

# Oracle Database Express Edition

## 1)Создание таблиц:

### Клиенты

```
CREATE TABLE Clients (  
    ClientID INT PRIMARY KEY,  
    Name VARCHAR2(100),  
    Email VARCHAR2(100)  
);
```

### Машины

```
CREATE TABLE Cars (  
    CarID INT PRIMARY KEY,  
    Model VARCHAR2(100),  
    CarNumber VARCHAR2(9),  
    Status VARCHAR2(50)  
);
```

### Заказы

```
CREATE TABLE Orders (  
    OrderID INT PRIMARY KEY,  
    ClientID INT,  
    CarID INT,  
    OrderDate DATE,  
    ReturnDate DATE,  
    Status VARCHAR2(50),  
    CONSTRAINT fk_client FOREIGN KEY (ClientID) REFERENCES  
Clients(ClientID),  
    CONSTRAINT fk_car FOREIGN KEY (CarID) REFERENCES Cars(CarID)  
);
```

**2) Создание триггера, который будет обновлять статус машины в таблице "cars", когда в таблице "orders" будет создан новый заказ:**

```
CREATE OR REPLACE TRIGGER update_car_status
AFTER INSERT ON Orders
FOR EACH ROW
BEGIN
    UPDATE Cars
    SET Status = 'Занята'
    WHERE CarID = :NEW.CarID;
END;
```

**3) Создание процедуры для обновления статусов заказов и машин при завершении аренды машины:**

```
CREATE OR REPLACE PROCEDURE update_order_status_proc AS
BEGIN
    UPDATE orders
    SET status = 'закрыт'
    WHERE returndate <= SYSDATE;

    UPDATE cars
    SET status = 'свободна'
    WHERE carid IN (SELECT carid FROM orders WHERE returndate <= SYSDATE);
END;
```

**Создание планировщика заданий (job scheduler) для вызова процедуры в определенное время:**

```
BEGIN
    DBMS_SCHEDULER.CREATE_JOB (
        job_name      => 'UPDATE_ORDER_STATUS_JOB',
        job_type       => 'PLSQL_BLOCK',
        job_action     => 'BEGIN update_order_status_proc; END;',
        start_date     => SYSTIMESTAMP,
        repeat_interval => 'FREQ=HOURLY; BYMINUTE=0; BYSECOND=0',
        enabled        => TRUE
    );END;
```

#### 4) Наполнение таблиц данными:

##### CLIENTS:

```
INSERT INTO Clients (ClientID, Name, Email) VALUES (1, 'Данила', 'danila@example.com');
```

```
INSERT INTO Clients (ClientID, Name, Email) VALUES (2, 'Иван', 'Ivan@example.com');
```

```
INSERT INTO Clients (ClientID, Name, Email) VALUES (3, 'Кристина', 'kris@example.com');
```

	CLIENTID	NAME	EMAIL
1	1	Данила	danila@example.com
2	2	Иван	Ivan@example.com
3	3	Кристина	kris@example.com

##### CARS:

```
INSERT INTO Cars (CarID, Model, CarNumber, Status) VALUES (1, 'ВАЗ Приора', 'K104MK161', 'Свободна');
```

```
INSERT INTO Cars (CarID, Model, CarNumber, Status) VALUES (2, 'Рено Логан', 'A112XX61', 'Свободна');
```

```
INSERT INTO Cars (CarID, Model, CarNumber, Status) VALUES (3, 'Рено Логан', 'X167KX61', 'Свободна');
```

	CARID	MODEL	CARNUMBER	STATUS
1	1	ВАЗ Приора	K104MK161	Свободна
2	2	Рено Логан	A112XX61	Свободна
3	3	Рено Логан	X167KX61	Свободна

##### ORDERS:

```
INSERT INTO Orders (OrderID, ClientID, CarID, OrderDate, ReturnDate, Status)
```

```
VALUES (1, 3, 2, TO_DATE('2023-11-03', 'YYYY-MM-DD'), TO_DATE('2023-11-05', 'YYYY-MM-DD'), 'Открыт');
```

```
VALUES (2, 2, 1, TO_DATE('2023-11-03', 'YYYY-MM-DD'), TO_DATE('2023-11-30', 'YYYY-MM-DD'), 'Открыт');
```

```
VALUES (3, 1, 3, TO_DATE('2023-11-01', 'YYYY-MM-DD'), TO_DATE('2023-11-24', 'YYYY-MM-DD'), 'Открыт');
```

	ORDERID	CLIENTID	CARID	ORDERDATE	RETURNDATE	STATUS
1	1	3	2	03.11.23	05.11.23	Открыт
2	2	2	1	03.11.23	30.11.23	Открыт
3	3	1	3	01.11.23	24.11.23	Открыт

Тут же срабатывает триггер:

	CARID	MODEL	CARNUMBER	STATUS
1	1	ВАЗ Приора	K104MK161	Занята
2	2	Рено Logan	A112XX61	Занята
3	3	Рено Logan	X167KX61	Занята

Отчет об истории сдачи в аренду авто:

```
SELECT *  
FROM orders  
WHERE CarID = 1;
```

(Выбираем id интересующей нас машины и получаем отчет заказов с этой машиной)

Отчет о финансовой деятельности сервиса за период (для простоты будем считать, что стоимость сдачи в аренду на сутки одного автомобиля константа):

```
SELECT  
    OrderDate,  
    ReturnDate,  
    SUM((ReturnDate - OrderDate) * 100) AS TotalMoney  
FROM  
    Orders  
WHERE  
    OrderDate BETWEEN TO_DATE('2023-11-02', 'YYYY-MM-DD') AND TO_DATE('2023-11-03', 'YYYY-MM-DD')  
GROUP BY  
    OrderDate, ReturnDate;
```

	ORDERDATE	RETURNDATE	TOTALMONEY
1	03.11.23	05.11.23	200
2	03.11.23	30.11.23	2700

Отчет о клиентах: когда какие авто брал в аренду, сколько заплатил и т.д.

```

SELECT c.ClientID, c.Name AS ClientName, o.CarID, o.OrderDate, o.ReturnDate, SUM((o.ReturnDate -
o.OrderDate) * 100) AS TotalMoney

FROM Clients c

JOIN Orders o ON c.ClientID = o.ClientID

GROUP BY c.ClientID, c.Name, o.CarID, o.OrderDate, o.ReturnDate;

```

	CLIE...	CLIENTNAME	CARID	ORDERDATE	RETURNDATE	TOTALMONEY
1	1	Данила	1	01.12.23	12.12.23	1100
2	1	Данила	3	01.11.23	24.11.23	2300
3	2	Иван	1	03.11.23	30.11.23	2700
4	3	Кристина	2	03.11.23	05.11.23	200

**Отчет о текущем состоянии автопарка: сколько в аренде, сколько свободны, сколько на ТО и т.д.**

```

SELECT

    Status,

    COUNT(*) AS NumberOfCars

FROM

    Cars

GROUP BY

    Status;

```

	STATUS	NUMBEROFCARS
1	Занята	3
2	Свободна	1

### Подбор(поиск) авто по параметрам

Добавим еще один параметр в таблицу Cars:

```

ALTER TABLE Cars

ADD CarType VARCHAR2(50);

Изменим добавленные данные:

UPDATE Cars

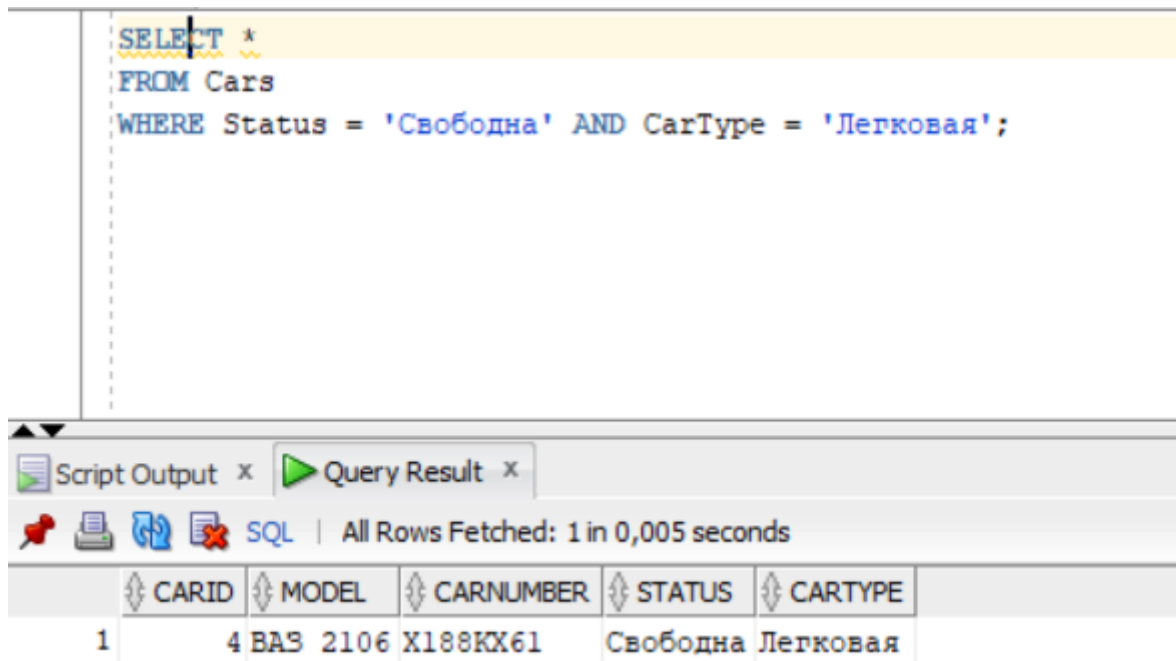
```

```
SET CarType = CASE
```

```
    WHEN CarID <> (SELECT MAX(CarID) FROM Cars) THEN 'Легковая'
```

```
    ELSE 'Грузовая'
```

```
END;
```

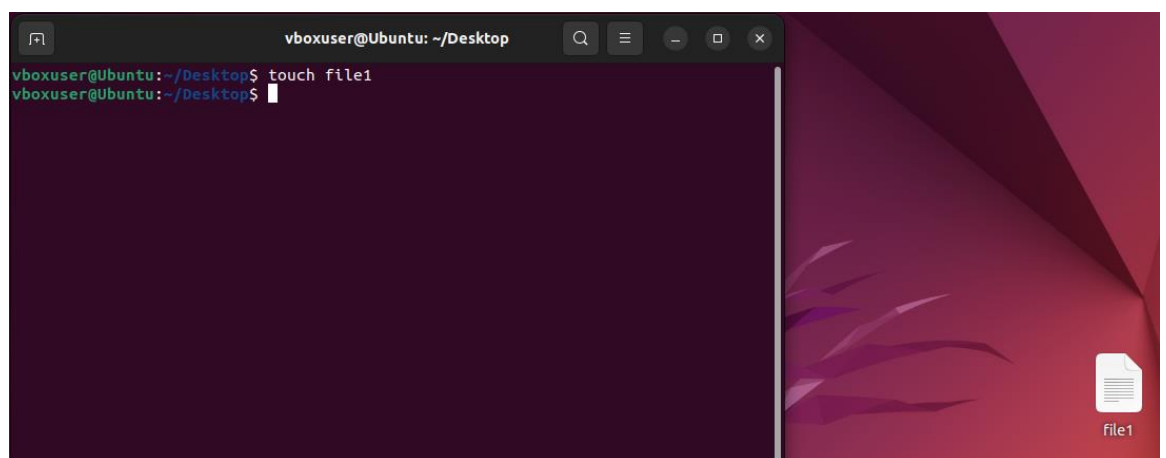


The screenshot shows a SQL query execution window. The query is: `SELECT * FROM Cars WHERE Status = 'Свободна' AND CarType = 'Легковая';`. The results are displayed in a table with 5 columns: CARID, MODEL, CARNUMBER, STATUS, and CARTYPE. One row is returned with the following values: 1, 4 BAS 2106, X188KX61, Свободна, and Легковая.

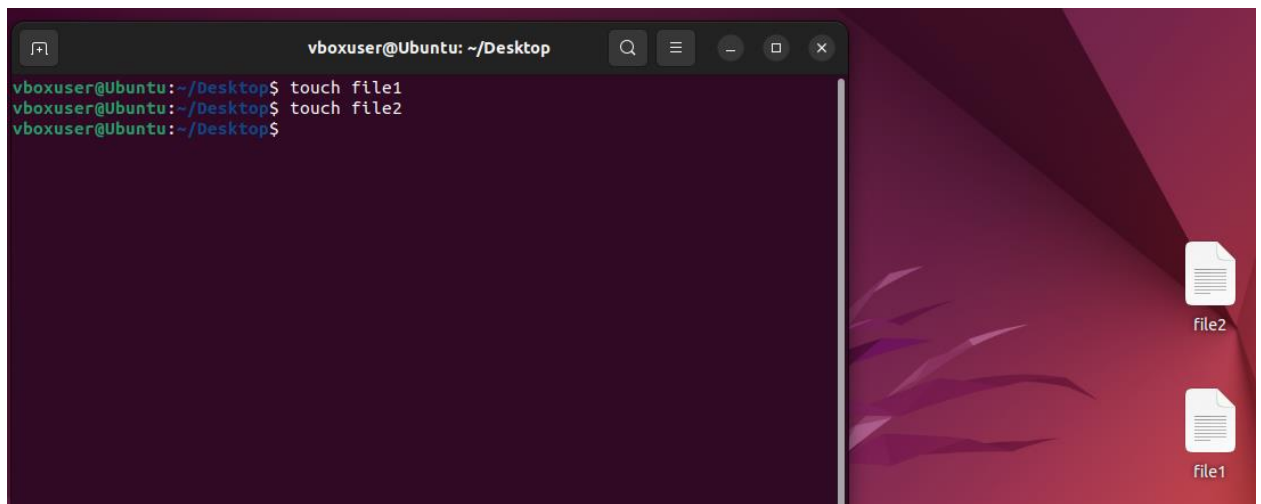
	CARID	MODEL	CARNUMBER	STATUS	CARTYPE
1	4	BAS 2106	X188KX61	Свободна	Легковая

## Ubuntu

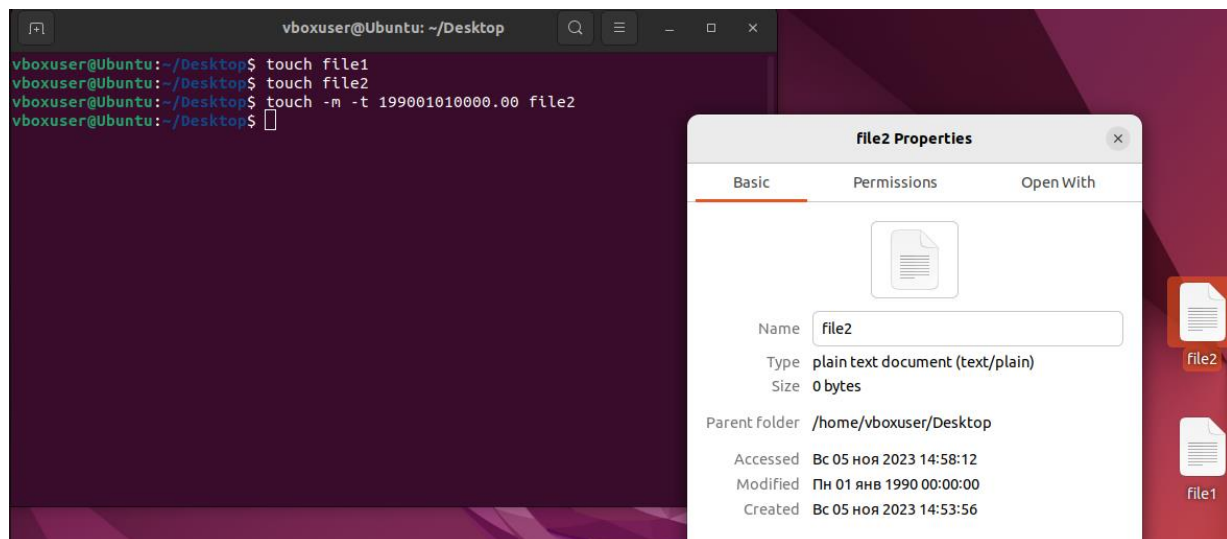
- a. Создайте файл file1 в текущей директории



- b. Создайте файл file2 в текущей директории

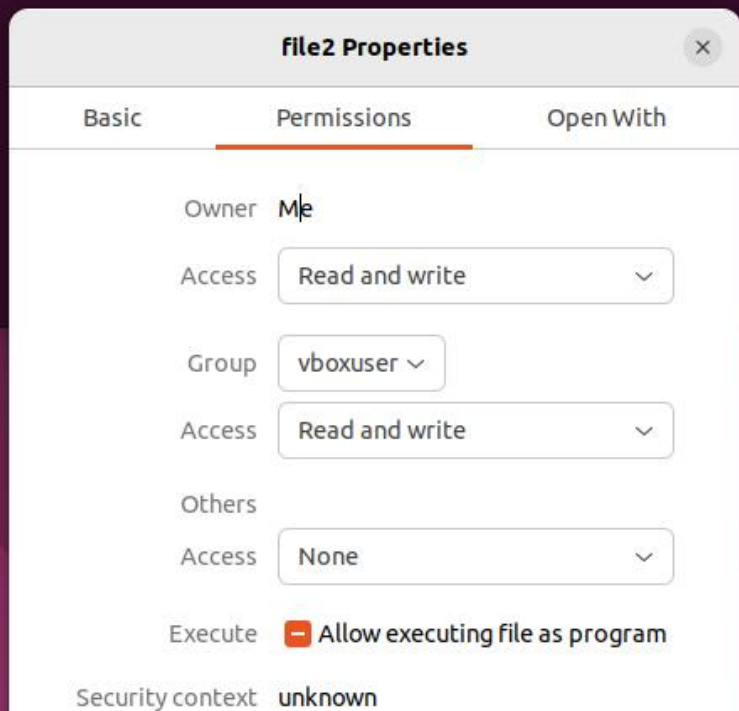


- c. **Задайте файлу file2 время модификации - 1 января 1990 года, 0 часов, 0 минут, 0 секунд**



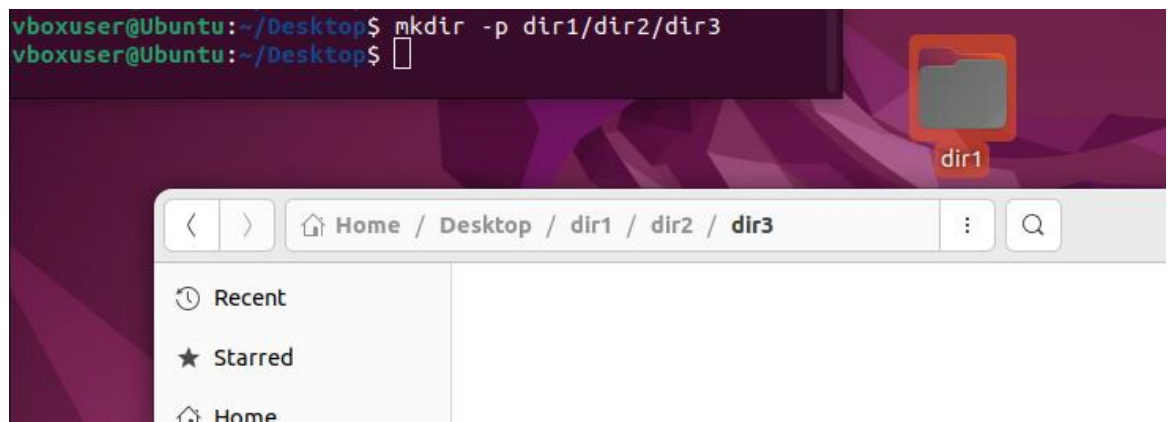
- d. **Задайте файлу file2 права доступа: rwxrw----**

```
vboxuser@Ubuntu:~/Desktop$ chmod 760 file2
vboxuser@Ubuntu:~/Desktop$
```



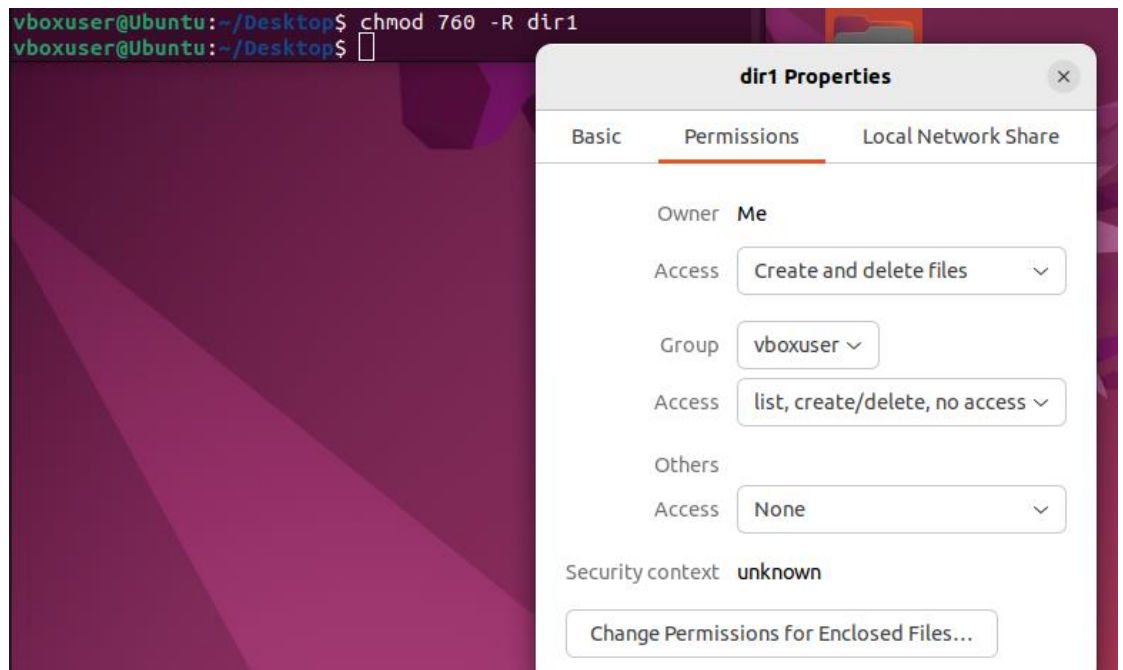
- d. Создайте вложенные директории `dir1/dir2/dir3/` в текущем каталоге

```
vboxuser@Ubuntu:~/Desktop$ mkdir -p dir1/dir2/dir3
vboxuser@Ubuntu:~/Desktop$
```

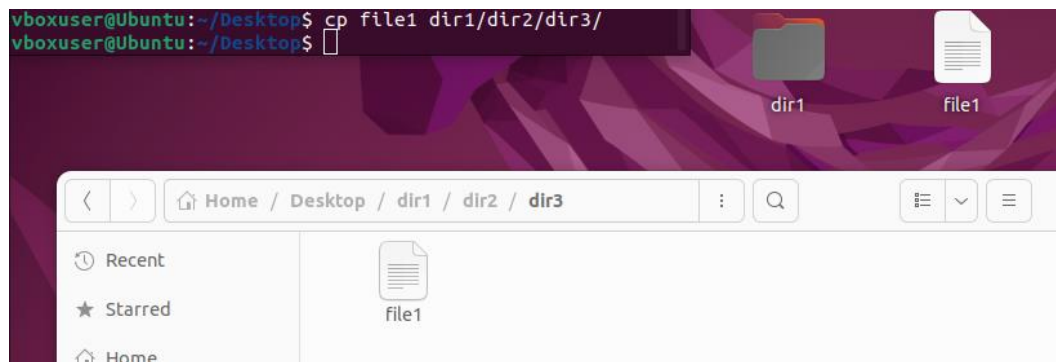


- f. Задайте цепочке вложенных директорий `dir1/dir2/dir3/` права доступа: `rwXrw----`

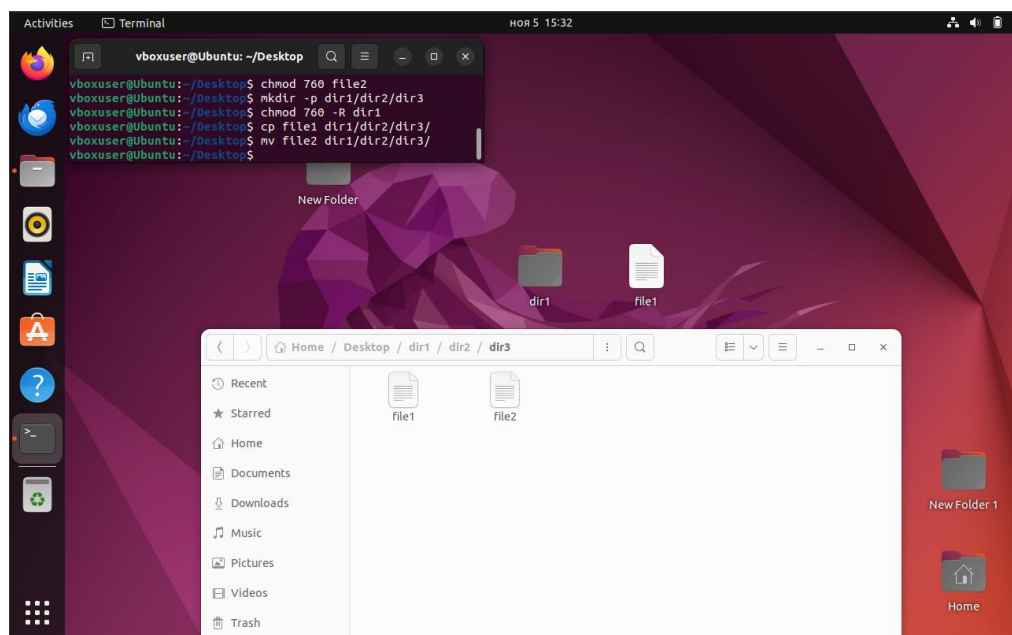




g. Скопируйте файл file1 в директорию dir1/dir2/dir3/



h. Переместите файл file2 в директорию dir1/dir2/dir3/



i. С помощью специальных команд Linux определите тип каталога dir1

```
vboxuser@Ubuntu:~/Desktop$ file dir1
dir1: directory
vboxuser@Ubuntu:~/Desktop$
```

- ж. С помощью специальных команд Linux выведите количество секунд, прошедших с 1 января 1970 года

```
vboxuser@Ubuntu:~/Desktop$ date +%s
1699188166
vboxuser@Ubuntu:~/Desktop$
```

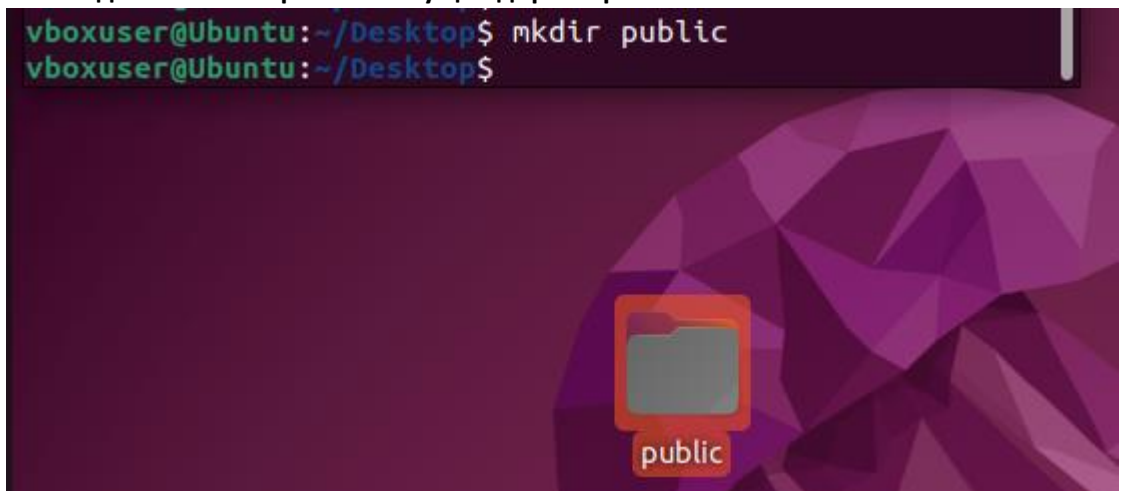
- к. Очистите текущий каталог от созданных файлов и каталогов

```
vboxuser@Ubuntu:~/Desktop$ rm -rf file1 dir1
vboxuser@Ubuntu:~/Desktop$
```

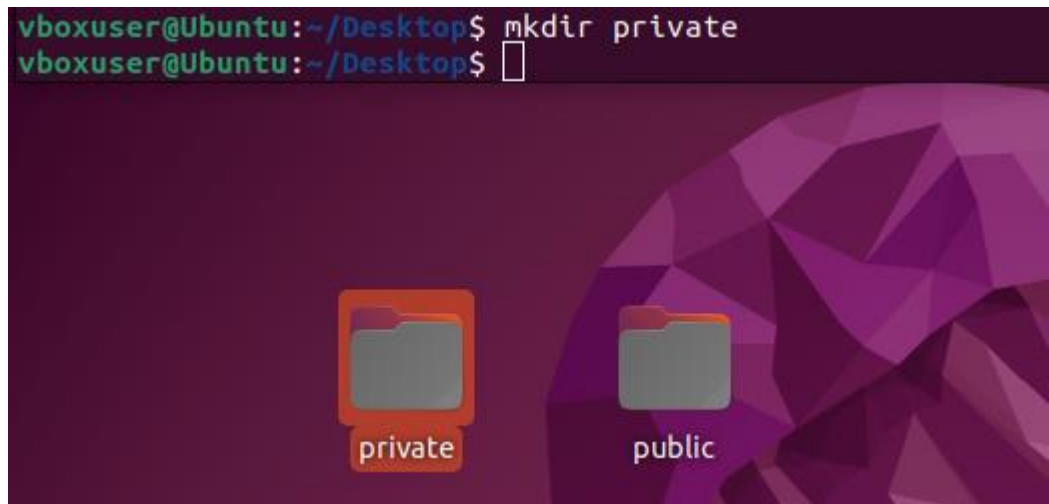


- а. Создайте каталог public в текущей директории

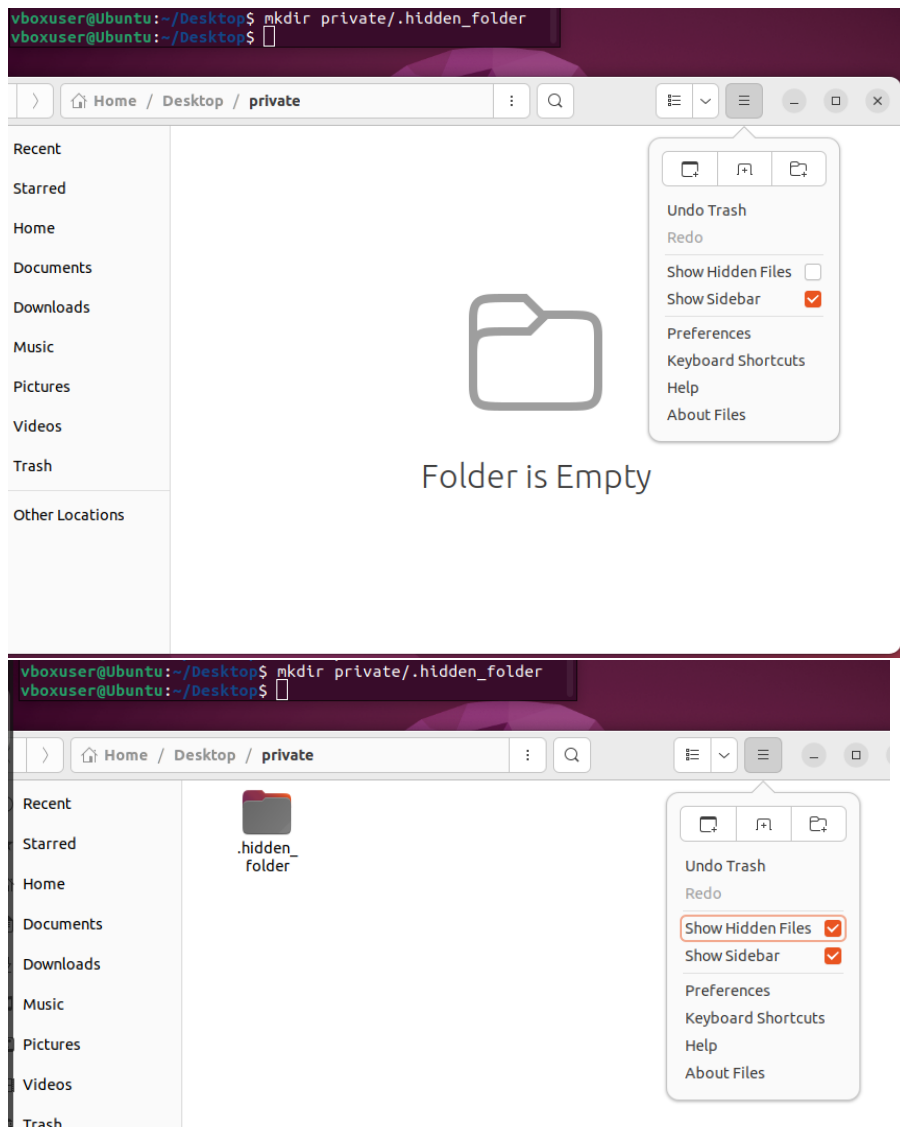
```
vboxuser@Ubuntu:~/Desktop$ mkdir public
vboxuser@Ubuntu:~/Desktop$
```



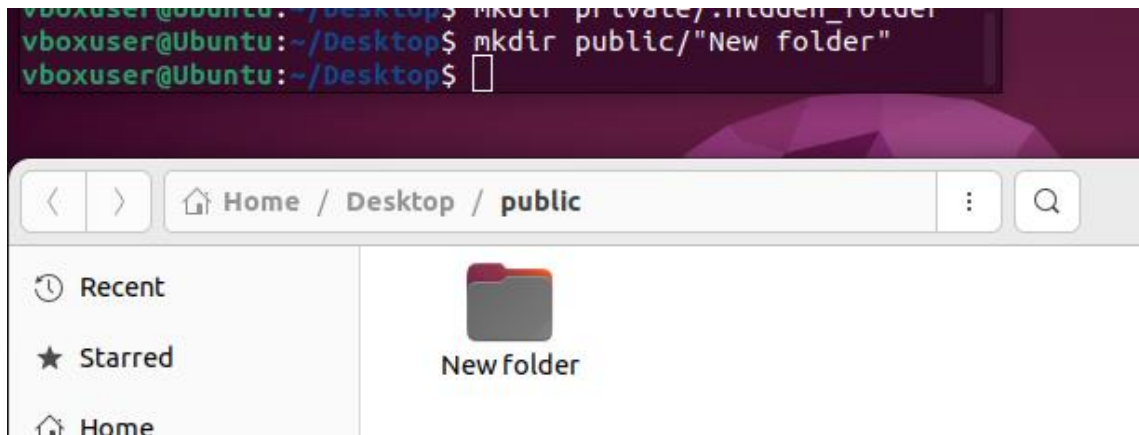
**б. Создайте каталог private в текущей директории**



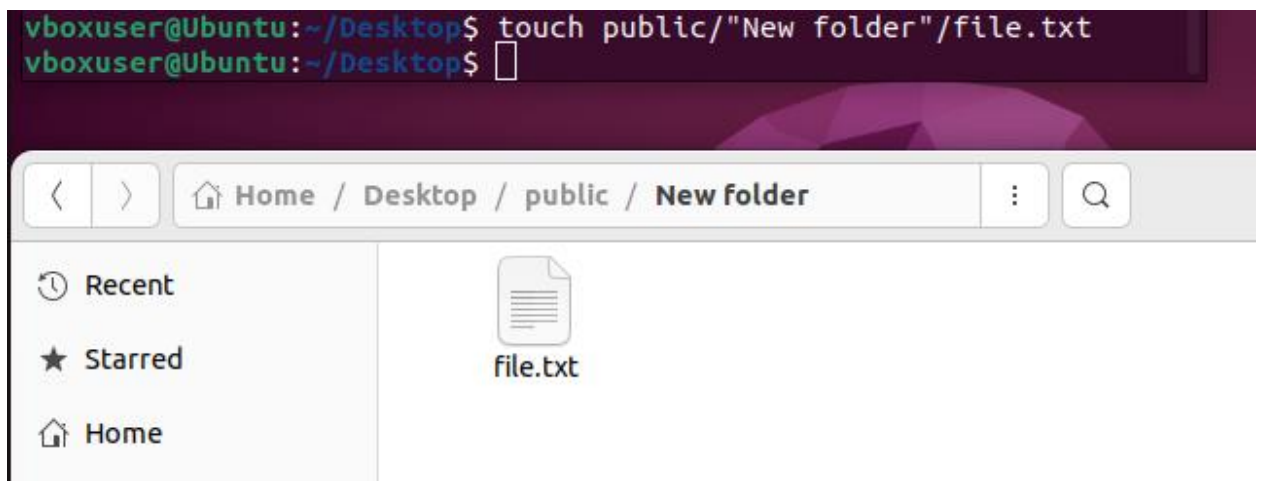
**в. Создайте в каталоге private скрытый каталог, имя может быть любым**



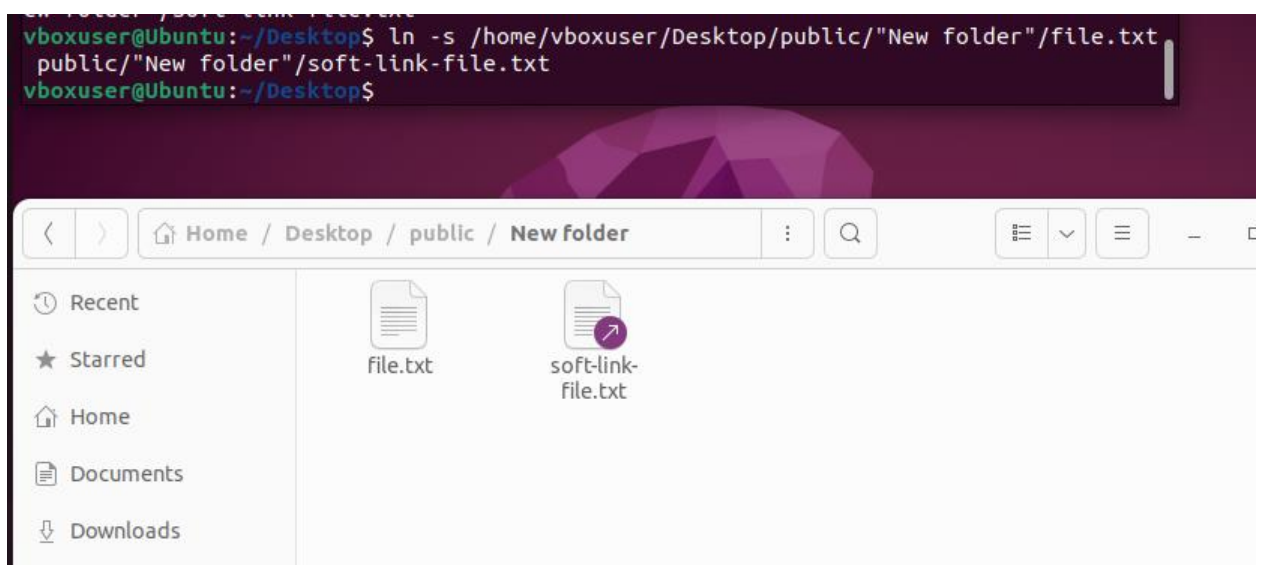
d. Создайте в каталоге public директорию с именем New folder



e. Создайте в каталоге New folder (из предыдущего шага) файл file.txt



f. Создайте мягкую ссылку soft-link-file.txt в той же директории New folder на файл file.txt

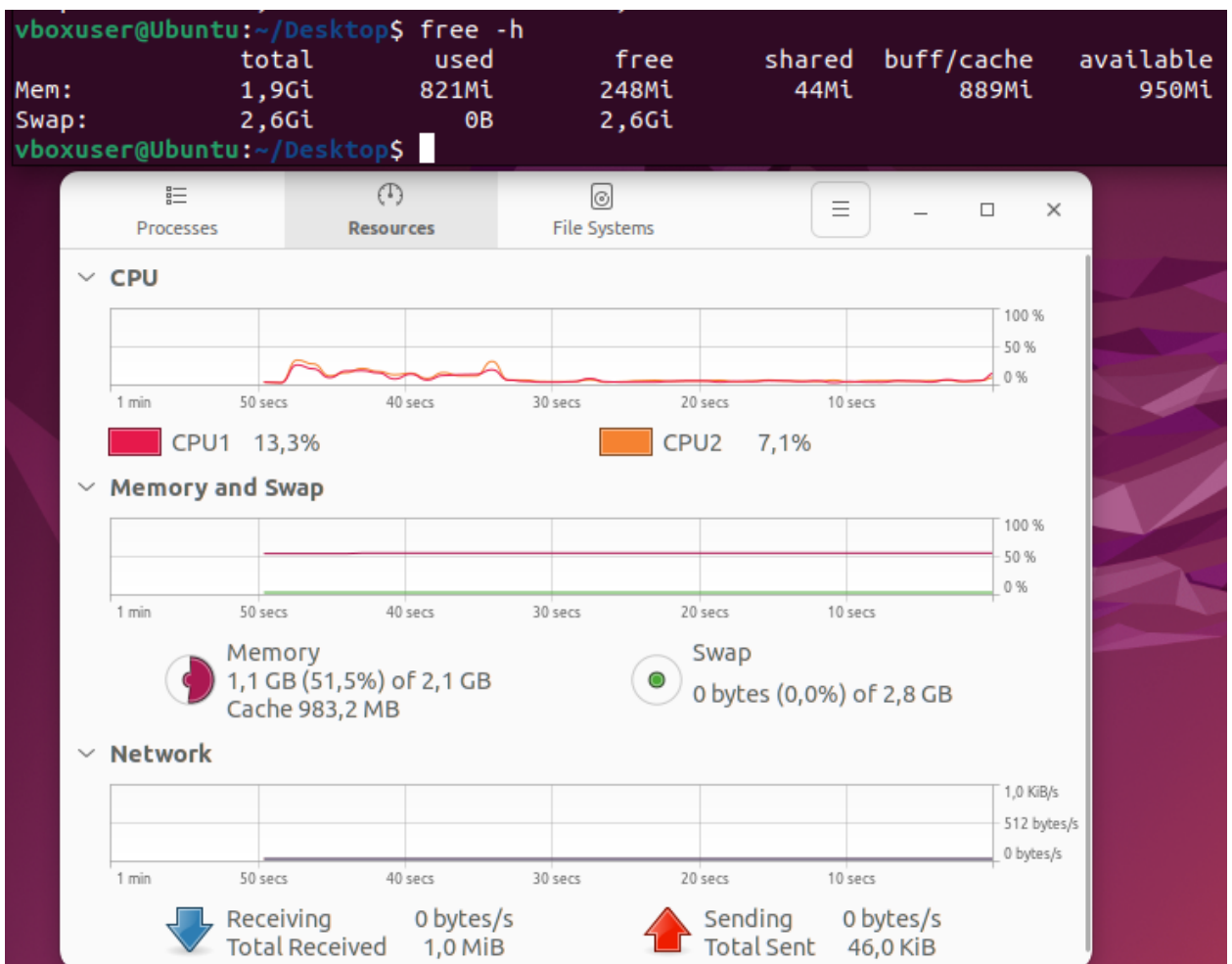


- g. Удалите все файлы и каталоги, созданные на предыдущих шагах

```
vboxuser@Ubuntu:~/Desktop$ rm -rf private public
vboxuser@Ubuntu:~/Desktop$
```



- а. Выяснить, сколько оперативной памяти доступно на компьютере. Сколько из них заняты, сколько свободны.



- b. Выясните, какие TCP порты открыты (прослушиваются). Адрес и порт должны отображаться в цифровом виде. Отображение не интернет портов считается ошибкой.

```
vboxuser@Ubuntu:~/Desktop$ ss -tln
State      Recv-Q      Send-Q      Local Address:Port      Peer Address:Port      Process
LISTEN     0            4096        127.0.0.53%lo:53        0.0.0.0:*
LISTEN     0            128         127.0.0.1:631           0.0.0.0:*
LISTEN     0            128         [::1]:631               [::]:*
```

- c. Выясните, какие порты открыты (прослушиваются). Адрес и порт должны отображаться в цифровом виде. Отображение не интернет портов считается ошибкой.

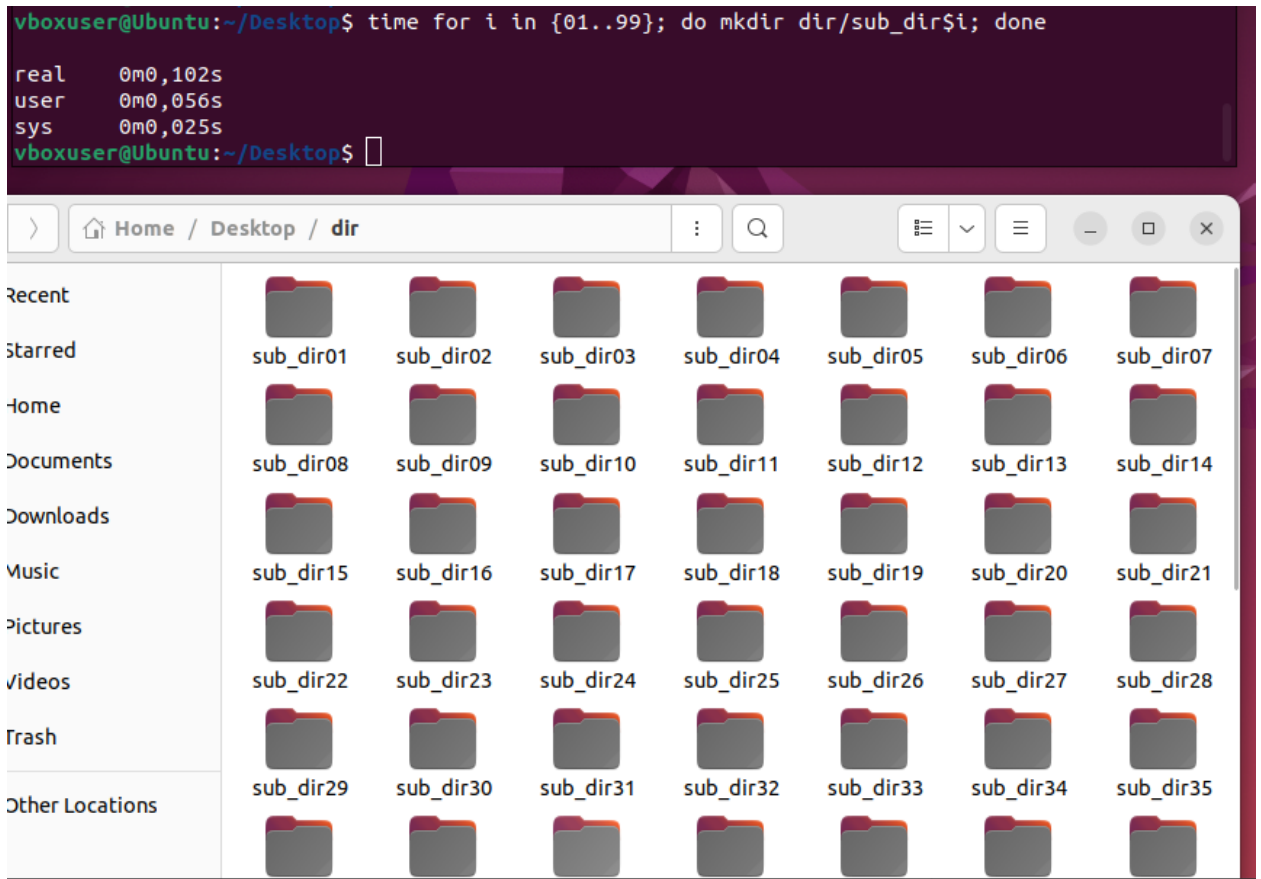
```
vboxuser@Ubuntu:~/Desktop$ ss -tuln
Netid      State      Recv-Q      Send-Q      Local Address:Port      Peer Address:Port      Process
udp        UNCONN     0            0            0.0.0.0:631              0.0.0.0:*
udp        UNCONN     0            0        127.0.0.53%lo:53        0.0.0.0:*
udp        UNCONN     0            0        0.0.0.0:46255           0.0.0.0:*
udp        UNCONN     0            0        0.0.0.0:5353            0.0.0.0:*
udp        UNCONN     0            0         [::]:57132              [::]:*
udp        UNCONN     0            0         [::]:5353               [::]:*
tcp        LISTEN     0            4096        127.0.0.53%lo:53        0.0.0.0:*
tcp        LISTEN     0            128         127.0.0.1:631           0.0.0.0:*
tcp        LISTEN     0            128         [::1]:631               [::]:*
```

- d. Создайте каталог dir в текущей директории

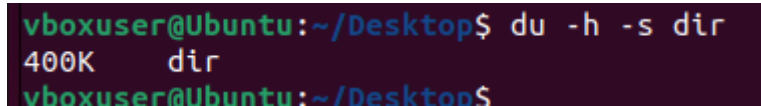


- e. Замерьте, сколько по времени выполняется следующая команда: Создание в каталоге dir 99 директорий с именами от sub\_dir01 до sub\_dir99





f. Выясните, сколько места на жёстком диске занимает каталог `dir` и всё его содержимое



g. Удалите все файлы и каталоги, созданные на предыдущих шагах

