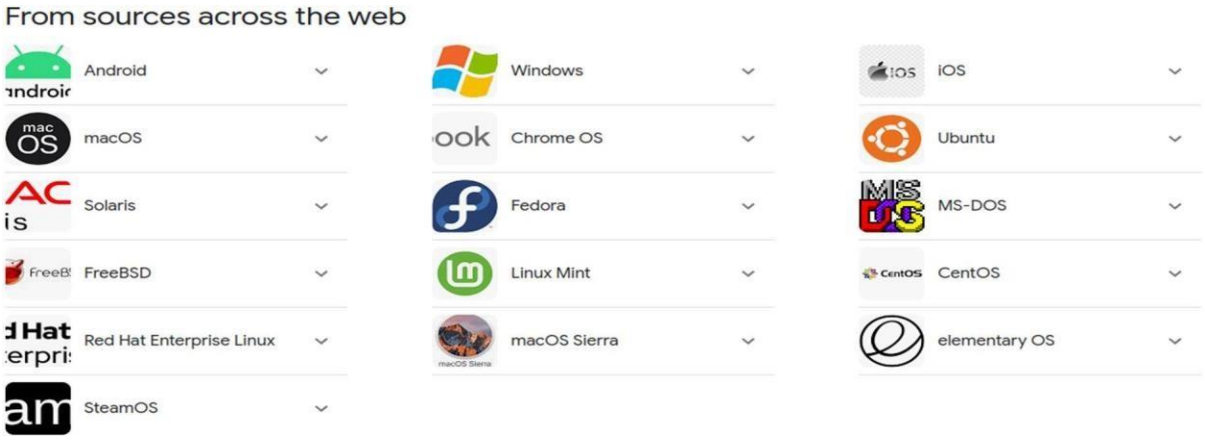
Experiment 1

Q1. Enlist Different Operating System that are available.
Ans:



Q2. Open-Source vs License Version which one is better and why?
Ans.
Open-Source Software:
Essentially, an open-source solution is a software distributed under a licensing agreement which permits the code to be shared and edited by other parties. This means that anyone who knows how can use and modify open-source software completely free of charge.

As a result, the software can evolve due to the many alterations of developers from around the world. And, often, it gets to the point where the final form is virtually unrecognizable from the original software. The benefit of open source is that this model produces an increasingly more diverse scope of design perspectives than a single company would ever be able to foster and sustain long-term. In other words, this is the perfect choice for anyone valuing innovation.

However, it can also leave users vulnerable to hackers.

Today, we have an organization called the Open-Source Initiative, which works to promote open- source software. Since its start in 1998, they have approved more than 80 open-source licenses worldwide. Interestingly, these licenses tend to fall into one of these categories: permissive licensesand copyleft licenses.

The most basic type of open-source license is the permissive license. With this one, you can do whatever you want with the software as long as you follow the general requirements, which are often phrased like this:

- You can do whatever you want with the code
- You must acknowledge the author
- However, you use it at your own risk

On the other hand, you have copyleft licenses. In addition to the requirements of a permissive license, they also require that you:

- Don't place any additional restrictions on the licensee's exercise of the license
- Make the source code for any binaries available
- The source code must be open and available under the same terms as which you got the code

Licensed Software:

Licensed software, on the other hand, is proprietary software distributed under a licensing agreement to authorized users only.

In other words, it's the complete opposite of open source, as the source code is not to be shared with the public for anyone to look at or modify.

Businesses are often defensive of their product and eager to preserve control of their brand, and the licensing agreements allow them to do this. Naturally, licensed software is the perfect opportunity for those looking for low-security risks, as dedicated developers are the only ones allowed to contribute to the software's code.

Main differences between Open Source and Licensed Software

Cost

Even though open-source software is technically free, there are long term costs associated with it such as implementation, innovation, support, and investing in the appropriate infrastructure as your organization progresses, technology evolves, and your requirements grow.

Additionally, it's becoming more common for open software providers to charge extra for add-ons, integration, and additional services. This can, in some instances, undo any cost-saving advantages you might have enjoyed.

The cost of licensed software, on the other hand, can vary considerably depending on the complexity of the solution you want. This can include a base fee for the software, integration, services and annual licensing fees. Though the hard cost can be higher, it's important to remember that you pay for a more customized product from a reliable name. Additionally, you'll also benefit from:

- Improved security
- Improved functionality
- Continuous innovation
- Greater scalability
- Ongoing training and support
- Lower requirement for technical skills

Support

In order to evolve, open-source software depends on a loyal and engaged online community providing support through forums and blogs.

Naturally, the response time of these communities is slower than dedicated support teams from well-known brands. This means that questions may go unanswered for some time, as there may not necessarily be any experts on hand. Additionally, there's no incentive for these communities to help – except for wanting to be cooperative.

The biggest advantage of licensed software is ongoing support, which can be imperative if you're a user without much technical skill. This support can include user manuals and points of contact for immediate assistance from experts who are closely acquainted with the product or service.

Security

As open-source software isn't developed in a controlled environment, we find that security is often a concern for many.

As the developers are situated all around the world, there is often a lack of continuity and shared direction that can counteract effective communication and collaboration. Additionally, as the software isn't always peer-reviewed or validated, a developer could potentially implant a backdoor Trojan into the software without the user being aware of it.

Naturally, this scares many off.

In comparison, licensed software tends to be perceived as the more secure option.

Unlike open-source software, a licensed solution is developed in a controlled environment by a focused team. This team of dedicated developers are the only people who can view or edit the source code, meaning that the product is heavily audited and the risk of backdoor Trojans is considerably diminished.

Practicality

As open-source software tends to accommodate the needs of developers rather than the majority of layperson users, the convenience and practicality of open source are frequently criticized.

Often, there are no user guides or manuals – as they are not a legal requirement – and when they are written, they tend to be written strictly for other developers. In other words, they're not written with the less technically experienced users in mind.

Expert usability testing has enabled licensed software to be more practical for a wider audience. User manuals are usually on hand for instant reference and swift training, and support services ensure that issues are solved quickly. Do you want to learn more about open-source software? Find our article on Open-Source Advantages right here.

2) How to choose between open source and licensed software?

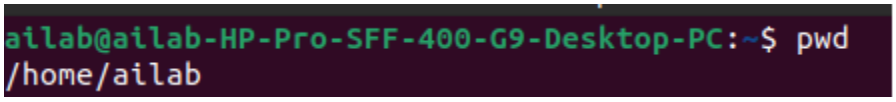
The pros and cons of open source and licensed software largely depend on your team's technical expertise – and the IT resources you have available.

Additionally, your choice will also depend on the needs and requirements of your business. Does the usefulness of a system which is completely free of cost, outweigh the running costs, security risks and lack of support that comes with it? If so, then it would be in your interest to join the growing trend of open-source software.

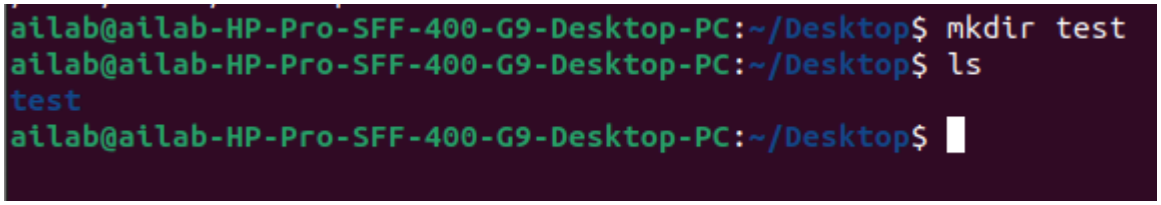
If you, on the other hand, are part of a large business with security concerns and a need for quick support, then you may be better suited to licensed software.

Q3. Use the following Linux commands.

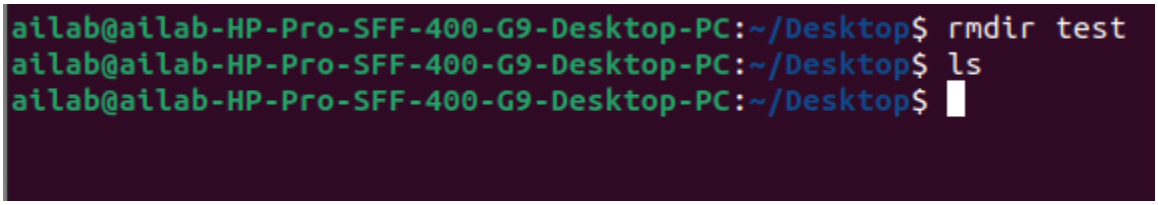
1. Pwd :



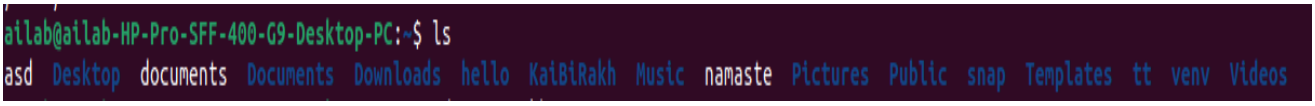
2. mkdir :



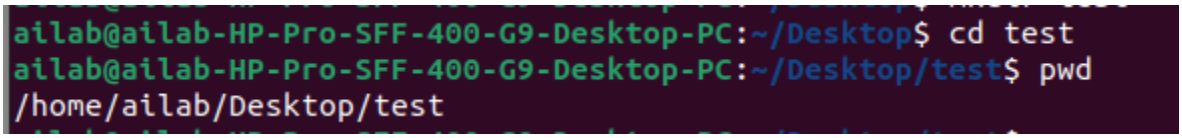
3. rmdir :



4. ls :



5. cd :



6. touch :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~$ touch first.txt
```

7. cat :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~$ cat first.txt
this is the first file.
```

8. rm :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~$ rm namuno.txt
```

9. cp :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~$ cp first.txt second.txt
```

10. mv :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~$ mv first.txt Desktop
```

11. head :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop$ head first.txt
this is the first file.

1
2
3
4
5
6
7
8
```

12. tail :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop$ tail first.txt
1
2
3
4
5
6
7
8
9
10
```

13. tac :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop$ tac first.txt
10
9
8
7
6
5
4
3
2
1

this is the first file.
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop$
```

14. wc :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop$ wc first.txt
12 15 46 first.txt
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop$
```

15. Ifconfig :

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop$ ifconfig
eno1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        ether 5c:60:ba:31:46:5e  txqueuelen 1000  (Ethernet)
        RX packets 32752  bytes 3451712 (3.4 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 736  bytes 130861 (130.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 19  memory 0x82480000-824a0000

enp2s0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 40:a6:b7:ac:cb:44  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device memory 0x82200000-822fffff

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 839  bytes 126964 (126.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 839  bytes 126964 (126.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlp0s20f3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.30.71.37  netmask 255.255.248.0  broadcast 10.30.71.255
        inet6 fe80::6514:c212:ea26:73ef  prefixlen 64  scopeid 0x20<link>
        ether d4:d8:53:5a:b3:83  txqueuelen 1000  (Ethernet)
        RX packets 1029  bytes 107091 (107.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 591  bytes 79906 (79.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Experiment 2

1:

```
#21BIT088
#to compare two number
echo "Enter the no a"
read d
echo "enter the no b"
read e
if [ $d -gt $e ]
then
echo "$d is greater than $e"
else
echo "$e is greater than $d"
fi
```

op:

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop/test$ ./practical2_1.sh
Enter the no a
12
enter the no b
32
32 is greater than 12
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop/test$
```

2:

```
#21BIT088
#triangle type
echo "Enter the length of triangle side 1"
read a
echo "Enter the length of triangle side 2"
read b
echo "Enter the length of triangle side 3"
```

```
read c
if [ $a -eq $b -a $b -eq $c ]then
echo "Triangle is equilateral triangle"
elif [ $a -eq $b -o $b -eq $c -o $c -eq $a ]
then
echo "Triangle is Isosceles Triangle"
else
echo "Triangle is scalene triange"
fi
```

op:

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop/test$ ./practical2_2.sh
Enter the length of triangle side 1
5
Enter the length of triangle side 2
5
Enter the length of triangle side 3
3
Triangle is Isosceles Triangle
```

3:

```
#21BIT088
#factorial
echo "Enter the number"
read num

fact=1
while [ $num -gt 1 ]
do
        fact=$((fact * num))
        num=$((num -1))
done
echo $fact
```

op:

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop/test$ ./practical2_3.sh
Enter the number
5
120
```

4:

```
#21BIT088
#fibonacci
echo "Enter the value of n"
read n
f=0
g=1
count=2
echo "Fibonacci series:"
echo $f
echo $g
while [ $count -le $n ]
do
fib=`expr $f + $g`
f=$g
g=$fib
echo $fib
count=`expr $count + 1`
done
```

op:

```
ailab@ailab-HP-Pro-SFF-400-G9-Desktop-PC:~/Desktop/test$ ./practical2_4.sh
Enter the value of n
4
Fibonacci series:
0
1
1
2
3
```

5:

```
#21BIT088
#to find palindrome
echo "Enter m"
read m
num=0
on=$m
while [ $m -gt 0 ]
do
num=$(expr $num \* 10)
k=$(expr $m % 10)
num=$(expr $num + $k)
m=$(expr $m / 10)
done
if [ $num -eq $on ]
then
echo palindrome
else
echo not palindrome
fi
```

op: