

Übung: IPC mit Pipes

Diese Übung ist als **PDF-Dokument** abzugeben.

Aufgabe 1: Anonyme versus benannte Pipes

Recherchieren Sie im Internet den Unterschied zwischen "anonymen" und "benannten" (named) Pipes und erstellen Sie eine kurze Zusammenfassung mit Beispielen im Protokoll.

Aufgabe 2: Verwendung von Pipes

Für diese Übung werden einige Bilddaten in einem Unterverzeichnis in Ihrem Linux-Homeverzeichnis benötigt:

- `/home/<username>/images`

Führen Sie folgende Bash-Befehle auf einem Linux-Terminal aus und beschreiben Sie den Unterschied der Resultate:

```
ls -R ~/images/ | grep '\.jpg$'
ls -R ~/images/ | grep -ci '\.jpg$'
```

Handelt es sich hier um die Verwendung von anonymen oder benannten Pipes?

Aufgabe 3: Named Pipe

Erstellen Sie in einem Linux-Terminal eine "named pipe" mit dem Namen "my_fifo". Dokumentieren Sie deren Dateiattribute in Ihrem Protokoll.

Führen Sie danach folgenden Befehl in diesem Terminal aus:

```
cat my_fifo
```

Starten Sie ein zweites Linux-Terminal und führen folgenden Befehl aus:

```
echo "Hier wird eine Zeile in die Pipe geschickt" > my_fifo
```

Nun wird wieder zum ersten Terminal gewechselt und folgender Befehl ausgeführt:

```
tail -f my_fifo
```

Führen Sie nun im zweiten Terminal erneut folgenden Befehl aus:

```
echo "Hier wird eine Zeile in die Pipe geschickt" > my_fifo
```

Öffnen Sie ein drittes Linux-Terminal und führe dort folgenden Befehl aus:

```
echo "Hier wird nun eine weitere Zeile in die Pipe geschickt" >
my_fifo
```

Was fällt dabei auf?

Beenden Sie den Befehl tail im ersten Terminal mit Ctrl+c und löschen Sie die named pipe "my_fifo" wieder.

Aufgabe 4: CSV-Filtern mit Pipes

Speichern Sie die Datei "SalesJan2009.csv" aus Moodle in Ihrem Linux-Homeverzeichnis und öffnen Sie sie mit einem Tabellenkalkulationsprogramm (z.B. LibreOffice).

Führen Sie folgende Befehle im Linux-Terminal aus (Vergleich mit dem geöffneten Tabellenkalkulationsprogramm):

```
cat ./SalesJan2009.csv | grep "Mastercard" | wc -l
cat ./SalesJan2009.csv | grep -i "adam" | wc -l
cat ./SalesJan2009.csv | grep -i 'adam\|Transaction_date' | cut -d "," -f1-5
```

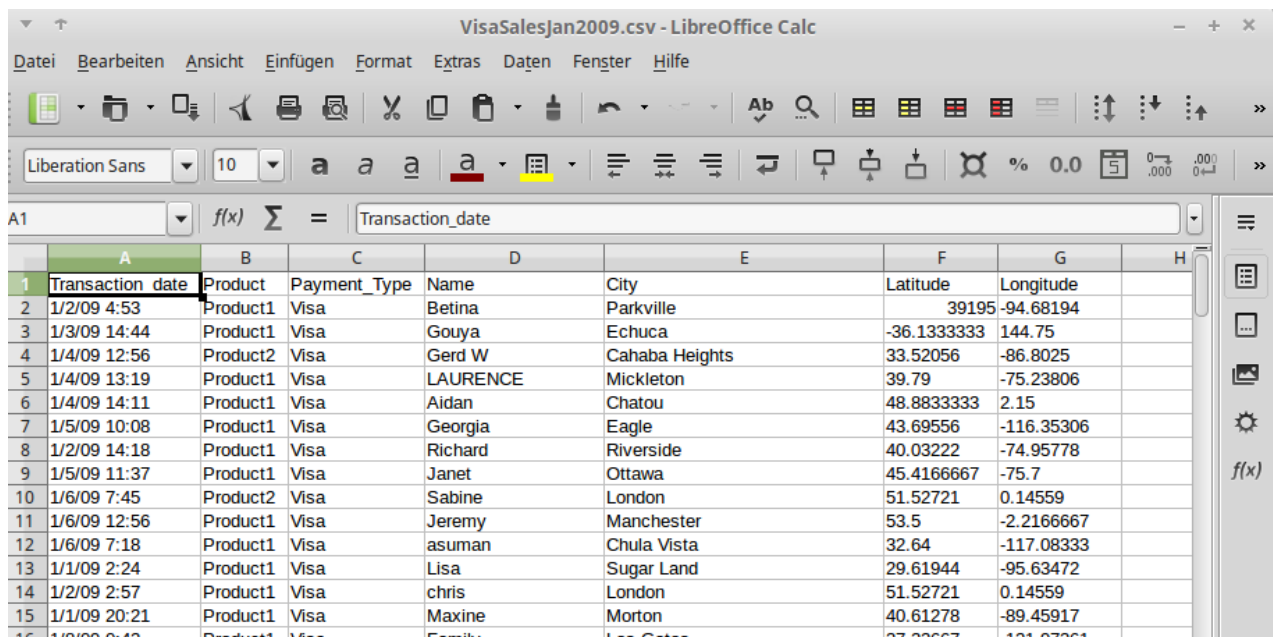
Um das Ergebnis eines Bash-Befehls abzuspeichern kann man die Ausgabe in eine Datei umleiten.

Mit folgendem Befehl werden von den gesuchten Einträgen die Spalten 1,2,5 und 8 in eine Datei namens "ergebnis.csv" umgeleitet.

```
cat ./SalesJan2009.csv | grep -i 'adam\|Transaction_date' | cut -d "," -f1,2,5,8 > ergebnis.csv
```

Verändern sie den Befehl so, daß alle Verkaufseinträge, die mit Visa bezahlt wurden, in eine Datei **VisaSalesJan2009.csv** geschrieben werden. Dabei sollen nur die Spalten **Transaction_date**, **Product**, **Payment_Type**, **Name**, **City**, **Latitude** und **Longitude** verwendet werden.

Kontrolle von "VisaSalesJan2009.csv" mit dem Tabellenkalkulationsprogramm:



	A	B	C	D	E	F	G	H
	Transaction_date	Product	Payment_Type	Name	City	Latitude	Longitude	
1	1/2/09 4:53	Product1	Visa	Betina	Parkville	39.195	-94.68194	
2	1/3/09 14:44	Product1	Visa	Gouya	Echuca	-36.1333333	144.75	
3	1/4/09 12:56	Product2	Visa	Gerd W	Cahaba Heights	33.52056	-86.8025	
4	1/4/09 13:19	Product1	Visa	LAURENCE	Mickleton	39.79	-75.23806	
5	1/4/09 14:11	Product1	Visa	Aidan	Chatou	48.8833333	2.15	
6	1/5/09 10:08	Product1	Visa	Georgia	Eagle	43.69556	-116.35306	
7	1/2/09 14:18	Product1	Visa	Richard	Riverside	40.03222	-74.95778	
8	1/5/09 11:37	Product1	Visa	Janet	Ottawa	45.4166667	-75.7	
9	1/6/09 7:45	Product2	Visa	Sabine	London	51.52721	0.14559	
10	1/6/09 12:56	Product1	Visa	Jeremy	Manchester	53.5	-2.2166667	
11	1/6/09 7:18	Product1	Visa	asuman	Chula Vista	32.64	-117.08333	
12	1/1/09 2:24	Product1	Visa	Lisa	Sugar Land	29.61944	-95.63472	
13	1/2/09 2:57	Product1	Visa	chris	London	51.52721	0.14559	
14	1/1/09 20:21	Product1	Visa	Maxine	Morton	40.61278	-89.45917	
15	1/9/09 0:42	Product1	Visa	Emily	Los Gatos	37.23667	-121.87261	