

سوال 1: یک ساختار دایرکتوری مانند زیر ایجاد کنید (فقط با یک دستور mkdir):

```
project/  
├── assets/  
│   ├── css/  
│   └── py/  
└── pages/
```

```
hastishafiei@DESKTOP-VO4V8: ~
```

```
hastishafiei@DESKTOP-VO4V8:~$ mkdir -p project/assets/css project/assets/py project/pages  
hastishafiei@DESKTOP-VO4V8:~$
```

سوال 2: یک فایل متنی به نام info.txt بسازید. برای آن یک هارد لینک به نام info_h.txt و یک سیمبোলیک لینک به نام info_s.txt ایجاد کنید. سپس فایل info.txt را حذف کنید. چه اتفاقی برای دو لینک دیگر میافتد؟ (با cat محتوایشان را چک کنید.)

```
hastishafiei@DESKTOP-VO4V8: ~/project
```

```
hastishafiei@DESKTOP-VO4V8:~/project$ touch info.txt  
hastishafiei@DESKTOP-VO4V8:~/project$ ln info.txt info_h.txt  
hastishafiei@DESKTOP-VO4V8:~/project$ ln -s info.txt info_s.txt  
hastishafiei@DESKTOP-VO4V8:~/project$ rm info.txt  
hastishafiei@DESKTOP-VO4V8:~/project$ cat info_h.txt  
this is a message in a info.txt  
hastishafiei@DESKTOP-VO4V8:~/project$ cat info_s.txt  
cat: info_s.txt: No such file or directory  
hastishafiei@DESKTOP-VO4V8:~/project$
```

هارد لینک: هنوز کار می‌کند چون مستقیماً به همان داده اشاره دارد.

سیمبোলیک لینک: خراب می‌شود چون فقط به مسیر فایل اشاره می‌کند و آن مسیر دیگر وجود ندارد.

سوال 3: آپشن دستور rm که قبل از حذف هر فایل از شما تایید می‌خواهد چیست؟ و آپشنی که این تایید را غیرفعال می‌کند (force) چیست؟ (از man استفاده کنید.)

```
hastishafiei@DESKTOP-VO4V8: ~/project
```

```
-f, --force  
    ignore nonexistent files and arguments, never prompt  
  
-i    prompt before every removal
```

آپشنی که تایید را غیرفعال (force) می‌کند: rm -f

آپشنی که قبل از حذف هر فایل تایید می‌خواهد: rm -i

```
hastishafiei@DESKTOP-VO4V8: ~/project
```

```
hastishafiei@DESKTOP-VO4V8:~/project$ rm -i hi.txt  
rm: remove regular empty file 'hi.txt'?
```