Project Part A: Exploratory Data Analysis

In [108]:
```
analyst = "Khoa Nguyen" # Replace this with your name
```

In [109]:
```
f = "setup.R"; for (i in 1:10) { if (file.exists(f)) break else f = paste0("../", f) }; source(f)
options(repr.matrix.max.rows=674)
update_geom_defaults("point", list(size=1))
update_geom_defaults("col", list(fill=PALETTE[1]))
out(5.8,2.1)
```

1 Introduction

1.1 Objective

Conduct an exploratory data analysis of public company fundamentals. Later, use the insights gleaned along with additional analysis to recommend a portfolio of 12 company investments that maximizes 12-month return of an overall $1,000,000 investment.

1.2 Approach

Retrieve a public company fundamentals dataset comprising thousands of US companies from quarters 1, 2, 3, and 4 of year 2017 + company stock price data for those companies from quarter 4 of year 2018.

Prepare the data so that each company and its associated information is represented as a single observation.

Apply various descriptive statistics and data visualizations to look for interesting patterns and inter-company relationships.

1.3 Data Source

Data files:

- Data Dictionary.csv
- Company Fundamentals 2017.csv
- Company Fundamentals 2018.csv

The datasets and accompanying data dictionary are sourced from ...

- Wharton Research Data Services > Compustat - Capital IQ from Standard & Poor's > North America - Daily > Fundamentals Quarterly (https://wrds-www.wharton.upenn.edu/ (https://wrds-www.wharton.upenn.edu/))
  - Date Variable: Data Date
  - Date Range: 2017-01 to 2017-12 -or- 2018-01 to 2018-12
  - Company Codes: Search the entire database
    - Consolidtaion Level: C, Output

    - Industry Format: INDL, FS, Output
    - Data Format: STD, Output
    - Population Source: D, Output
    - Quarter Type: Fiscal View, Output
    - Currency: USD, Output (not CAD)
    - Company Status: Active, Output (not Inactive)
- Variable Types: Data Items, Select All (674)
- Query output:
    - Output format: comma-delimited text
    - Compression type: None
    - Data format: MMDDYY10

The datasets are restricted to select US active, publicly held companies that reported quarterly measures including stock prices for 1st, 2nd, 3rd, and 4th quarters in years 2017 and 2018. All non-missing stock prices exceed $3 per share. File formats are all comma-separated values (CSV).

The data dictionary is from Variable Descriptions tab, copied to Excel, saved in csv format.

For this project, do not source any additional data from year 2019.

2 Data

2.1 Data Dictionary

In [110]:
```
# Retrieve the data dictionary.
# How many variable entries in the company fundamentals datasets?
# What are the variable names, types, and descriptions?

dictionary = read.csv("Data Dictionary.csv", header=TRUE, na.strings=c("NA", ""), stringsAsFactors=FALSE)
fmtx(size(dictionary))
fmtx(dictionary)
```

dictionary

Variable.Name	Data.Type	Variable.Description	Help
ACCHGQ	NUM	ACCHGQ -- Accounting Changes - Cumulative Effect	NA
ACCHGY	NUM	ACCHGY -- Accounting Changes - Cumulative Effect	NA
ACCTCHGQ	CHAR	ACCTCHGQ -- Adoption of Accounting Changes	NA
ACCTSTDQ	CHAR	ACCTSTDQ -- Accounting Standard	NA
ACOMINCQ	NUM	ACOMINCQ -- Accumulated Other Comprehensive Income (Loss)	NA
ACOQ	NUM	ACOQ -- Current Assets - Other - Total	NA
ACTQ	NUM	ACTQ -- Current Assets - Total	NA
ADD1	CHAR	ADD1 -- Address Line 1	NA
ADD2	CHAR	ADD2 -- Address Line 2	NA
ADD3	CHAR	ADD3 -- Address Line 3	NA
ADD4	CHAR	ADD4 -- Address Line 4	NA

2.2 Data for Current Year

2.2.1 Retrieve Raw Data

In [*]:
```
# Retrieve the 2017 data.
# How many observations and variables?
# Present the first few observations.

data.raw = read.csv("Company Fundamentals 2017.csv", header=TRUE, na.strings=c("NA", ""), stringsAsFactors=FAI

fmtx(size(data.raw))
fmtx(data.raw[1:10,], FFO)
```

size(data.raw)

observations	variables
33,269	680

In [112]:
```
# How many unique companies?

# You can use length(unique(...))

unique_companies_in_current_year_dataset = length(unique(data.raw$gvkey))
fmtx(unique_companies_in_current_year_dataset)
```

unique_companies_in_current_year_dataset
8,496

2.2.2 Partition Data by Calendar Quarter

Partition the dataset by calendar quarter in which information is reported. Filter in observations to include only those with non-missing `prccq` ≥ 3. Then remove any observations about companies that reported more than once per quarter. Then change all the variable

In [113]:
```
# Partition the dataset as described.
# How many observations and variables in each quarter dataset?

q = quarter(mdy(data.raw$datadate))

data.current.q1 = data.raw[(q==1) & !is.na(data.raw$prccq) & (data.raw$prccq>=3),]
data.current.q2 = data.raw[(q==2) & !is.na(data.raw$prccq) & (data.raw$prccq>=3),]
data.current.q3 = data.raw[(q==3) & !is.na(data.raw$prccq) & (data.raw$prccq>=3),]
data.current.q4 = data.raw[(q==4) & !is.na(data.raw$prccq) & (data.raw$prccq>=3),]

data.current.q1 = data.current.q1[!duplicated(data.current.q1$gvkey),]
data.current.q2 = data.current.q2[!duplicated(data.current.q2$gvkey),]
data.current.q3 = data.current.q3[!duplicated(data.current.q3$gvkey),]
data.current.q4 = data.current.q4[!duplicated(data.current.q4$gvkey),]

data.current.q1 = rename_with(data.current.q1, ~ifelse(. %in% c("gvkey","tic","conm"), ., paste0(.,".q1")))
data.current.q2 = rename_with(data.current.q2, ~ifelse(. %in% c("gvkey","tic","conm"), ., paste0(.,".q2")))
data.current.q3 = rename_with(data.current.q3, ~ifelse(. %in% c("gvkey","tic","conm"), ., paste0(.,".q3")))
data.current.q4 = rename_with(data.current.q4, ~ifelse(. %in% c("gvkey","tic","conm"), ., paste0(.,".q4")))


fmtsx(fmt(size(data.current.q1)),
      fmt(size(data.current.q2)),
      fmt(size(data.current.q3)),
      fmt(size(data.current.q4)))
```

size(data.current.q1)

observations	variables
4,324	680

size(data.current.q2)

observations	variables
4,387	680

size(data.current.q3)

observations	variables
4,397	680

size(data.current.q4)

observations	variables
4,434	680

2.2.3 Consolidate Data by Company

Consolidate the four quarter datasets into one dataset, with one observation per company that includes variables for all four quarters. Remove any observations with missing `prccq.q4` values.

In [114]:
```
# Consolidate the partitions as described.
# How many observations and variables in the resulting dataset?
# Present the first few observations and some variables of the resulting dataset.

# You can use merge(..., ..., by=c("gvkey","tic","conm"), all=TRUE, sort=TRUE) threee times.
data.quarter.12 = merge(data.current.q1, data.current.q2,by=c("gvkey","tic","conm"), all=TRUE, sort=TRUE)
data.quarter.123 = merge(data.quarter.12, data.current.q3,by=c("gvkey","tic","conm"), all=TRUE, sort=TRUE)
data.current = merge(data.quarter.123, data.current.q4,by=c("gvkey","tic","conm"), all=TRUE, sort=TRUE)
data.current = data.current[!is.na(data.current$prccq.q4), ]

fmtx(size(data.current))
fmtx(data.current[1:3,1:13], "data.current (first few observations, first few variables)")
```

size(data.current)

observations	variables
4,434	2,711

data.current (first few observations, first few variables)

gvkey	tic	conm	datadate.q1	fyearq.q1	fqtr.q1	fyr.q1	indfmt.q1	consol.q1	popsrc.q1	datafmt.q1	cusip.q1	acctchg
1,004	AIR	AAR CORP	02/28/2017	2,016	3	5	INDL	C	D	STD	000361105	
1,045	AAL	AMERICAN AIRLINES GROUP INC	03/31/2017	2,017	1	12	INDL	C	D	STD	02376R102	
1,050	CECE	CECO ENVIRONMENTAL CORP	03/31/2017	2,017	1	12	INDL	C	D	STD	125141101	

2.3 Data for Next Year

2.3.1 Retrieve Raw Data

In [115]:
```
# Retrieve the 2018 data.
# How many observations and variables?
# Present the first few observations.

data.raw = read.csv("Company Fundamentals 2018.csv", header=TRUE, na.strings=c("NA", ""), stringsAsFactors=FAl
fmtx(size(data.raw))
fmtx(data.raw[1:3,], FFO)
```

size(data.raw)

observations	variables
35,728	680

gvkey	datadate	fyearq	fqtr	fyr	indfmt	consol	popsrc	datafmt	tic	cusip	conm	acctchgq	acctstdq	adrrq	ajexq	ajpq	b
1,004	02/28/2018	2,017	3	5	INDL	C	D	STD	AIR	000361105	AAR CORP	NA	DS	NA	1	1	
1,004	05/31/2018	2,017	4	5	INDL	C	D	STD	AIR	000361105	AAR CORP	NA	DS	NA	1	1	

gvkey	datadate	fyearq	fqtr	fyr	indfmt	consol	popsrc	datafmt	tic	cusip	conm	acctchgq	acctstdq	adrrq	ajexq	ajpq	b
1,004	08/31/2018	2,018	1	5	INDL	C	D	STD	AIR	000361105	AAR CORP	ASU14-09	DS	NA	1	1	

2.3.2 Filter Data by Calendar Quarter 4

Filter the dataset by calendar quarter in which information is reported, keeping only observations with information reported in quarter 4. Additionally, filter in observations to include only those with non-missing `prccq`, and keep only the `gvkey` and `prccq` variables. Then remove any observations about companies that reported more than once per quarter.

In [116]:
```
# Filter the dataset as described.
# Present the first few observations of the resulting dataset.
q = quarter(mdy(data.raw$datadate))

data.future.q4 = data.raw[(q==4) & !is.na(data.raw$prccq), ]

data.future.q4 = data.future.q4[!duplicated(data.future.q4$gvkey), c("gvkey","prccq")]


fmtsx(fmt(size(data.future.q4)))
fmtx(data.future.q4[1:6,], FFO)
```

size(data.future.q4)

observations	variables
5,968	2

data.future.q4 (first few observations)

gvkey	prccq
1,004	43.69
1,045	32.11
1,050	6.75
1,062	8.66
1,072	15.25
1,075	85.20

2.4 Data for Consolidated Current Year / Next Year

Consolidate the processed 2017 dataset and processed 2018 dataset, keeping only observations that have both 2017 and 2018 information. Then add these 2 synthetic variables:

$$\text{growth} = \left(\text{prccq} - \text{prccq.q4}\right) \div \text{prccq.q4}$$

In [117]:
```
# Consolidate the datasets as described.
# How many observations and variables in the resulting dataset?
# Present the first few observations of the resulting dataset.

threshold = 0.3

data = merge(data.current, data.future.q4, by="gvkey", all=FALSE)

data$growth = (data$prccq - data$prccq.q4) / data$prccq.q4
data$big_growth = factor(data$growth >= threshold, levels=c(TRUE, FALSE), labels=c("YES", "NO"))

data = relocate(data, big_growth, growth, prccq)


fmtx(size(data))
fmtx(data[1:3, 1:13], "data (first few observations, first few variables)")
# fmtx(data[1:3,], FFO)  # takes about 2 minutes to display all variables
```

size(data)

observations	variables
4,305	2,714

data (first few observations, first few variables)

big_growth	growth	prccq	gvkey	tic	conm	datadate.q1	fyearq.q1	fqtr.q1	fyr.q1	indfmt.q1	consol.q1	popsrc.q1
NO	0.0507	43.69	1,004	AIR	AAR CORP	02/28/2017	2,016	3	5	INDL	C	D
NO	-0.3829	32.11	1,045	AAL	AMERICAN AIRLINES GROUP INC	03/31/2017	2,017	1	12	INDL	C	D
YES	0.3158	6.75	1,050	CECE	CECO ENVIRONMENTAL CORP	03/31/2017	2,017	1	12	INDL	C	D

3 Exploratory Data Analysis

3.1 Descriptive Statistics

In [118]:
```
# How many observations and variables?
fmtx(size(data))
```

size(data)

observations	variables
4,305	2,714

In [119]:
```
# Present a variety of statistics about growth.
growth = describe(data$growth)
fmtx(growth)
```

growth

vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
1	4,305	-0.1186	0.4689	-0.1492	-0.1519	0.2629	-0.9956	10.24	11.24	6.58	102.6	0.0071

In [120]:
```
# What is the correlation between prccq.q4 and prccq?
# What is the correlation between prccq.q4 and growth?
prccq.q4_prccqq = cor(data$prccq, data$prccq.q4)
prccq.q4_growth = cor(data$growth, data$prccq.q4)
correlations = data.frame(prccq.q4_prccqq,prccq.q4_growth)
fmtx(correlations)
```

correlations

prccq.q4_prccqq	prccq.q4_growth
0.9603	0.0088

In [121]:
```
# What fraction of observations are missing price data
# (i.e, what fraction are missing data from any of prccq.q1, prccq.q2, prccq.q3, or prccq.q4)?

fmtsx(fmt(sum(is.na(data$prccq.q1) |
        is.na(data$prccq.q2) |
        is.na(data$prccq.q3) )/nrow(data), "faction of observations with missing price data", blank=TRUE))
```

faction of observations with missing price data

0.0857

In [122]:
```
# Present some additional interesting descriptive statistics.

cov.1_2_3_4_growth = cov(data[,c("prccq.q1","prccq.q2","prccq.q3","prccq.q4","growth","prccq")]) # covariance

fmtsx(fmt(cov.1_2_3_4_growth, "Covariance between quarters", row.names=TRUE))
```

Covariance between quarters

	prccq.q1	prccq.q2	prccq.q3	prccq.q4	growth	prccq
prccq.q1	NA	NA	NA	NA	NA	NA
prccq.q2	NA	NA	NA	NA	NA	NA
prccq.q3	NA	NA	NA	NA	NA	NA
prccq.q4	NA	NA	NA	105,973.371	1.3476	96,525.580
growth	NA	NA	NA	1.348	0.2199	6.469
prccq	NA	NA	NA	96,525.580	6.4690	95,345.195

In [123]:
```
# Present some additional interesting descriptive statistics.
fmtsx(fmt(sum(data$big_growth == "YES")/nrow(data), "faction of observations as big growth", blank=TRUE))
```

faction of observations as big growth
0.0836

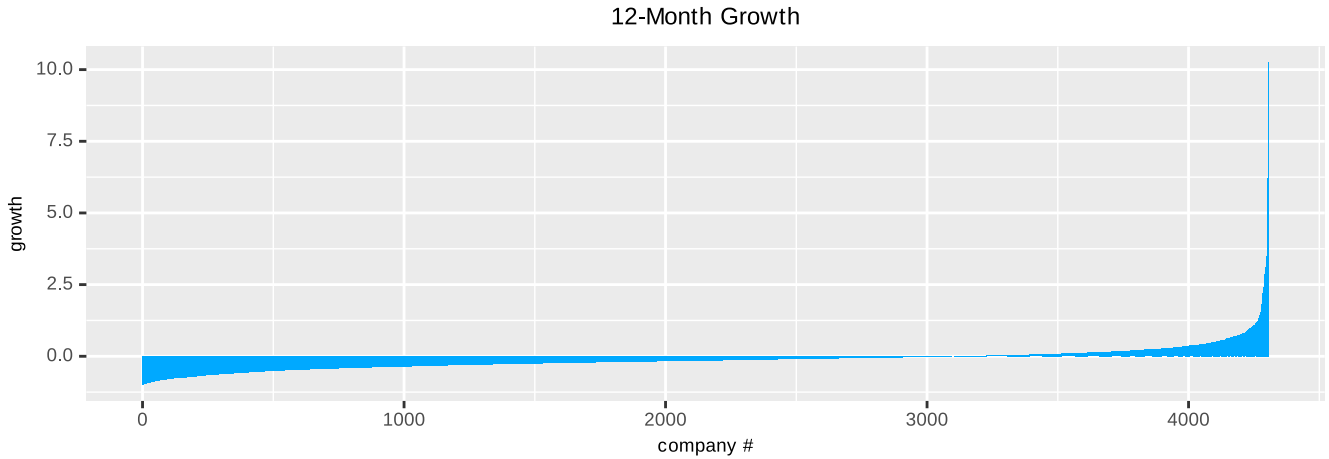
In [124]:
```
# Present some additional interesting descriptive statistics.
as.data.frame(describe(data[,c("prccq.q1","prccq.q2","prccq.q3","prccq.q4","prccq","growth")]))[,c(
    'mean','sd','median','min','max', 'range')]
```

A data.frame: 6 × 6

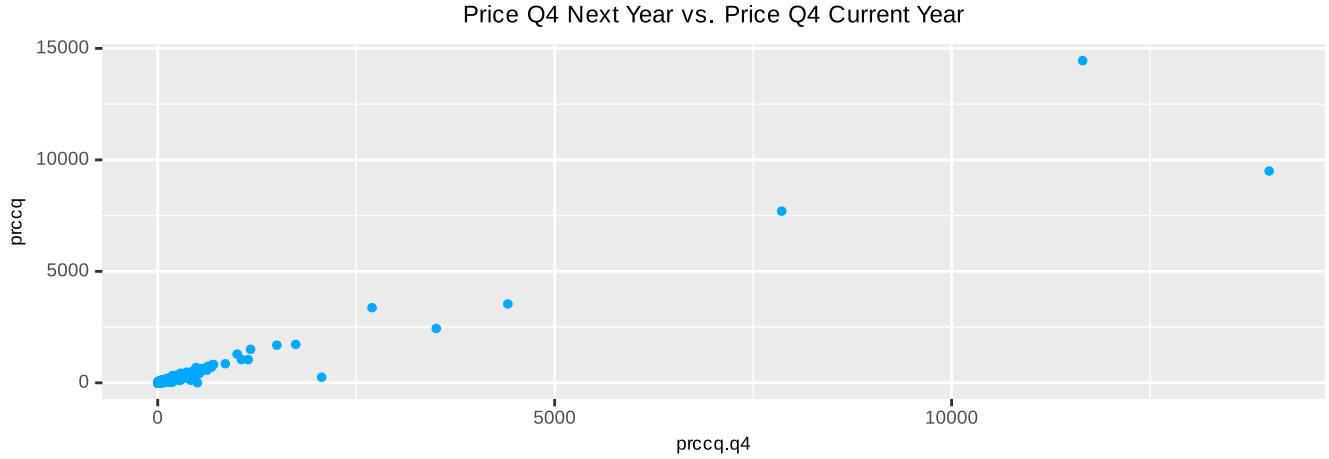
	mean	sd	median	min	max	range
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
prccq.q1	51.8391	295.4245	24.5750	3.0190	14000.00	13996.98
prccq.q2	52.9379	314.5788	24.2600	3.0000	14000.00	13997.00
prccq.q3	54.9956	350.7298	24.9200	3.0000	15500.00	15497.00
prccq.q4	54.7038	325.5355	24.6922	3.0000	14000.00	13997.00
prccq	49.5639	308.7802	20.9800	0.0250	14450.00	14449.98
growth	-0.1186	0.4689	-0.1492	-0.9956	10.24	11.24

3.2 Data Visualization

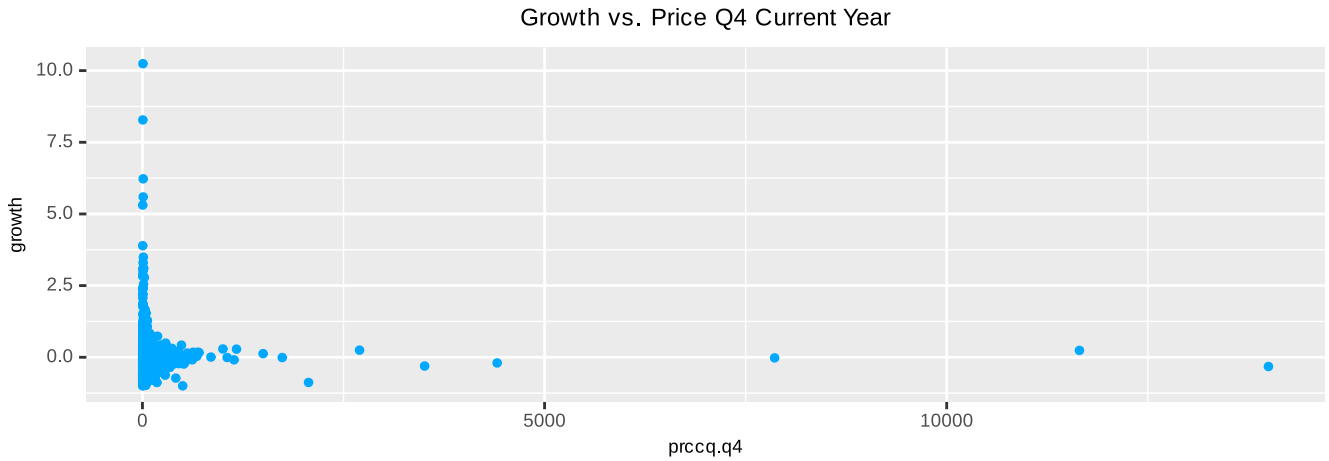
In [125]:
```
# Present a barchart to visualize growth across companies (sorted lowest to highest).
# You can use geom_col(aes(x=rank(growth, ties.method="first"), y=growth))
out(7,2.5)
ggplot(data) + geom_col(aes(x=rank(growth, ties.method="first"), y=growth)) +
ylab("growth") + xlab("company #") + ggtitle("12-Month Growth")
```



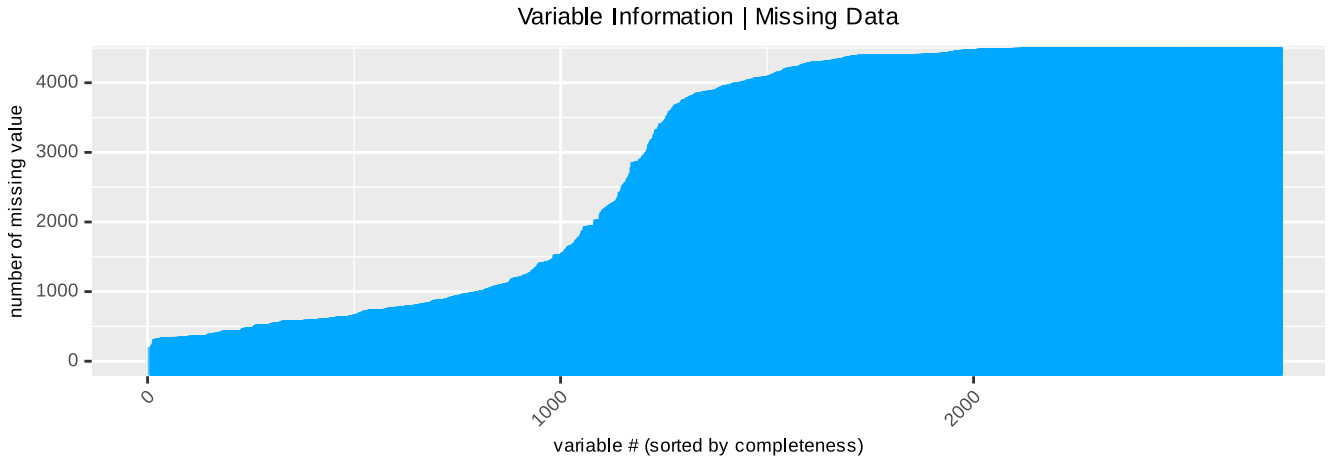
In [126]:
```
# Present a scatterplot to visualize correlation of price at Q4 next year vs. price at Q4 current year.
out(7,2.5)
ggplot(data) + geom_point(aes(x=prccq.q4, y=prccq)) +
ylab("prccq") + xlab("prccq.q4") + ggtitle("Price Q4 Next Year vs. Price Q4 Current Year")
```



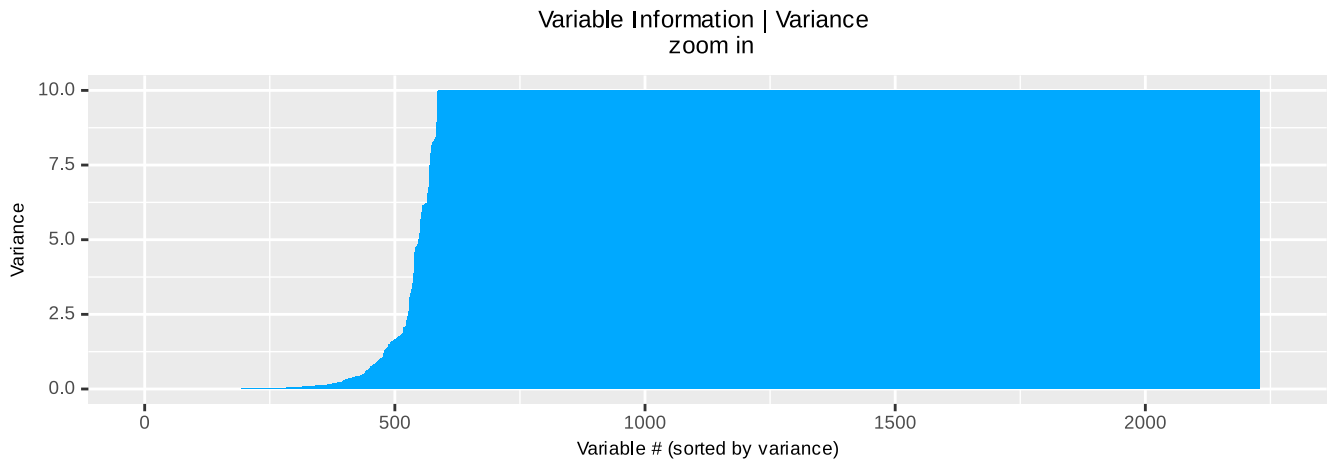
In [127]:
```
# Present a scatterplot to visualize correlation of growth vs. price at Q4 current year.
out(7,2.5)
ggplot(data) + geom_point(aes(x=prccq.q4, y=growth)) +
ylab("growth") + xlab("prccq.q4") + ggtitle("Growth vs. Price Q4 Current Year")
```

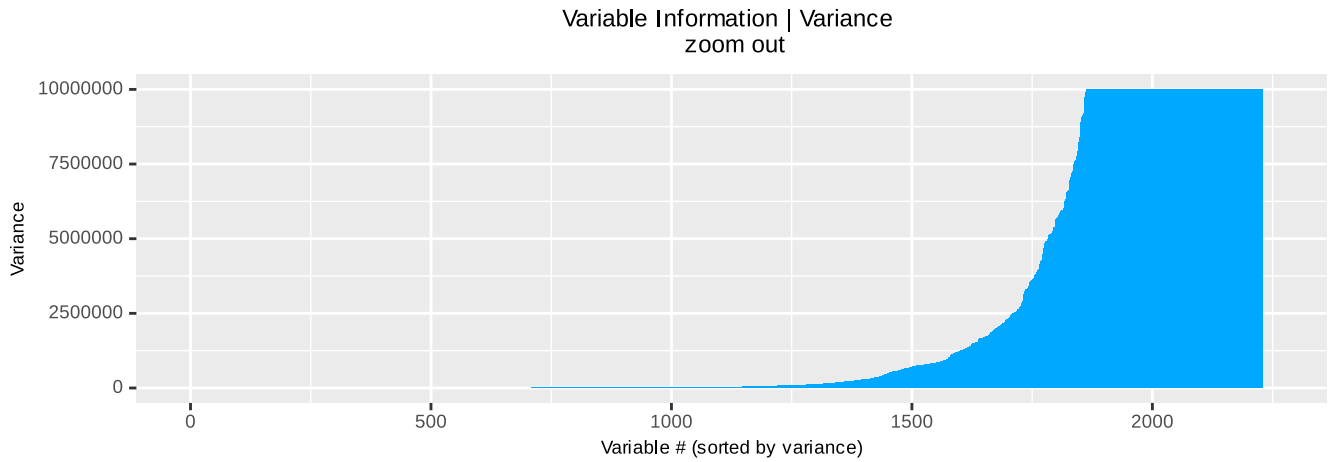


In [128]:
```
# Present a barchart to visualize the number of missing data across variables
# (variables on the horizontal axis sorted by most complete to least complete,
#  number of missing values on the vertical axis).
#
# You can use data.frame(na_count=as.numeric(summarize_all(data, ~sum(is.na(.)))))
# You can use geom_col(aes(x=rank(na_count, ties.method="first"), ...))
ggplot(data.frame(na_count=as.numeric(summarize_all(data, ~sum(is.na(.)))))) +
 geom_col(aes(x=rank(na_count, ties.method="first"), y = na_count), color=PALETTE[1]) +
ggtitle("Variable Information | Missing Data") + xlab("variable # (sorted by completeness)") +
ylab("number of missing value") + theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

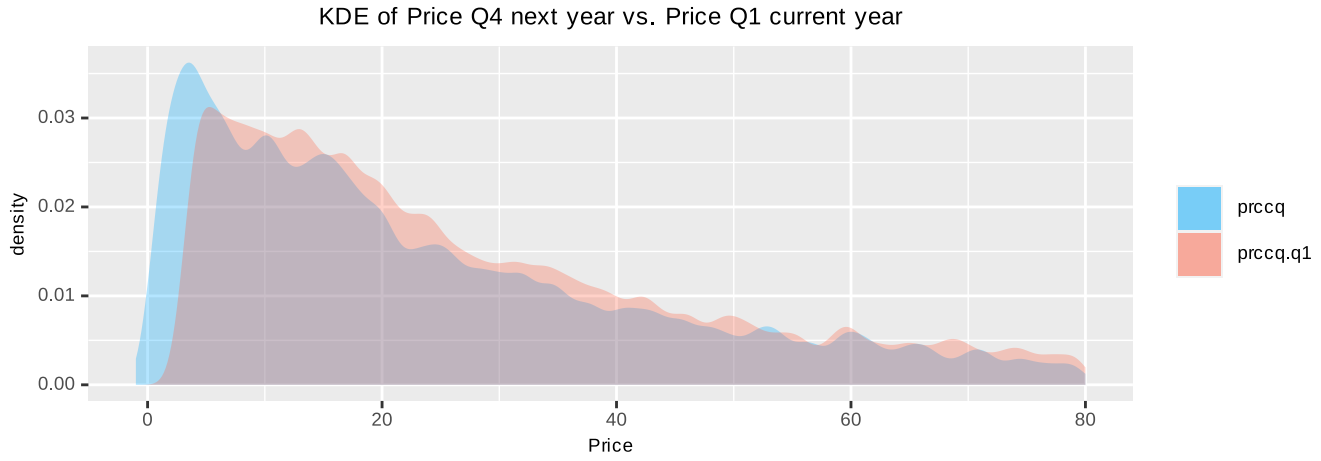


In [129]:
```
# Present a barchart to visualize variance across variables
# (only for numerical variables, sorted lowest to highest, zoom in to 0≤variance≤10, zoom out to 00≤variance≤1
#
# You can use select_if(..., ...)
# You can use as.numeric(summarize_all(..., ...))
# You can use pmin(..., ...)

data.col = data.frame(variance=as.numeric(summarize_all(select_if(data, ~is.numeric(.)), ~var(., na.rm=TRUE))))

# Create the barchart


ggplot(data.col) +
 geom_col(aes(x = rank(variance, ties.method = "first"), y = pmin(10, variance))) +
 xlim(0, 2250) +
 ylim(0, 10) +
 ggtitle("Variable Information | Variance \n zoom in") +
 xlab("Variable # (sorted by variance)") +
 ylab("Variance")

ggplot(data.col) +
 geom_col(aes(x = rank(variance, ties.method = "first"), y = pmin(10000000, variance))) +
 xlim(0, 2250) +
 ylim(0, 10000000) +
 ggtitle("Variable Information | Variance \n zoom out")+
xlab("Variable # (sorted by variance)") +
 ylab("Variance")
```

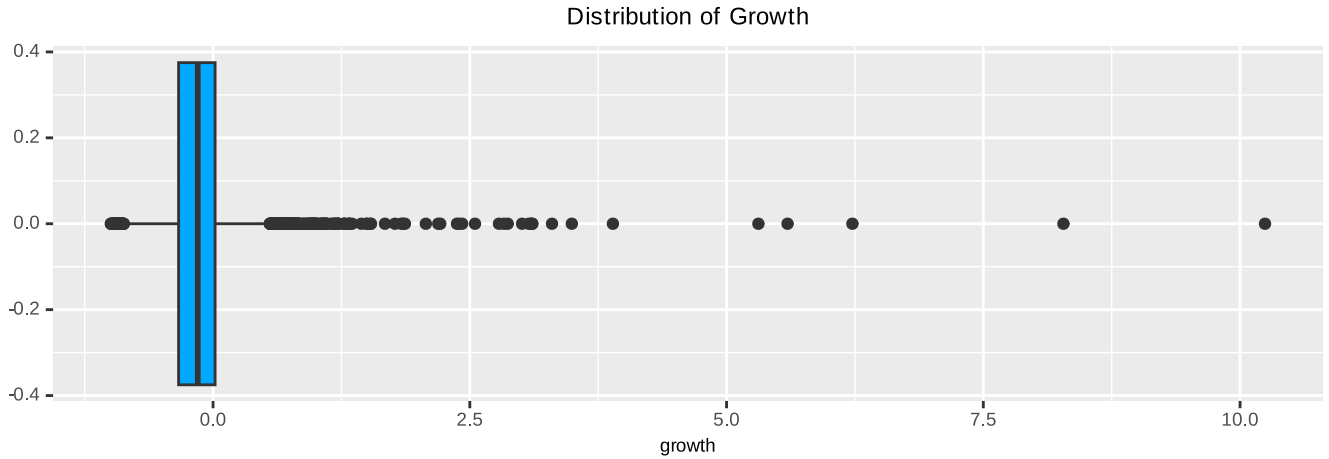

Variable Information | Variance
zoom in
10.0
7.5
5.0
2.5
0.0
Variance
0
500
1000
1500
2000
Variable # (sorted by variance)


Variable Information | Variance
zoom out
10000000
7500000
5000000
2500000
0
Variance
0
500
1000
1500
2000
Variable # (sorted by variance)

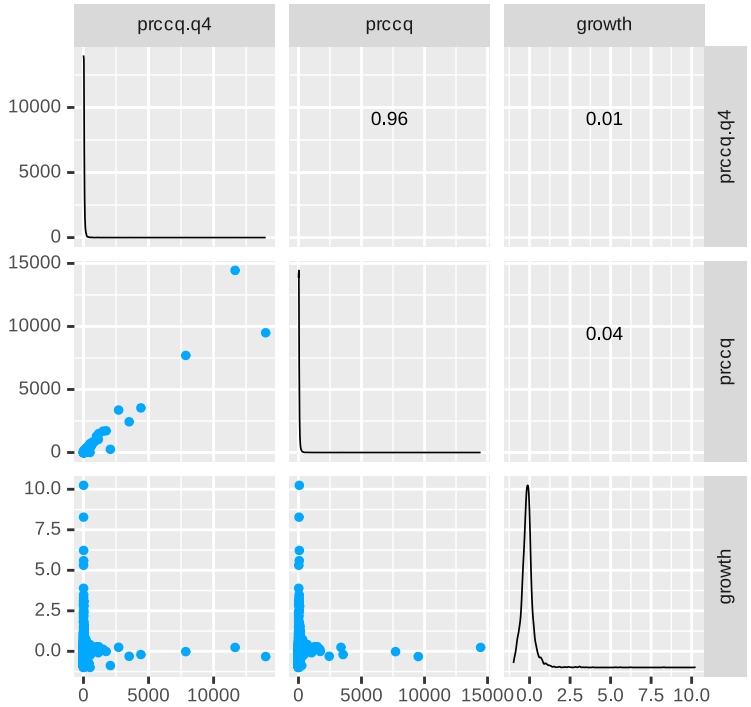
In [130]:
```
# Present an additional interesting data visualization.
out(7,2.5)
ggplot(data)+ xlim(-1,80) +
geom_density(aes(prccq, fill="prccq"), kernel="gaussian", bw=1, alpha = 0.3) +
geom_density(aes(prccq.q1,fill="prccq.q1"), kernel="gaussian", bw=1, alpha = 0.3) +
scale_color_manual(values=c("prccq"=PALETTE[1], "prccq.q1"=PALETTE[3])) +xlab("Price") +
ggtitle("KDE of Price Q4 next year vs. Price Q1 current year")
```



In [131]:
```
# Present an additional interesting data visualization.
ggplot(data) + geom_boxplot(aes(x = growth), fill=PALETTE[1]) +
ggtitle("Distribution of Growth")
```



In [132]:
```
# Present an additional interesting data visualization.
out(4,4)
ggscatmat(data[,c("prccq.q4","prccq","growth")])
```



3.3 Store Data

In [133]:
```
# Store the preapred data (4305 observations, 2714 variables)
write.csv(data, "My Prepared Data.csv", row.names=FALSE)
```