

یک فایل متنی به نام `oslabfile1.txt` با دستور `touch` ایجاد کردیم.

```
mahtab@mahtab-vm:~$ touch oslabfile1.txt
mahtab@mahtab-vm:~$ ls
Desktop  Downloads  Music  oslabfile1.txt  Public  Templates
Documents  fl          nl      Pictures        snap    Videos
```

و در آن یک متن دلخواه با دستور `cat` می‌نویسیم.

```
mahtab@mahtab-vm:~$ cat > oslabfile1.txt
oslab1 testing
mahtab@mahtab-vm:~$ cat oslabfile1.txt
oslab1 testing
```

Inode number : Inode در واقع مخفف کلمه های `index node` است. `Index Node` ها ساختارهای داده ای هستند که شامل اطلاعاتی در خصوص فایل هایی هستند که توسط فایل سیستم ایجاد می شود. خود `inode` ها در جدولی به نام `inode table` ذخیره می شوند. هر `inode` ویژگی ها و مکان های بلوک دیسک داده های `object` را ذخیره می کند. اطلاعاتی که از هر فایل در `index node` مربوط به آن فایل ذخیره می شود شامل اشاره گر به آدرس محل فیزیکی ذخیره فایل ، مالک و گروه فایل ، قوانین دسترسی به فایل ، اندازه فایل ، زمان آخرین دسترسی ، زمان آخرین تغییر در فایل و زمان آخرین تغییر در `inode` فایل است.

حال با دستور `ls -li oslabfile1.txt` ، `inode number` فایل را به دست می‌آوریم:

```
mahtab@mahtab-vm:~$ ls -li oslabfile1.txt
260819 -rw-rw-r-- 2 mahtab mahtab 15 Dec 13 05:52 o
slabfile1.txt
```

با دستور `ln` میان دو فایل `oslabfile1.txt` و `oslabfile2.txt` یک لینک ایجاد می‌کنیم:

```
mahtab@mahtab-vm:~$ ln oslabfile1.txt oslabfile2.txt
```

`Inode number` فایل `oslabfile2.txt` را پیدا می‌کنیم:

```
mahtab@mahtab-vm:~$ ls -li oslabfile2.txt
260819 -rw-rw-r-- 2 mahtab mahtab 15 Dec 13 05:52 oslabfile2.txt
```

جواب سوال 1) برابر است با : 260819

جواب سوال 2) مقادیر `inode number` هر دو فایل یکسان است.

جواب سوال 3) محتوای هر دو فایل نیز یکسان است.

```
mahtab@mahtab-vm:~$ cat oslabfile2.txt
oslab1 testing
```

جواب سوال 4) پس از ویرایش محتوای فایل oslabfile2.txt می بینیم که محتوای این دو فایل یکسان است.

```
mahtab@mahtab-vm:~$ nano oslabfile2.txt
mahtab@mahtab-vm:~$ cat oslabfile2.txt
oslab1 edited!!!
mahtab@mahtab-vm:~$ cat oslabfile1.txt
oslab1 edited!!!
```

جواب سوال 5) بله وجود دارد.

```
mahtab@mahtab-vm:~$ rm oslabfile1.txt
mahtab@mahtab-vm:~$ ls
Desktop  Downloads  Music  oslabfile2.txt  Public  Templates
```

جواب سوال 6) strace یک ابزار تشخیصی ، debugging است که از آن برای نظارت و تغییرات در تعاملات بین فرآیندها و کرنل لینوکس استفاده می شود که شامل system call ها ، signal deliveries و تغییر وضعیت فرآیند است. هر سیستمی که توسط یک فرمان ساخته شده است را رهگیری و ضبط می کند. علاوه بر این، هر سیگنال لینوکس ارسال شده به فرآیند را نیز ضبط می کند.

جواب سوال 7) the **execve** syscalls

The **execve** is a syscall that executes the program referred to by the first argument.

با دستور touch فایل متنی oslabfile3 را ایجاد می کنیم و با دستور cat در آن یک محتوای دلخواه می نویسیم:

```
mahtab@mahtab-vm:~$ touch oslabfile3.txt
mahtab@mahtab-vm:~$ cat > oslabfile3.txt
oslab3 content
mahtab@mahtab-vm:~$ cat oslabfile3.txt
oslab3 content
```

حال با دستور زیر یک soft link به oslabfile3.txt ایجاد می کنیم:

```
mahtab@mahtab-vm:~$ ln -s oslabfile3.txt oslabfile4.txt
```

inode number های دو فایل oslabfile3.txt و oslabfile4.txt را با دستور زیر به دست می آوریم:

```
mahtab@mahtab-vm:~$ ls -li oslabfile*.txt
260844 -rw-rw-r-- 1 mahtab mahtab 15 Dec 15 10:21 oslabfile3.txt
260851 lrwxrwxrwx 1 mahtab mahtab 14 Dec 15 10:26 oslabfile4.txt -> oslabfile3.txt
```

جواب سوال 8) مقادیر inode number این دو فایل متفاوت است.

جواب سوال 9) با دستور nano محتوای oslabfile4 را تغییر می‌دهیم و می‌بینیم که محتوای oslabfile3 نیز تغییر یافته است.

```
mahtab@mahtab-vm:~$ nano oslabfile4.txt
mahtab@mahtab-vm:~$ cat oslabfile4.txt
!this is oslab4 edited content!
mahtab@mahtab-vm:~$ cat oslabfile3.txt
!this is oslab4 edited content!
```

جواب سوال 10) پس از حذف کردن oslabfile3.txt هنگامی که می‌خواهیم محتوای oslabfile4.txt را ویرایش کنیم ، آن را خالی از محتوا نمایش می‌دهد.

```
mahtab@mahtab-vm:~$ rm oslabfile3.txt
```

جواب سوال 11) لینک مانند نشانگر یک فایل است. همانطور که در زبان‌های برنامه نویسی pointer داریم ، لینک‌ها اشاره‌گرهایی هستند که به یک فایل یا یک دایرکتوری اشاره می‌کنند. ساختن لینک برای یک فایل شبیه به یک shortcut برای دسترسی به فایل است. دو نوع لینک داریم : soft link و hard link

Soft link : شبیه به ویژگی shortcut است که در سیستم عامل های ویندوز استفاده می شود. هر فایل پیوندی نرم حاوی یک مقدار Inode جداگانه است که به فایل اصلی اشاره می کند. مانند لینک های سخت، هر گونه تغییر در داده ها در هر یک از فایل ها در فایل دیگر منعکس می شود. پیوندهای نرم افزار را می توان در سیستم های فایل مختلف پیوند داد، اگرچه فایل اصلی حذف یا منتقل شود، فایل پیوند شده به درستی کار نمی کند که به نام hanging link می گویند. یک soft link می تواند به یک دایرکتوری پیوند دهد.

Hard link : به هر فایل hard link، همان مقدار Inode اصلی اختصاص داده می شود که به فایل اصلی اختصاص داده شده، بنابراین به همان مکان فایل فیزیکی اشاره می کند. Hard link ها انعطاف پذیرتر هستند و حتی اگر فایل های اصلی یا پیوند شده در سراسر سیستم فایل جابه جا شوند، پیوند باقی می ماند، اگرچه hard link ها قادر به عبور از فایل سیستم های مختلف نیستند.

جواب سوال 12) از مزایای استفاده از لینک می توان به access permission اشاره کرد برای مثال اگر بخواهیم دسترسی یک فایل را تغییر دهیم ، فقط مجوز دسترسی فایل اصلی (source file) را تغییر می دهیم درحالی که اگر فایل ها لینک نبودند و کپی بودند باید دسترسی هر کدام را تک به تک تغییر می دادیم. از دیگر کاربردهای لینک میتوان صرفه جویی در فضای دیسک ، مدیریت آسان ، ایجاد کردن shortcut و ... را نام برد.