Lecture 2: Introduction to Julia

Dataset

We will use Traffic Stops from Cincinnati City.

Data description: This dataset captures all subjects of traffic stops involving motor vehicles. Time of incident, officer assignment, race/sex of stop subject, and outcome of the stop ("Action taken") are also included in this data. Individual traffic stops may populate multiple data rows to account for multiple subjects and multiple outcomes: "incident number" is the unique identifier for every one (1) traffic stop.

Filename: "Traffic_Crash_Reports__CPD__Aug2018.csv" *Make sure this file in the same directory as the ipynb file*

Setup: Use Julia 0.6.4 kernel. Install the packages CSV, Gadfly, Cairo and Fontconfig.

In [5]:
```
Pkg.add("CSV",VersionNumber("0.2.5"));
Pkg.add("Gadfly",VersionNumber("0.8.0"));
Pkg.add("Cairo",VersionNumber("0.5.6"));
Pkg.add("Fontconfig",VersionNumber("0.1.1"))
Pkg.add("RDatasets",VersionNumber("0.4.0"))
```

```
INFO: Package CSV is already installed
INFO: Package Gadfly is already installed
INFO: Package Cairo is already installed
INFO: Package Fontconfig is already installed
INFO: Package RDatasets is already installed
```

Use the packages...

In [2]:
```
using CSV, DataFrames, Gadfly, Cairo, Fontconfig;
```

Questions

Q 1: Write Julia code to load this data into memory.

In [3]:
```
data = CSV.read("Traffic_Crash_Reports__CPD__Aug2018.csv", missingstring="NA",
rows_for_type_detect=2567)
```

Out[3]:

	ADDRESS_X	LATITUDE_X	LONGITUDE_X	AGE	COMMUNITY_COUNCIL_NEI
1	47XX READING RD	39.1721	84.4678	18-25	N/A
2	29XX MONTANA AV	39.1489	84.5963	31-40	N/A
3	39XX W LIBERTY ST	39.118	-84.578	26-30	WEST PRICE HILL
4	29XX WASSON RD	39.1436	84.4347	41-50	N/A
5	S I75 AT 1-8 MM	39.1148	-84.5319	26-30	WEST END
6	56XX RIDGE AV	39.1751	-84.427	UNDER 18	PLEASANT RIDGE
7	26XX FIRTREE CT	39.1744	-84.5459	26-30	NORTHSIDE
8	50XX PADDOCK RD	39.1778	-84.4777	61-70	BOND HILL
9	40XX HAMILTON AV	39.1583	-84.54	18-25	NORTHSIDE
10	29XX VERNON PL	39.1344	-84.4998	51-60	AVONDALE

	ADDRESS_X	LATITUDE_X	LONGITUDE_X	AGE	COMMUNITY_COUNCIL_NEI
11	13XX W NORTH BEND RD	39.2017	-84.5415	UNKNOWN	COLLEGE HILL
12	4XX E MARTIN LUTHER KING DR	39.1351	84.4994	51-60	N/A
13	39XX BEEKMAN ST	39.1582	84.5495	51-60	N/A
14	S I75 AT 1-2 MM	39.1061	-84.5301	41-50	QUEENSGATE
15	7XX W KING DR	39.1403	84.5316	18-25	N/A
16	6XX CLEMMER AV	39.1288	-84.5277	UNKNOWN	CUF
17	1XX W MCMILLAN ST	39.1271	-84.5169	18-25	CUF - HEIGHTS
18	54XX BAHAMA TE	39.1899	84.568	UNKNOWN	N/A
19	N I471 AT 5.4	39.1035	84.4978	18-25	N/A
20	N I75 AT 2-5 MM	39.126	-84.5342	26-30	CAMP WASHINGTON

	ADDRESS_X	LATITUDE_X	LONGITUDE_X	AGE	COMMUNITY_COUNCIL_NEI
21	32XX GLENWAY AV	39.1125	84.563	31-40	N/A
22	S I71 AT 2-6 MM	39.119	-84.4993	18-25	WALNUT HILLS
23	36XX GLENWAY AV	39.1133	84.5722	41-50	N/A
24	3XX FOREST AV	39.146	-84.5001	51-60	AVONDALE
25	13XX TENNESSEE AV	39.1679	84.4705	26-30	N/A
26	1XX W 3RD ST	39.0991	84.5145	61-70	N/A
27	49XX GLENWAY AV	39.1209	84.601	31-40	N/A
28	1XX CRAFT ST	39.1784	-84.5115	31-40	WINTON HILLS
29	2XX GOODMAN AV	39.1366	-84.5032	31-40	CORRYVILLE
30	37XX GLENWAY AV	39.113	84.5746	UNKNOWN	N/A
⋮	⋮	⋮	⋮	⋮	⋮

Q 2: What is the size of the data?

```
In [13]: size(data)
```

```
Out[13]: (2567, 25)
```

Q 3: Create a new Dataframe by selecting the columns AGE, CRASHSEVERITY, DAYOFWEEK, GENDER, INJURIES, LIGHTCONDITIONSPRIMARY, LOCALREPORTNO, MANNEROFCRASH, ROADSURFACE, WEATHER, ZIP

From here on wards work with the new Dataframe.

In [11]:
```
data1 = data[:,[:AGE,
        :CRASHSEVERITY,
        :DAYOFWEEK,
        :GENDER,
        :INJURIES,
        :LIGHTCONDITIONSPRIMARY,
        :LOCALREPORTNO,
        :MANNEROFCRASH,
        :ROADSURFACE,
        :WEATHER,
        :ZIP]]
```

Out[11]:

	AGE	CRASHSEVERITY	DAYOFWEEK	GENDER	INJURIES	LIGHTCON
1	18-25	3 - PROPERTY DAMAGE ONLY (PDO)	FRI	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
2	31-40	2 - INJURY	THU	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
3	26-30	2 - INJURY	FRI	M - MALE	2 - POSSIBLE	5 - DARK – LIGHTED
4	41-50	3 - PROPERTY DAMAGE ONLY (PDO)	MON	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
5	26-30	3 - PROPERTY DAMAGE ONLY (PDO)	FRI	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
6	UNDER 18	2 - INJURY	WED	F - FEMALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
7	26-30	3 - PROPERTY DAMAGE ONLY (PDO)	SAT	M - MALE	1 - NO INJURY / NONE REPORTED	9 - UNKNO
8	61-70	3 - PROPERTY DAMAGE ONLY (PDO)	TUE	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
9	18-25	2 - INJURY	FRI	F - FEMALE	2 - POSSIBLE	1 - DAYLIG
10	51-60	3 - PROPERTY DAMAGE ONLY (PDO)	TUE	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
11	UNKNOWN	2 - INJURY	SUN	missing	1 - NO INJURY / NONE REPORTED	4 - DARK - ROADWAY
12	51-60	3 - PROPERTY DAMAGE ONLY (PDO)	FRI	F - FEMALE	1 - NO INJURY / NONE REPORTED	4 - DARK - ROADWAY
13	51-60	3 - PROPERTY DAMAGE ONLY (PDO)	TUE	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG

	AGE	CRASHSEVERITY	DAYOFWEEK	GENDER	INJURIES	LIGHTCON
14	41-50	2 - INJURY	THU	F - FEMALE	3 - NON-INCAPACITATING	3 - DUSK
15	18-25	3 - PROPERTY DAMAGE ONLY (PDO)	WED	F - FEMALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
16	UNKNOWN	3 - PROPERTY DAMAGE ONLY (PDO)	TUE	missing	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
17	18-25	3 - PROPERTY DAMAGE ONLY (PDO)	SAT	M - MALE	1 - NO INJURY / NONE REPORTED	4 - DARK - ROADWAY
18	UNKNOWN	3 - PROPERTY DAMAGE ONLY (PDO)	SAT	missing	1 - NO INJURY / NONE REPORTED	4 - DARK - ROADWAY
19	18-25	3 - PROPERTY DAMAGE ONLY (PDO)	SAT	M - MALE	1 - NO INJURY / NONE REPORTED	4 - DARK - ROADWAY
20	26-30	3 - PROPERTY DAMAGE ONLY (PDO)	TUE	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
21	31-40	3 - PROPERTY DAMAGE ONLY (PDO)	FRI	F - FEMALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
22	18-25	3 - PROPERTY DAMAGE ONLY (PDO)	THU	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
23	41-50	3 - PROPERTY DAMAGE ONLY (PDO)	SUN	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
24	51-60	3 - PROPERTY DAMAGE ONLY (PDO)	FRI	M - MALE	1 - NO INJURY / NONE REPORTED	4 - DARK - ROADWAY
25	26-30	2 - INJURY	TUE	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG

	AGE	CRASHSEVERITY	DAYOFWEEK	GENDER	INJURIES	LIGHTCON
26	61-70	3 - PROPERTY DAMAGE ONLY (PDO)	TUE	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
27	31-40	3 - PROPERTY DAMAGE ONLY (PDO)	MON	F - FEMALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
28	31-40	3 - PROPERTY DAMAGE ONLY (PDO)	SAT	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
29	31-40	3 - PROPERTY DAMAGE ONLY (PDO)	MON	M - MALE	1 - NO INJURY / NONE REPORTED	1 - DAYLIG
30	UNKNOWN	2 - INJURY	FRI	missing	1 - NO INJURY / NONE REPORTED	4 - DARK - ROADWAY
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Q 4: Using showcols, list the different element types in the new data frame. Also list the columns in which there are missing values.

In [36]:
```
unique(showcols(data1)[:,[:eltype]])
```

```
WARNING: showcols(df::AbstractDataFrame, all::Bool=false, values::Bool=true)
 is deprecated, use describe(df, stats=[:eltype, :nmissing, :first, :last]) i
nstead.
Stacktrace:
 [1] depwarn(::String, ::Symbol) at ./deprecated.jl:70
 [2] showcols(::DataFrames.DataFrame, ::Bool, ::Bool) at ./deprecated.jl:57
 (repeats 2 times)
 [3] include_string(::String, ::String) at ./loading.jl:522
 [4] include_string(::Module, ::String, ::String) at /users/PES0801/nifaulla
h/.julia/v0.6/Compat/src/Compat.jl:84
 [5] execute_request(::ZMQ.Socket, ::IJulia.Msg) at /usr/local/julia/0.6.4/si
te/v0.6/IJulia/src/execute_request.jl:180
 [6] (::Compat.#inner#6{Array{Any,1},IJulia.#execute_request,Tuple{ZMQ.Socke
t,IJulia.Msg}})() at /users/PES0801/nifaullah/.julia/v0.6/Compat/src/Compat.j
l:125
 [7] eventloop(::ZMQ.Socket) at /usr/local/julia/0.6.4/site/v0.6/IJulia/src/e
ventloop.jl:8
 [8] (::IJulia.##15#18)() at ./task.jl:335
while loading In[36], in expression starting on line 1
```

Out[36]:

	eltype
1	CategoricalArrays.CategoricalString{UInt32}
2	Int64

In [39]:
```
showcols(data1)
```

```
WARNING: showcols(df::AbstractDataFrame, all::Bool=false, values::Bool=true)
 is deprecated, use describe(df, stats=[:eltype, :nmissing, :first, :last]) i
nstead.
Stacktrace:
 [1] depwarn(::String, ::Symbol) at ./deprecated.jl:70
 [2] showcols(::DataFrames.DataFrame, ::Bool, ::Bool) at ./deprecated.jl:57
 (repeats 2 times)
 [3] include_string(::String, ::String) at ./loading.jl:522
 [4] include_string(::Module, ::String, ::String) at /users/PES0801/nifaulla
h/.julia/v0.6/Compat/src/Compat.jl:84
 [5] execute_request(::ZMQ.Socket, ::IJulia.Msg) at /usr/local/julia/0.6.4/si
te/v0.6/IJulia/src/execute_request.jl:180
 [6] (::Compat.#inner#6{Array{Any,1},IJulia.#execute_request,Tuple{ZMQ.Socke
t,IJulia.Msg}})() at /users/PES0801/nifaullah/.julia/v0.6/Compat/src/Compat.j
l:125
 [7] eventloop(::ZMQ.Socket) at /usr/local/julia/0.6.4/site/v0.6/IJulia/src/e
ventloop.jl:8
 [8] (::IJulia.##15#18)() at ./task.jl:335
while loading In[39], in expression starting on line 1
```

Out[39]:

	variable	eltype	nmissing	firs
1	AGE	CategoricalArrays.CategoricalString{UInt32}	0	18-
2	CRASHSEVERITY	CategoricalArrays.CategoricalString{UInt32}	0	3 - PR DA ON (PD
3	DAYOFWEEK	CategoricalArrays.CategoricalString{UInt32}	0	FR
4	GENDER	CategoricalArrays.CategoricalString{UInt32}	299	M -
5	INJURIES	CategoricalArrays.CategoricalString{UInt32}	10	1 - INJ NC RE
6	LIGHTCONDITIONSPRIMARY	CategoricalArrays.CategoricalString{UInt32}	0	1 - DA
7	LOCALREPORTNO	CategoricalArrays.CategoricalString{UInt32}	0	185
8	MANNEROFCRASH	CategoricalArrays.CategoricalString{UInt32}	0	2 - EN
9	ROADSURFACE	CategoricalArrays.CategoricalString{UInt32}	0	1 - CC
10	WEATHER	CategoricalArrays.CategoricalString{UInt32}	0	1 -
11	ZIP	Int64	18	452

1. Gender
2. Injuries
3. Zip

Q 5: Remove the rows in the missing values from the Dataframe. Comment on the number of rows that have been removed in this process.

In [40]:
```
data1 = dropmissing(data1);
size(data1)
```

Out[40]: (2250, 11)

317 rows are removed

Q 6: List the unique entries in the CRASHSEVERITY column

In [41]:
```
unique(data1[:,[:CRASHSEVERITY]])
```

Out[41]:

	CRASHSEVERITY
1	3 - PROPERTY DAMAGE ONLY (PDO)
2	2 - INJURY
3	1 - FATAL INJURY

Q 7: Find out the different types of crashes in this data.

In [42]:
```
unique(data1[:,[:MANNEROFCRASH]])
```

Out[42]:

	MANNEROFCRASH
1	2 - REAR-END
2	6 - ANGLE
3	8 - SIDESWIPE, OPPOSITE DIRECTION
4	5 - BACKING
5	1 - NOT COLLISION BETWEEN TWO MOTOR VEHICLES IN TRANSPORT
6	7 - SIDESWIPE, SAME DIRECTION
7	3 - HEAD-ON
8	9 - UNKNOWN
9	4 - REAR-TO-REAR

Q 8: Find out the different types of WEATHER conditions in this data.

In [43]:
```
unique(data1[:,[:WEATHER]])
```

Out[43]:

	WEATHER
1	1 - CLEAR
2	2 - CLOUDY
3	4 - RAIN
4	9 - OTHER/UNKNOWN
5	3 - FOG, SMOG, SMOKE

Q 9: Determine the number of crashes happened in each of these weather conditions using by() function.

In [44]:
```
ans = by(data1, :WEATHER, nrow)
```

```
WARNING: imported binding for ans overwritten in module Main
```

Out[44]:

	WEATHER	x1
1	1 - CLEAR	1519
2	2 - CLOUDY	402
3	4 - RAIN	321
4	9 - OTHER/UNKNOWN	7
5	3 - FOG, SMOG, SMOKE	1

Q 10: Find out the different light conditions in this data.

In [45]:
```
unique(data1[:,[:LIGHTCONDITIONSPRIMARY]])
```

Out[45]:

	LIGHTCONDITIONSPRIMARY
1	1 - DAYLIGHT
2	5 - DARK – ROADWAY NOT LIGHTED
3	9 - UNKNOWN
4	4 - DARK - LIGHTED ROADWAY
5	6 - DARK – UNKNOWN ROADWAY LIGHTING
6	2 - DAWN
7	3 - DUSK

Q 11: Determine the number of crashes happened in each combination of weather and light conditions using by() function. State your observations.

In [49]:
```
ans = by(data1, [:WEATHER, :LIGHTCONDITIONSPRIMARY] , nrow)
```

Out[49]:

	WEATHER	LIGHTCONDITIONSPRIMARY	x1
1	1 - CLEAR	1 - DAYLIGHT	1200
2	1 - CLEAR	5 - DARK – ROADWAY NOT LIGHTED	9
3	2 - CLOUDY	1 - DAYLIGHT	347
4	4 - RAIN	1 - DAYLIGHT	245
5	1 - CLEAR	9 - UNKNOWN	1
6	1 - CLEAR	4 - DARK - LIGHTED ROADWAY	259
7	4 - RAIN	4 - DARK - LIGHTED ROADWAY	62
8	2 - CLOUDY	4 - DARK - LIGHTED ROADWAY	37
9	1 - CLEAR	6 - DARK – UNKNOWN ROADWAY LIGHTING	4
10	4 - RAIN	2 - DAWN	8
11	1 - CLEAR	2 - DAWN	9
12	1 - CLEAR	3 - DUSK	37
13	2 - CLOUDY	3 - DUSK	9
14	4 - RAIN	5 - DARK – ROADWAY NOT LIGHTED	3
15	2 - CLOUDY	9 - UNKNOWN	1
16	9 - OTHER/UNKNOWN	4 - DARK - LIGHTED ROADWAY	4
17	2 - CLOUDY	2 - DAWN	6
18	2 - CLOUDY	5 - DARK – ROADWAY NOT LIGHTED	2
19	9 - OTHER/UNKNOWN	9 - UNKNOWN	2
20	4 - RAIN	3 - DUSK	3
21	3 - FOG, SMOG, SMOKE	4 - DARK - LIGHTED ROADWAY	1
22	9 - OTHER/UNKNOWN	1 - DAYLIGHT	1

Combination of Clear weather & Day Light has disproportionately higher amount of crash. Crashes in cloudy weather and Day light are distant second, followed by Clear & Dark lighted Roadway and Rainy weather & Day light

Q 12: How many ZIP codes are covered in this data.

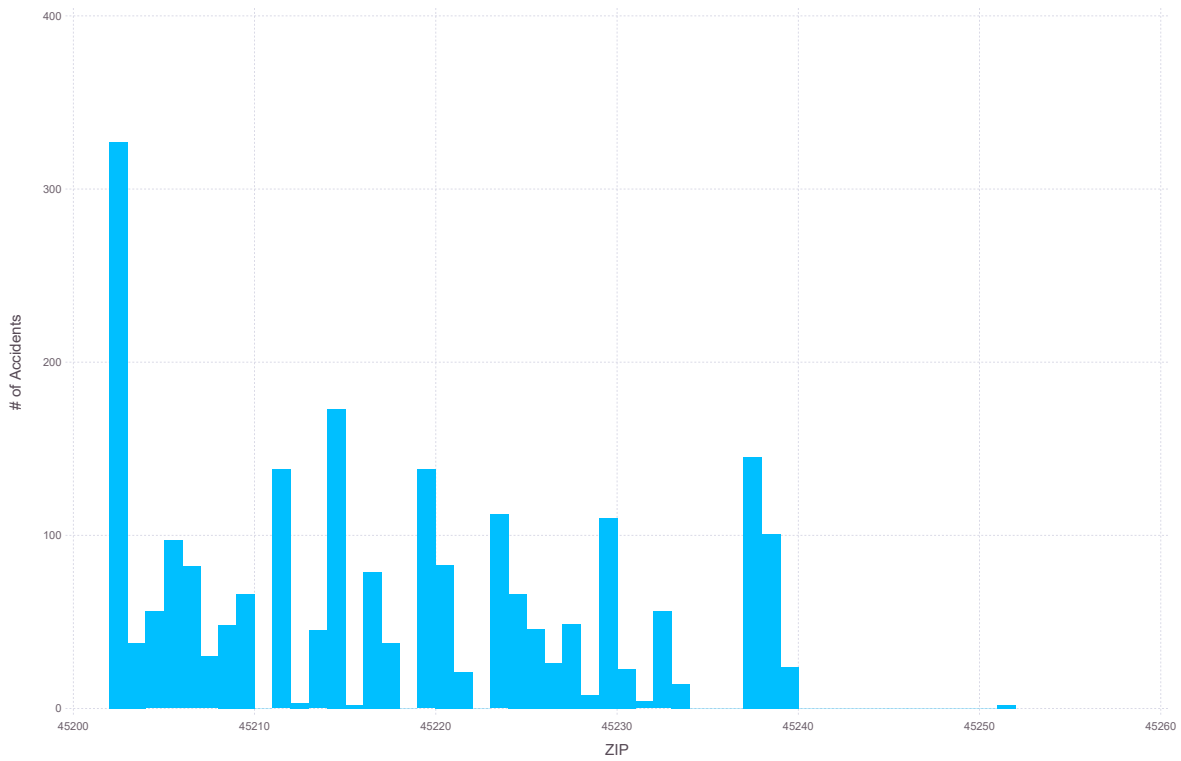
In [48]:
```
size(unique(data1[:,[:ZIP]]))
```

Out[48]: (33, 1)

Q 13: Plot a bar graph showing the number of accidents in each of the ZIP codes

In [71]:
```
Gadfly.plot(data1, x="ZIP", Geom.histogram, Guide.ylabel("# of Accidents"))
```

Out[71]:



Step 14: Draw a scatter plot between weather and light conditions. State your observations. Please use set_default_plot_size(12inch, 8inch) function to adjust the figure size as needed for visibility.

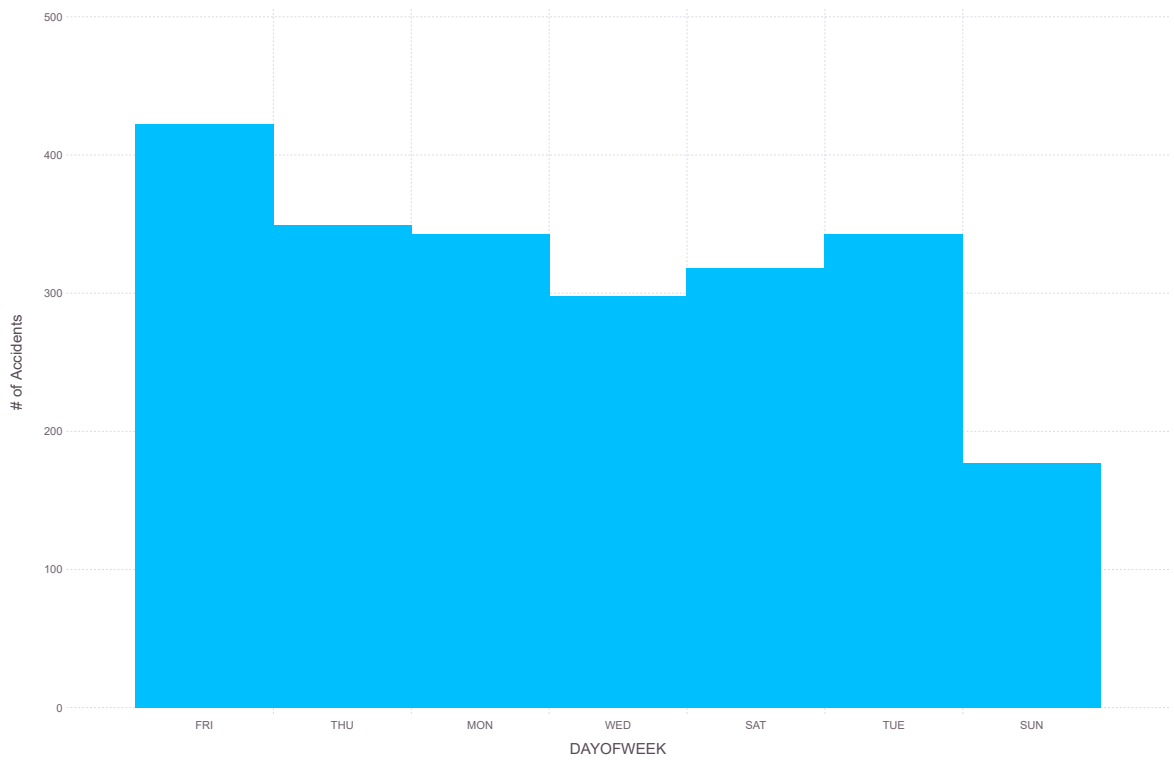
In [72]:
```
set_default_plot_size(12inch, 8inch)
Gadfly.plot(data1, x="WEATHER", y ="LIGHTCONDITIONSPRIMARY" , Geom.point,
Stat.x_jitter(range=0.5, seed = 10),
Stat.y_jitter(range=0.5, seed = 25))
```

Out[72]:

Step 15: Make a plot to view the number of crashes on different days of the week. On which day of the week fewer crashes happen? On which day of the week more crashes happen?

In [70]:
```
Gadfly.plot(data1, x="DAYOFWEEK", Geom.histogram, Guide.ylabel("# of Accidents"))
```

Out[70]:

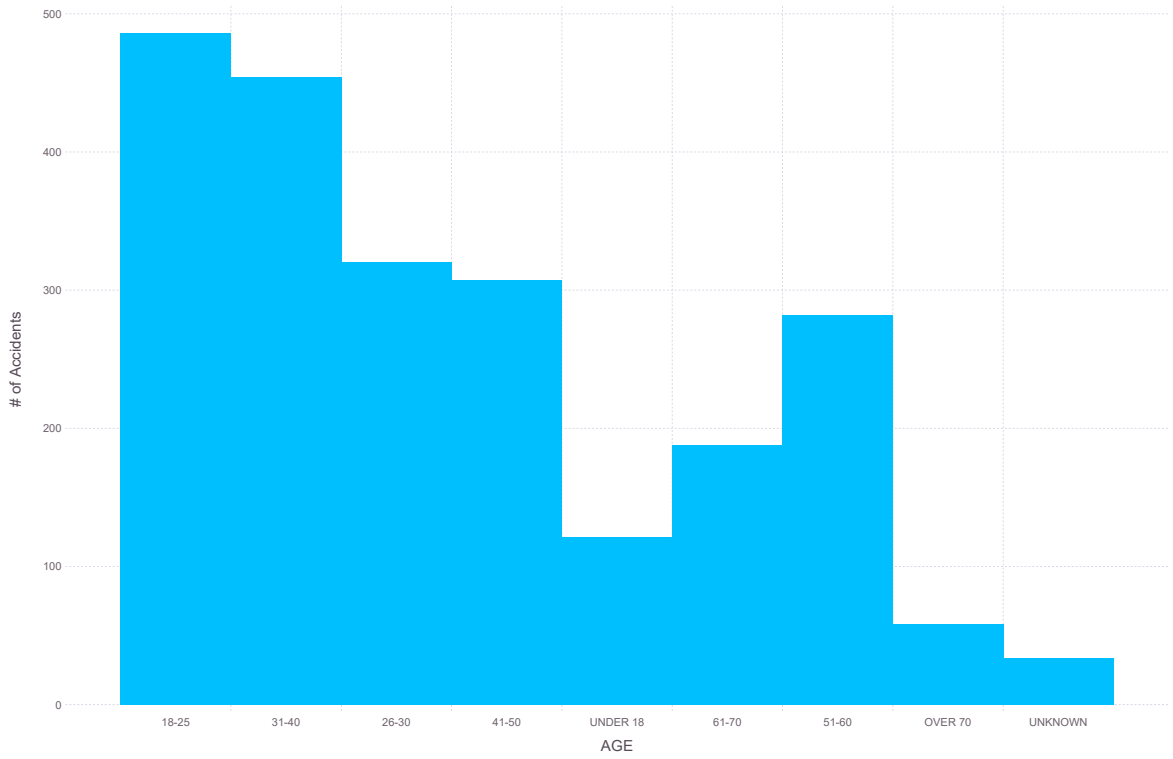


Least Crashes - Sunday
Most Crashes - Friday

Step 16: Make a plot to view the number of crashes reported per age-group. State your observations. State your observations.

In [69]:
```
Gadfly.plot(data1, x="AGE", Geom.histogram, Guide.ylabel("# of Accidents"))
```

Out[69]:



Age group 18-25 and 31-40 have higher number of crashes
Age group under 18 and over 70 proportionately have lower number of crashes

Step 17: Use the following two lines of code to load the "iris" dataset:

using RDatasets

iris = dataset("datasets", "iris");

This dataset has information about flowers from three plant species.

Do:

1. List attributes in this data
2. Generate a scatter plot between "PetalLength" and "PetalWidth" where each point is colored based on "Sepecies". What observations can you make about the flowers from the three plant species based on this plot.

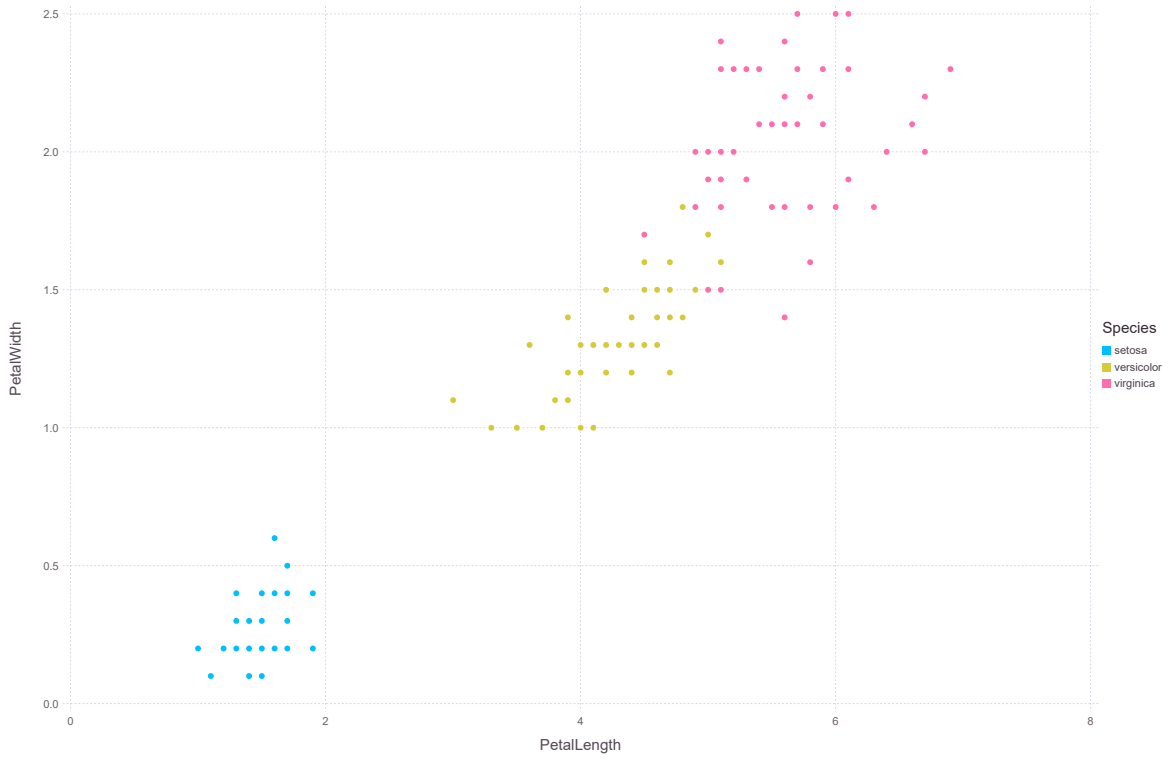
In [59]:
```
using RDatasets
iris = dataset("datasets", "iris");
```

In [60]:
```
names(iris)
```

Out[60]:
```
5-element Array{Symbol,1}:
 :SepalLength
 :SepalWidth
 :PetalLength
 :PetalWidth
 :Species
```

In [63]:
```
set_default_plot_size(12inch, 8inch)
Gadfly.plot(iris, x="PetalLength", y ="PetalWidth" , Geom.point, color="Species")
```

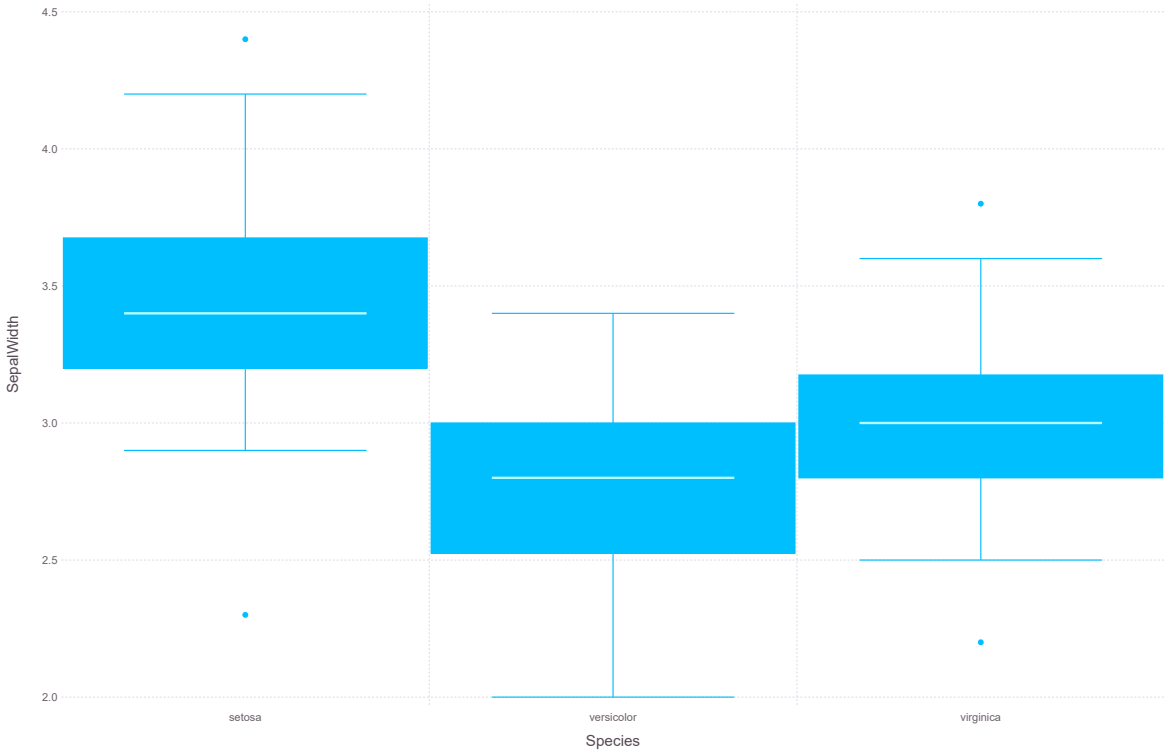
Out[63]:



Step 18: Using IRIS dataset draw a box plot to compare the SepalWidth for the three plant species. What observations can you make based on this plot?

In [66]:
```
Gadfly.plot(iris, x="Species", y ="SepalWidth" , Geom.boxplot)
```

Out[66]:

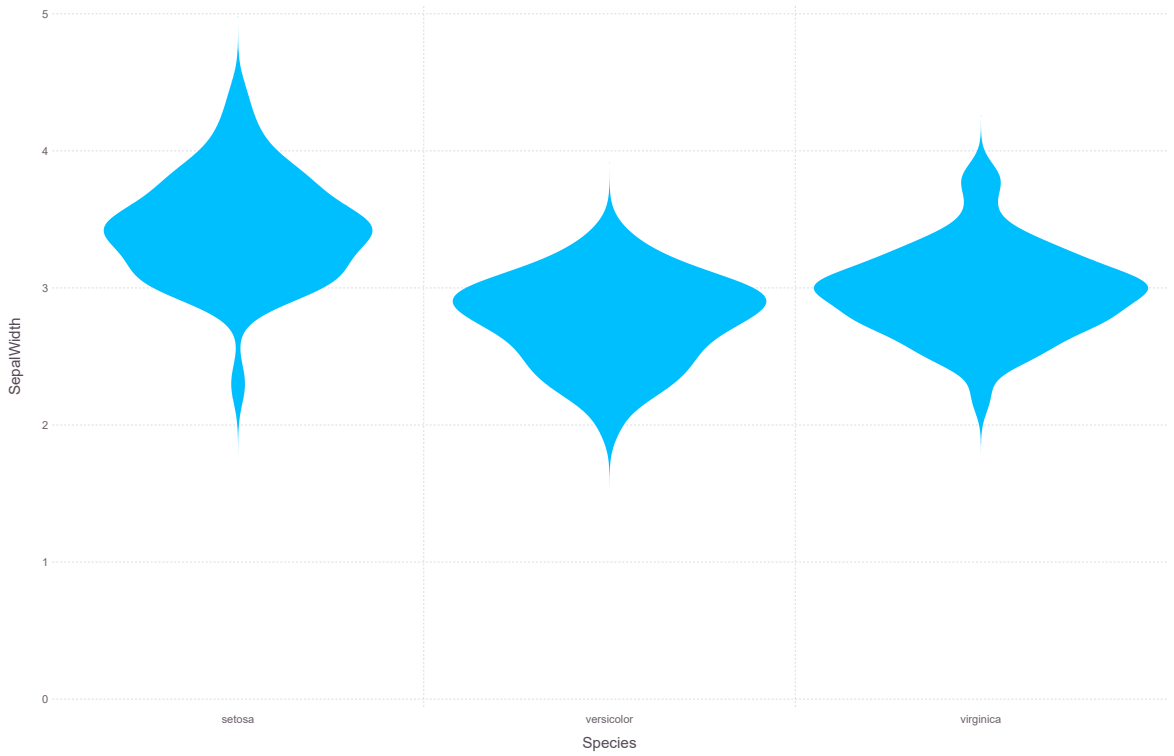


Median value for setosa is greater than virginica which is greater than versicolor.
Lower half of virginica overlaps with upper half of versicolor.
This looks to be a good variable to differentitate setosa from other species.

Step 19: Draw a violin plot for SepalWidth (similar to the box plot above) and state any new observations you may have.

In [68]:
```
Gadfly.plot(iris, x="Species", y ="SepalWidth" , Geom.violin)
```

Out[68]:



Density of Sepal width is greater for setosa against other 2 species. Most values for versicolor and virginica appear around similar region which tells us that it may not be good differentiator for differentiating between versicolor and virginica as it appeared from the box plot.