "php" "rb" "go" "rs" "kt" "scala" "pl" "sh" "sql" "ts")
```

```
# Function to get the appropriate language for enscript
```

```
get_language() {  
    case "$1" in  
        swift) echo "swift" ;;  
        c|h|cpp|hpp) echo "c" ;;  
        py) echo "python" ;;  
        java) echo "java" ;;  
        js|ts) echo "javascript" ;;  
        html|css) echo "html" ;;  
        php) echo "php" ;;  
        rb) echo "ruby" ;;  
        go) echo "go" ;;  
        rs) echo "rust" ;;  
        kt) echo "kotlin" ;;  
        scala) echo "scala" ;;  
        pl) echo "perl" ;;  
        sh) echo "bash" ;;  
        sql) echo "sql" ;;  
        *) echo "text" ;;  
    esac  
}
```

```
# Clear the temporary PS file if it exists
```

```
> "$TEMP_PS"
```

```
# Process each directory
```

```
for dir in "$@"; do  
    if [ ! -d "$dir" ]; then  
        echo "Warning: $dir is not a valid directory. Skipping."  
        continue  
    fi
```

```
    echo "Processing directory: $dir"
```

```
    # Find and process files
```

```
    find "$dir" \( -type d \( -name node_modules -o -name vendor -o -name public \) -prune \) -o \( -type f \( -name "*.swift" -o -name "*.c" -o -name "*.cpp" -o -name "*.h" -o -name "*.hpp" -o -name "*.py" -o -name "*.java" -o -name "*.js" -o -name "*.html" -o -name "*.css" -o -name "*.php" -o -name "*.rb" -o -name "*.go" -o -name "*.rs" -o -name "*.kt" -o -name "*.scala" -o -name "*.pl" -o -name "*.sh" -o -name "*.sql" -o -name "*.ts" \) -print0 \) | while IFS= read -r -d '' file; do  
        echo "Found file: $file"  
        # Get the file extension  
        ext="${file##*.}"  
        # Get the appropriate language for enscript  
        lang=$(get_language "$ext")  
        echo "Processing $file as $lang"  
        # Use the appropriate syntax highlighting based on the file extension  
        if ! enscript -p - --highlight="$lang" --color=1 -fCourier8 --header="$file|Page \% of \%=" "$file" >>  
"$TEMP_PS"; then  
            echo "Warning: Failed to process $file"  
        else  
            echo "Successfully processed $file"  
        fi  
    done
```

```
# Check if any files were processed
```

```
if [ ! -s "$TEMP_PS" ]; then
```

/Users/abdulazizdot/Desktop/Fanzc/FanzcPage 2 of 2f.sh

```
    echo "Error: No files were found or processed. The output PDF will not be created."
    rm "$TEMP_PS"
    exit 1
fi

# Convert PostScript to PDF
if ps2pdf "$TEMP_PS" "$OUTPUT"; then
    echo "Successfully created PDF: $OUTPUT"
else
    echo "Error: Failed to create PDF"
    exit 1
fi

# Remove temporary PostScript file
rm "$TEMP_PS"

echo "Conversion complete. Output saved as $OUTPUT"

# Disable verbose mode
set +x
```