

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

```
# Initialize predefined folder and subfolders
```

```
initialize_folders() {  
    base_dir="MyData"  
    subfolders=("text" "images" "videos" "documents" "others")  
    mkdir -p "$base_dir"  
    for folder in "${subfolders[@]}; do  
        mkdir -p "$base_dir/$folder"  
    done  
}
```

```
# Function to display the main menu
```

```
show_menu() {  
    echo "-----"  
    echo " File Management System"  
    echo "-----"  
    echo "1. Create a file"  
    echo "2. Create a directory"  
    echo "3. List files and directories"  
    echo "4. Move or rename a file/directory"  
    echo "5. Search for a file"  
    echo "6. Delete a file"  
    echo "7. Delete a directory"  
    echo "8. Auto Organize Files"  
    echo "9. Exit"  
    echo "-----"  
    echo -n "Enter your choice: "  
}
```

Function to create a file

```
create_file() {  
    echo -n "Enter file name to create: "  
    read file_name  
    touch "$file_name" && echo "File '$file_name' created." echo "Failed to create file."  
}
```

Function to create a directory

```
create_directory() {  
    echo -n "Enter directory name to create: "  
    read dir_name  
    mkdir -p "$dir_name" && echo "Directory '$dir_name' created." echo "Failed to create directory."  
}
```

Function to list files and directories

```
list_items() {  
    echo "Listing files and directories:"  
    ls -l  
}
```

Function to move or rename a file/directory

```
move_or_rename() {  
    echo -n "Enter the current name of the file/directory: "  
    read current_name  
    echo -n "Enter the new name or location: "  
    read new_name  
    mv "$current_name" "$new_name" && echo "Moved/Renamed successfully." echo "Failed to  
move/rename."
```

```
}
```

```
# Function to search for a file
```

```
search_file() {
```

```
    echo -n "Enter the name of the file to search: "
```

```
    read search_name
```

```
    find . -name "$search_name" && echo "Search complete." echo "File not found."
```

```
}
```

```
# Function to delete a file
```

```
delete_file() {
```

```
    echo -n "Enter the name of the file to delete: "
```

```
    read file_name
```

```
    rm -i "$file_name" && echo "File '$file_name' deleted." echo "Failed to delete file."
```

```
}
```

```
# Function to delete a directory
```

```
delete_directory() {
```

```
    echo -n "Enter the name of the directory to delete: "
```

```
    read dir_name
```

```
    rm -ir "$dir_name" && echo "Directory '$dir_name' deleted." echo "Failed to delete directory."
```

```
}
```

```
# Function to auto organize files
```

```
auto_organize() {
```

```
    echo "Organizing files in the 'MyData' folder..."
```

```
    base_dir="MyData"
```

```
    declare -A extension_map=(["txt"]="text" ["jpg"]="images" ["png"]="images" ["mp4"]="videos"  
["pdf"]="documents" ["docx"]="documents")
```

```

for file in "$base_dir"/*; do
    if [[ -f "$file" ]]; then
        extension="${file##*.}"
        if [[ -n "${extension_map[$extension]}" ]]; then
            folder="${extension_map[$extension]}"
        else
            echo "Unknown file type detected: ${basename "$file"}"
            echo "Available types: ${subfolders[*]}"
            read -p "Enter the type for this file: " folder
            mkdir -p "$base_dir/$folder"
            extension_map["$extension"]="$folder"
        fi
        mv "$file" "$base_dir/$folder/"
        echo "Moved '${basename "$file"}' to '$folder/'"
    fi
done
echo "Auto-organization complete!"
}

```

Main program loop

initialize_folders

while true; do

show_menu

read choice

case \$choice in

1) create_file ;;

2) create_directory ;;

3) list_items ;;

```
4) move_or_rename ;;
5) search_file ;;
6) delete_file ;;
7) delete_directory ;;
8) auto_organize ;;
9) echo "Exiting File Management System."; exit 0 ;;
*) echo "Invalid choice! Please try again." ;;

esac

echo # Print a blank line for readability

done
```