In [126…

```
using Pkg
# Pkg.add("DataFrames")
# Pkg.add("CSV")
# Pkg.add("Plots")
# Pkg.add("Statistics")
# Pkg.add("FreqTables")
using DataFrames
using Plots
using Statistics
using CSV
using Dates
using FreqTables
```

In [127…

```
df = DataFrame(CSV.File("transactions_cleaned.csv"))
```

177×6 DataFrame

152 rows omitted

Row	Date	Description	Amount	Transaction_Type	Category	Account_Name
	String15	String15	Float64	String7	String31	String7
1	12/15/2022	Transaction 1	31.25	debit	Shopping	A1
2	10/18/2022	Transaction 5	79.74	debit	Food_Dining	BCC2
3	9/20/2022	Transaction 6	81.69	debit	Food_Dining	BCC1
4	10/03/2022	Transaction 7	80.48	debit	Food_Dining	BCC2
5	9/10/2022	Transaction 8	62.17	debit	Food_Dining	BCC2
6	9/12/2022	Transaction 9	66.81	debit	Bills_Utilities	BCC1
7	12/08/2022	Transaction 10	4.99	debit	Shopping	CCC1
8	9/15/2022	Transaction 11	15.68	debit	Shopping	BCC1
9	11/29/2022	Transaction 12	32.31	debit	Shopping	CCC1
10	12/04/2022	Transaction 13	10.78	debit	Shopping	CCC1
11	11/21/2022	Transaction 14	17.01	debit	Shopping	BCC1
12	12/18/2022	Transaction 15	215.49	debit	Shopping	CCC1
13	12/11/2022	Transaction 20	8.19	credit	Shopping	CCC1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
166	11/15/2022	Transaction 293	1.5	debit	Food_Dining	C
167	11/15/2022	Transaction 294	11.25	debit	Food_Dining	BCC1
168	10/18/2022	Transaction 295	4.91	debit	Health_Fitness	BCC1
169	11/29/2022	Transaction 296	10.56	debit	Food_Dining	BCC1
170	12/05/2022	Transaction 297	9.26	debit	Shopping	BCC1
171	12/06/2022	Transaction 298	42.02	debit	Shopping	BCC1
172	12/10/2022	Transaction 299	9.26	credit	Shopping	BCC1
173	10/27/2022	Transaction 303	3.62	debit	Food_Dining	C
174	12/04/2022	Transaction 304	9.33	debit	Food_Dining	CCC1
175	9/19/2022	Transaction 305	69.0	debit	Shopping	BCC1

Row	Date	Description	Amount	Transaction_Type	Category	Account_Name
	String15	String15	Float64	String7	String31	String7
176	10/12/2022	Transaction 309	43.56	debit	Business Services	BCC2
177	12/19/2022	Transaction 311	10.0	credit	Bills_Utilities	W

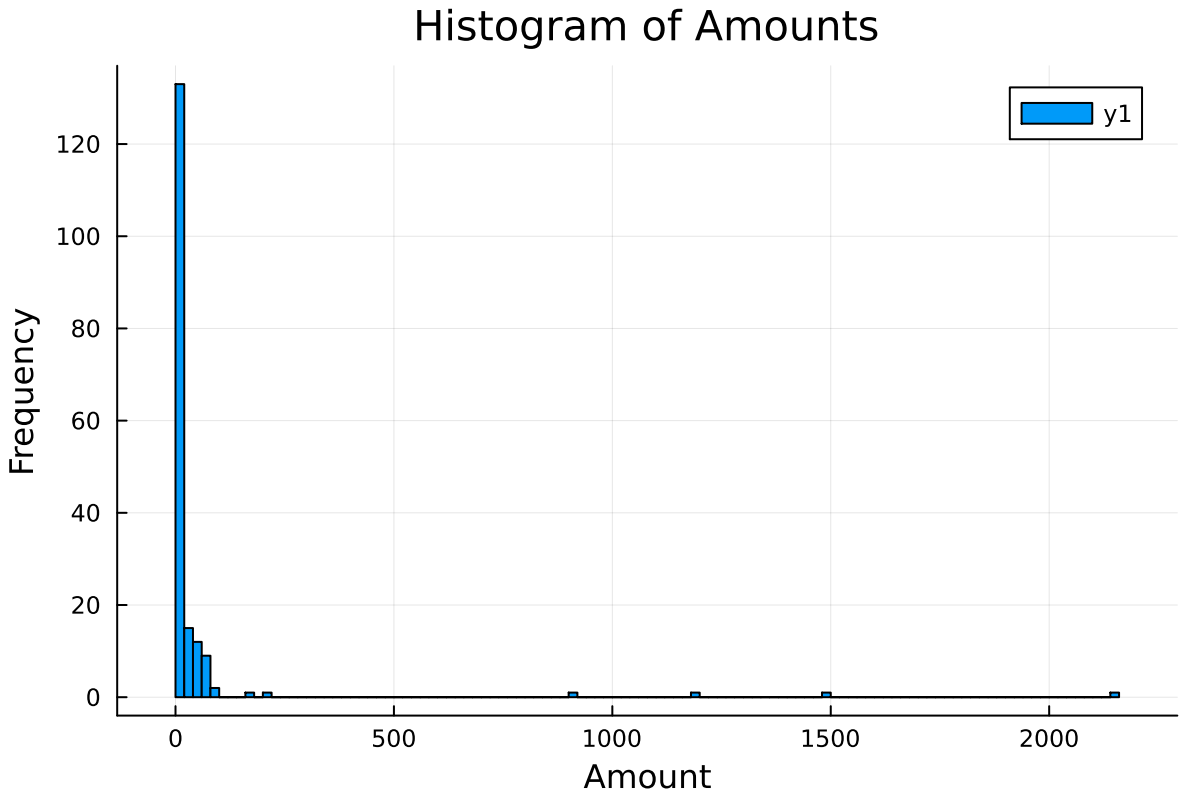
In [128…

```
println("Minimum: ", minimum(df.Amount), "\nMaximum: ", maximum(df.Amount), "\nMean: ", me
```

```
Minimum: 0.01
Maximum: 2151.0
Mean: 50.52248587570621
Median: 10.0
```

In [129…

```
# plot histogram over time
Plots.histogram(df.Amount, bins=100, title="Histogram of Amounts", xlabel="Amount", ylabel
```



In [130…

```
# convert Date column to DateTime
df.Date = Date.(df.Date, "m/d/y")
```

```
177-element Vector{Date}:
 2022-12-15
 2022-10-18
 2022-09-20
 2022-10-03
 2022-09-10
 2022-09-12
 2022-12-08
 2022-09-15
 2022-11-29
 2022-12-04
 ⋮
 2022-11-29
 2022-12-05
 2022-12-06
```

```
2022-12-10
2022-10-27
2022-12-04
2022-09-19
2022-10-12
2022-12-19
```

In [131…

```
# remove rows if they have B, C, W, A as they are debit accounts
df = filter!(row -> !(row.Account_Name in ["B", "C", "W", "A"]), df)


# remove rows if they have the category "Credit Card Payment"
df = filter!(row -> !(row.Category in ["Credit Card Payment", "Deposit", "Paycheck"]), df)
```

165×6 DataFrame

140 rows omitted

Row	Date	Description	Amount	Transaction_Type	Category	Account_Name
	Date	String15	Float64	String7	String31	String7
1	2022-12-15	Transaction 1	31.25	debit	Shopping	A1
2	2022-10-18	Transaction 5	79.74	debit	Food_Dining	BCC2
3	2022-09-20	Transaction 6	81.69	debit	Food_Dining	BCC1
4	2022-10-03	Transaction 7	80.48	debit	Food_Dining	BCC2
5	2022-09-10	Transaction 8	62.17	debit	Food_Dining	BCC2
6	2022-09-12	Transaction 9	66.81	debit	Bills_Utilities	BCC1
7	2022-12-08	Transaction 10	4.99	debit	Shopping	CCC1
8	2022-09-15	Transaction 11	15.68	debit	Shopping	BCC1
9	2022-11-29	Transaction 12	32.31	debit	Shopping	CCC1
10	2022-12-04	Transaction 13	10.78	debit	Shopping	CCC1
11	2022-11-21	Transaction 14	17.01	debit	Shopping	BCC1
12	2022-12-18	Transaction 15	215.49	debit	Shopping	CCC1
13	2022-12-11	Transaction 20	8.19	credit	Shopping	CCC1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
154	2022-09-22	Transaction 290	2.25	debit	Food_Dining	BCC2
155	2022-10-06	Transaction 291	3.25	debit	Food_Dining	BCC2
156	2022-09-22	Transaction 292	3.0	debit	Food_Dining	BCC2
157	2022-11-15	Transaction 294	11.25	debit	Food_Dining	BCC1
158	2022-10-18	Transaction 295	4.91	debit	Health_Fitness	BCC1
159	2022-11-29	Transaction 296	10.56	debit	Food_Dining	BCC1
160	2022-12-05	Transaction 297	9.26	debit	Shopping	BCC1
161	2022-12-06	Transaction 298	42.02	debit	Shopping	BCC1
162	2022-12-10	Transaction 299	9.26	credit	Shopping	BCC1
163	2022-12-04	Transaction 304	9.33	debit	Food_Dining	CCC1
164	2022-09-19	Transaction 305	69.0	debit	Shopping	BCC1
165	2022-10-12	Transaction 309	43.56	debit	Business Services	BCC2

In [132…

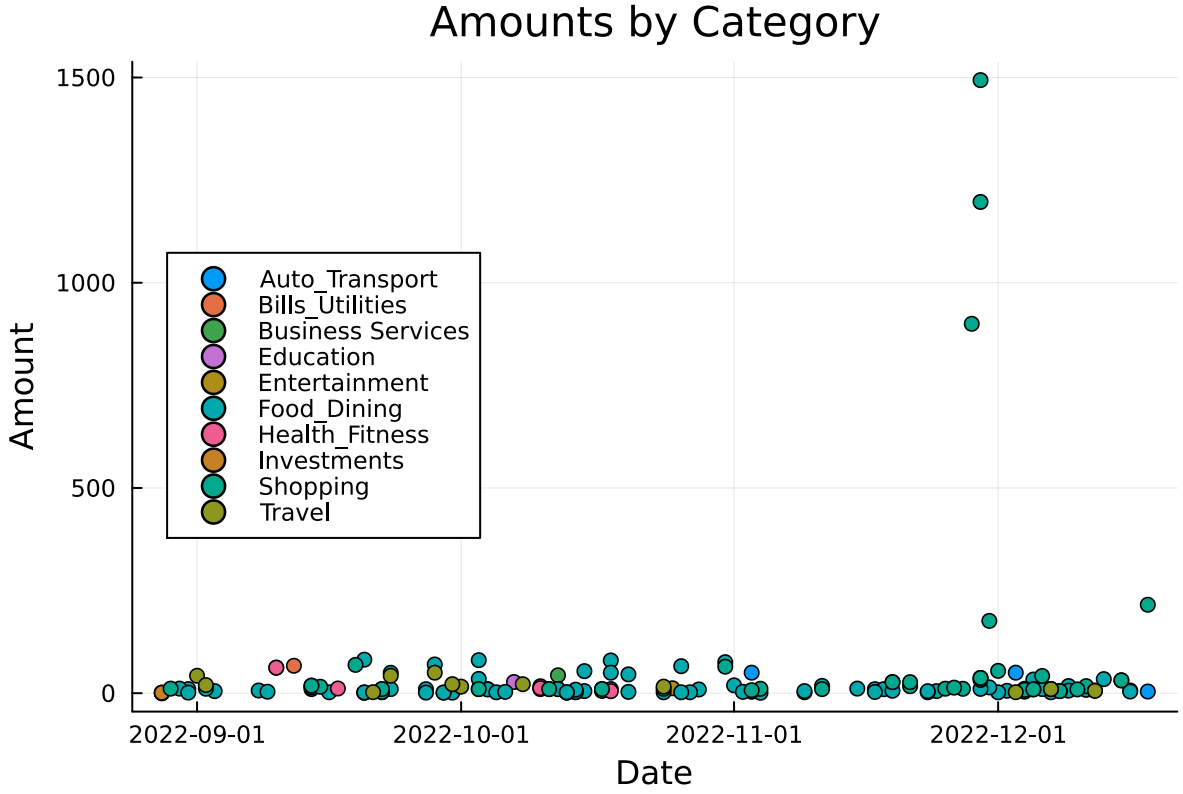
```
# sort dateframe by amount
sort(df, :Amount, rev=true)
```

165×6 DataFrame *140 rows omitted*

Row	Date	Description	Amount	Transaction_Type	Category	Account_Name
	Date	String15	Float64	String7	String31	String7
1	2022-11-29	Transaction 33	1493.62	debit	Shopping	BCC1
2	2022-11-29	Transaction 34	1196.81	debit	Shopping	BCC2
3	2022-11-28	Transaction 32	900.0	debit	Shopping	A1
4	2022-12-18	Transaction 15	215.49	debit	Shopping	CCC1
5	2022-11-30	Transaction 21	176.18	debit	Shopping	CCC1
6	2022-09-20	Transaction 6	81.69	debit	Food_Dining	BCC1
7	2022-10-03	Transaction 7	80.48	debit	Food_Dining	BCC2
8	2022-10-18	Transaction 5	79.74	debit	Food_Dining	BCC2
9	2022-10-31	Transaction 149	75.62	debit	Food_Dining	BCC1
10	2022-09-28	Transaction 243	69.99	debit	Food_Dining	BCC2
11	2022-09-19	Transaction 305	69.0	debit	Shopping	BCC1
12	2022-09-12	Transaction 9	66.81	debit	Bills_Utilities	BCC1
13	2022-10-26	Transaction 242	65.79	debit	Food_Dining	BCC1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
154	2022-10-13	Transaction 280	1.5	debit	Food_Dining	BCC2
155	2022-08-31	Transaction 281	1.5	debit	Food_Dining	BCC1
156	2022-09-29	Transaction 288	1.5	debit	Food_Dining	BCC2
157	2022-09-30	Transaction 90	1.37	debit	Food_Dining	BCC2
158	2022-08-28	Transaction 100	1.05	debit	Investments	CRYPTO
159	2022-11-04	Transaction 236	1.04	debit	Food_Dining	BCC1
160	2022-09-20	Transaction 274	1.0	debit	Food_Dining	BCC1
161	2022-08-28	Transaction 97	0.97	credit	Investments	CRYPTO
162	2022-08-28	Transaction 98	0.97	credit	Investments	CRYPTO
163	2022-08-28	Transaction 101	0.97	debit	Investments	CRYPTO
164	2022-08-28	Transaction 96	0.01	credit	Investments	CRYPTO
165	2022-08-28	Transaction 99	0.01	credit	Investments	CRYPTO

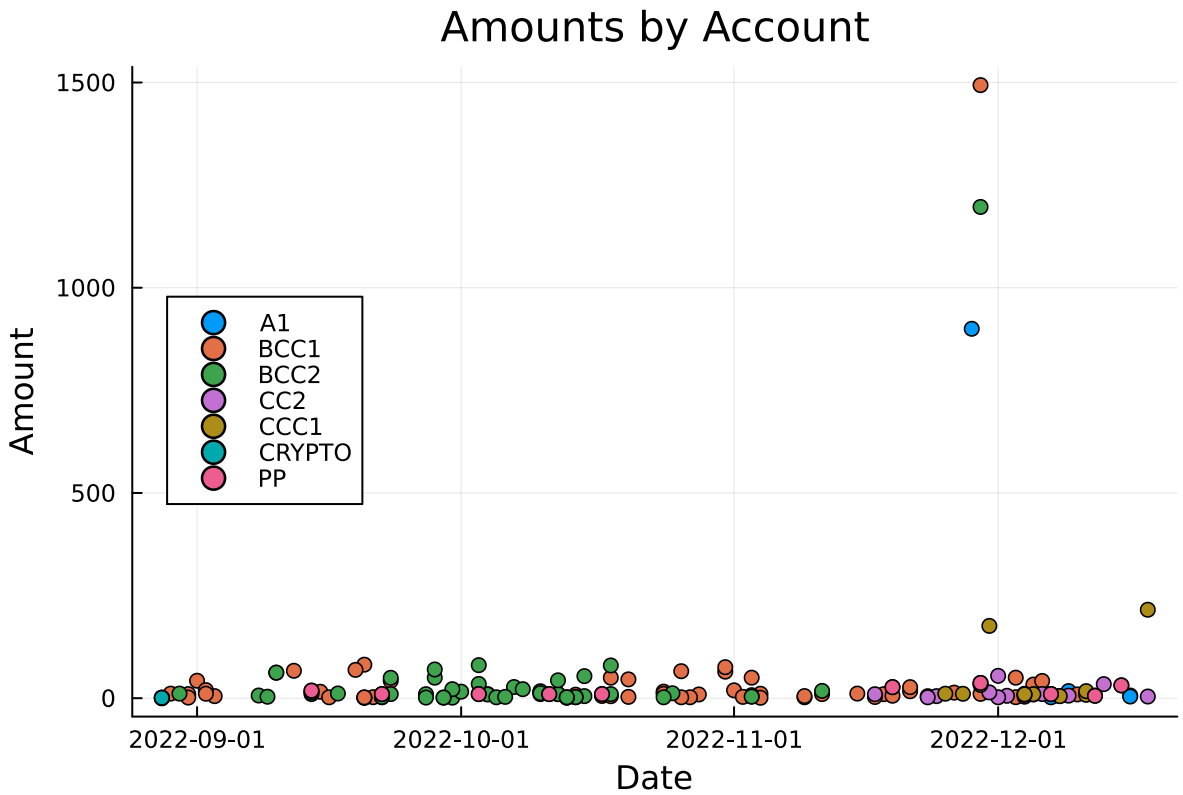
In [133…

```
# categorical plot by category
Plots.scatter(df.Date, df.Amount, group=df.Category, title="Amounts by Category", xlabel='
```



In [134…

```
# plot by account
Plots.scatter(df.Date, df.Amount, group=df.Account_Name, title="Amounts by Account", xlabe
```



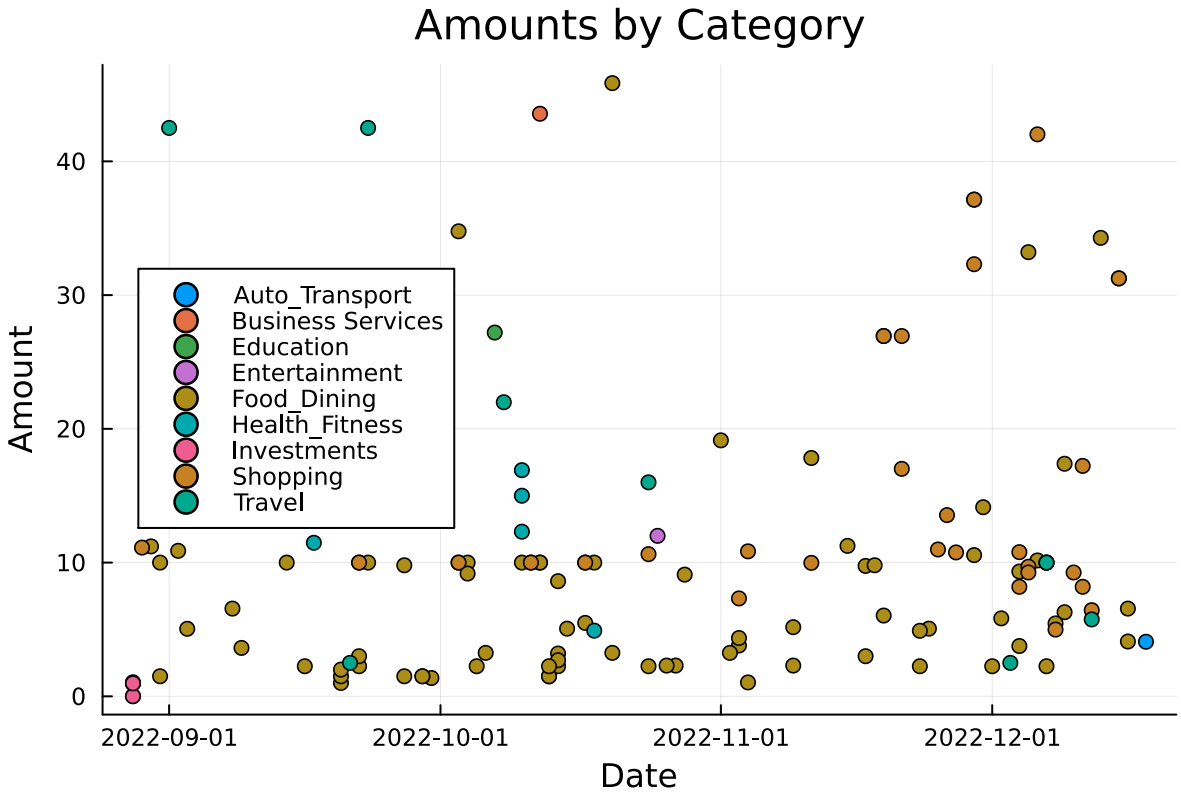
In [135…

```
# calculate outliers
q1 = quantile(df.Amount, 0.25)
q3 = quantile(df.Amount, 0.75)
iqr = q3 - q1
lower = q1 - 1.5 * iqr
upper = q3 + 1.5 * iqr
println("Lower: ", lower, "\nUpper: ", upper)
```

```
# remove outliers
df = filter(row -> row.Amount > lower, df)


df = filter(row -> row.Amount < upper, df)
# plot scatterplot
Plots.scatter(df.Date, df.Amount, group=df.Category, title="Amounts by Category", xlabel="
```

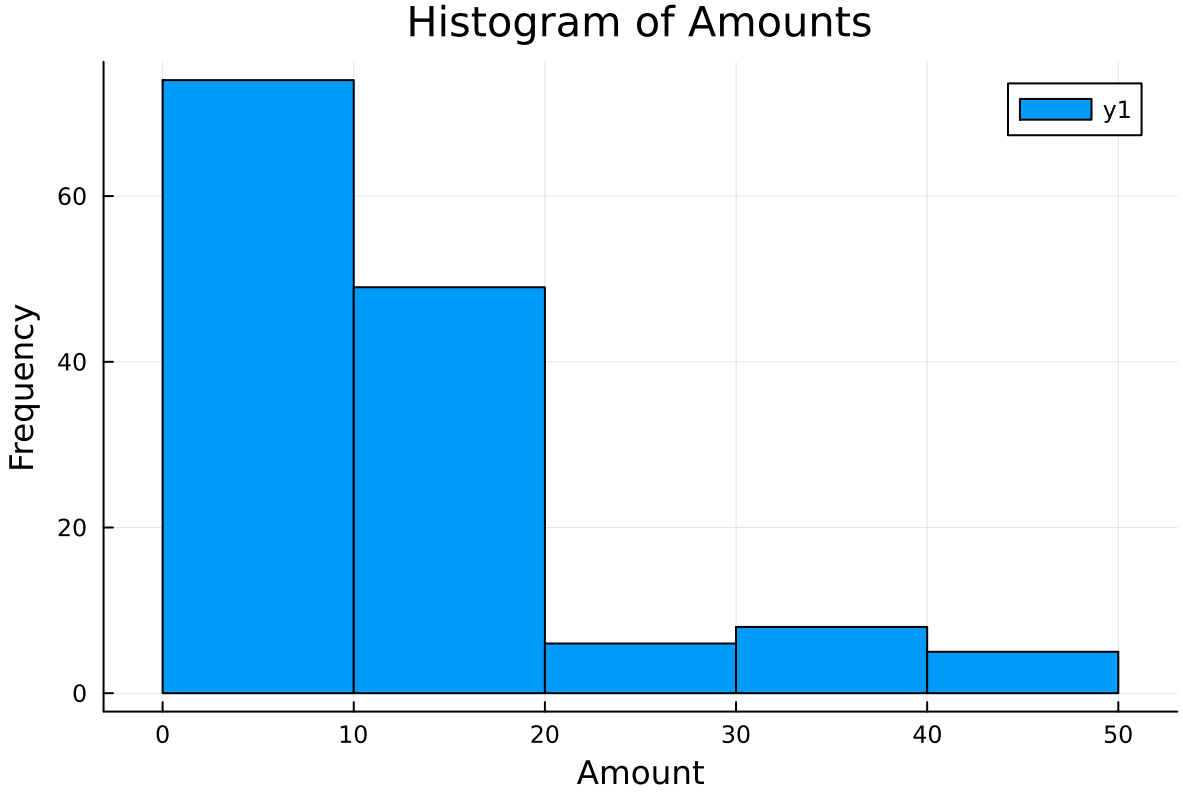
```
Lower: -22.784999999999997
Upper: 48.85499999999999
```



In [136…

```
binwidth = 2 * iqr * length(df.Amount)^(-1 / 3) # calculate binwidth, 2*iqr*n^(-1/3) using

# plot histogram
Plots.histogram(df.Amount, bins=trunc(Int64, binwidth), title="Histogram of Amounts", xlab
```



In [137…

```
# sort df by Date
sort(df, :Date)
```

142×6 DataFrame

117 rows omitted

Row	Date	Description	Amount	Transaction_Type	Category	Account_Name
	Date	String15	Float64	String7	String31	String7
1	2022-08-28	Transaction 96	0.01	credit	Investments	CRYPTO
2	2022-08-28	Transaction 97	0.97	credit	Investments	CRYPTO
3	2022-08-28	Transaction 98	0.97	credit	Investments	CRYPTO
4	2022-08-28	Transaction 99	0.01	credit	Investments	CRYPTO
5	2022-08-28	Transaction 100	1.05	debit	Investments	CRYPTO
6	2022-08-28	Transaction 101	0.97	debit	Investments	CRYPTO
7	2022-08-29	Transaction 260	11.12	debit	Shopping	BCC1
8	2022-08-30	Transaction 160	11.21	debit	Food_Dining	BCC2
9	2022-08-31	Transaction 91	10.0	debit	Food_Dining	BCC1
10	2022-08-31	Transaction 281	1.5	debit	Food_Dining	BCC1
11	2022-09-01	Transaction 70	42.5	debit	Travel	BCC1
12	2022-09-02	Transaction 151	19.76	debit	Travel	BCC1
13	2022-09-02	Transaction 259	10.88	debit	Food_Dining	BCC1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
131	2022-12-09	Transaction 232	6.29	debit	Food_Dining	CC2
132	2022-12-10	Transaction 299	9.26	credit	Shopping	BCC1
133	2022-12-11	Transaction 20	8.19	credit	Shopping	CCC1

Row	Date	Description	Amount	Transaction_Type	Category	Account_Name
	Date	String15	Float64	String7	String31	String7
134	2022-12-11	Transaction 22	17.23	debit	Shopping	CCC1
135	2022-12-12	Transaction 28	6.44	debit	Shopping	CCC1
136	2022-12-12	Transaction 252	5.75	debit	Travel	PP
137	2022-12-13	Transaction 141	34.28	debit	Food_Dining	CC2
138	2022-12-15	Transaction 1	31.25	debit	Shopping	A1
139	2022-12-15	Transaction 197	31.25	debit	Shopping	PP
140	2022-12-16	Transaction 117	6.56	debit	Food_Dining	A1
141	2022-12-16	Transaction 248	4.1	debit	Food_Dining	A1
142	2022-12-18	Transaction 69	4.08	debit	Auto_Transport	CC2

In [138…

```
df_before_october_3rd = filter(row -> row.Date < Date(2022, 10, 3), df)
df_after_october_3rd = filter(row -> row.Date >= Date(2022, 10, 3), df)
```

102×6 DataFrame

77 rows omitted

Row	Date	Description	Amount	Transaction_Type	Category	Account_Name
	Date	String15	Float64	String7	String31	String7
1	2022-12-15	Transaction 1	31.25	debit	Shopping	A1
2	2022-12-08	Transaction 10	4.99	debit	Shopping	CCC1
3	2022-11-29	Transaction 12	32.31	debit	Shopping	CCC1
4	2022-12-04	Transaction 13	10.78	debit	Shopping	CCC1
5	2022-11-21	Transaction 14	17.01	debit	Shopping	BCC1
6	2022-12-11	Transaction 20	8.19	credit	Shopping	CCC1
7	2022-12-11	Transaction 22	17.23	debit	Shopping	CCC1
8	2022-11-27	Transaction 24	10.76	debit	Shopping	CCC1
9	2022-11-25	Transaction 25	10.98	debit	Shopping	CCC1
10	2022-12-05	Transaction 26	9.69	debit	Shopping	CCC1
11	2022-12-04	Transaction 27	8.19	debit	Shopping	CCC1
12	2022-12-12	Transaction 28	6.44	debit	Shopping	CCC1
13	2022-12-09	Transaction 29	17.39	debit	Food_Dining	A1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
91	2022-10-20	Transaction 286	3.25	debit	Food_Dining	BCC1
92	2022-10-24	Transaction 287	2.25	debit	Food_Dining	BCC2
93	2022-10-13	Transaction 289	2.25	debit	Food_Dining	BCC2
94	2022-10-06	Transaction 291	3.25	debit	Food_Dining	BCC2
95	2022-11-15	Transaction 294	11.25	debit	Food_Dining	BCC1
96	2022-10-18	Transaction 295	4.91	debit	Health_Fitness	BCC1

Row	Date	Description	Amount	Transaction_Type	Category	Account_Name
	Date	String15	Float64	String7	String31	String7
97	2022-11-29	Transaction 296	10.56	debit	Food_Dining	BCC1
98	2022-12-05	Transaction 297	9.26	debit	Shopping	BCC1
99	2022-12-06	Transaction 298	42.02	debit	Shopping	BCC1
100	2022-12-10	Transaction 299	9.26	credit	Shopping	BCC1
101	2022-12-04	Transaction 304	9.33	debit	Food_Dining	CCC1
102	2022-10-12	Transaction 309	43.56	debit	Business Services	BCC2

In [139…

```
println(freqtable(df_before_october_3rd.Category))
println(freqtable(df_after_october_3rd.Category))
```

```
5-element Named Vector{Int64}
Dim1                          │
──────────────────────────────┼────
String31("Food_Dining")       │ 23
String31("Health_Fitness")    │  1
String31("Investments")       │  6
String31("Shopping")          │  4
String31("Travel")            │  6
8-element Named Vector{Int64}
Dim1                            │
────────────────────────────────┼────
String31("Auto_Transport")      │  1
String31("Business Services")   │  1
String31("Education")           │  1
String31("Entertainment")       │  1
String31("Food_Dining")         │ 58
String31("Health_Fitness")      │  4
String31("Shopping")            │ 31
String31("Travel")              │  5
```

In [140…

```
#  i know :( it looks ugly but it works
b4 = []
categories = ["Food_Dining", "Shopping", "Public_Transportation", "Health_Fitness", "Inves
for i in 1:length(categories)
    push!(b4, sum(filter(row -> occursin(categories[i], row.Category), df_before_october_3
end
```

In [141…

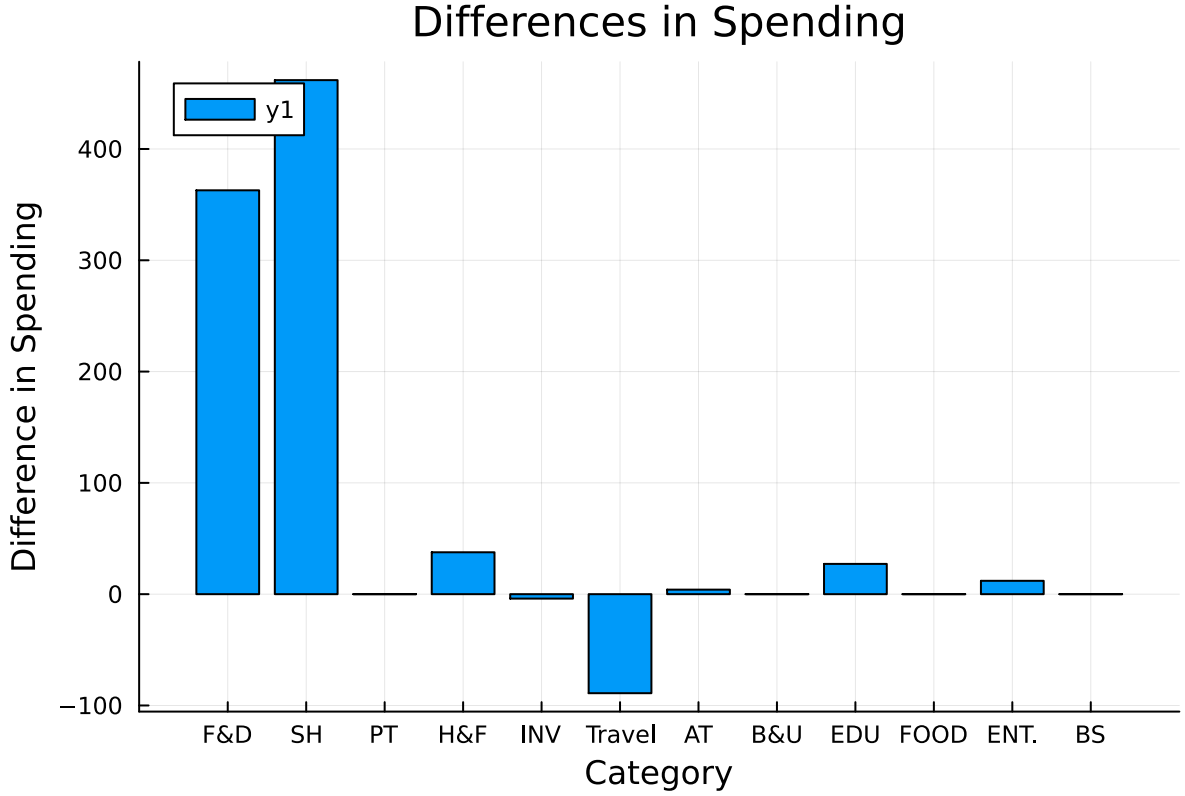
```
aft = []
for i in 1:length(categories)
    push!(aft, sum(filter(row -> occursin(categories[i], row.Category), df_after_october_3
end
```

In [142…

```
differences = []
for i in 1:12
    push!(differences, aft[i] - b4[i])
end
```

In [143…

```
# plot bar graph,put a legend on the bar graph
Plots.bar(["F&D", "SH", "PT", "H&F", "INV", "Travel", "AT", "B&U", "EDU", "FOOD", "ENT.",
```



In [144…

```julia
for i in 1:12
    if differences[i] > 0
        print("You spent more. Amount spent: ", round(differences[i]), " dollars, ")
        print("Category: ", categories[i], " \n")
    else
        print("You spent less. Amount spent: ", round(differences[i]), " dollars, ")
        print("Category: ", categories[i], " \n")
    end
end
```

```
You spent more. Amount spent: 363.0 dollars, Category: Food_Dining
You spent more. Amount spent: 462.0 dollars, Category: Shopping
You spent less. Amount spent: 0.0 dollars, Category: Public_Transportation
You spent more. Amount spent: 38.0 dollars, Category: Health_Fitness
You spent less. Amount spent: -4.0 dollars, Category: Investments
You spent less. Amount spent: -89.0 dollars, Category: Travel
You spent more. Amount spent: 4.0 dollars, Category: Auto_Transport
You spent less. Amount spent: 0.0 dollars, Category: Bills_Utilities
You spent more. Amount spent: 27.0 dollars, Category: Education
You spent less. Amount spent: 0.0 dollars, Category: Food_Delivery
You spent more. Amount spent: 12.0 dollars, Category: Entertainment
You spent less. Amount spent: 0.0 dollars, Category: Business_Services
```