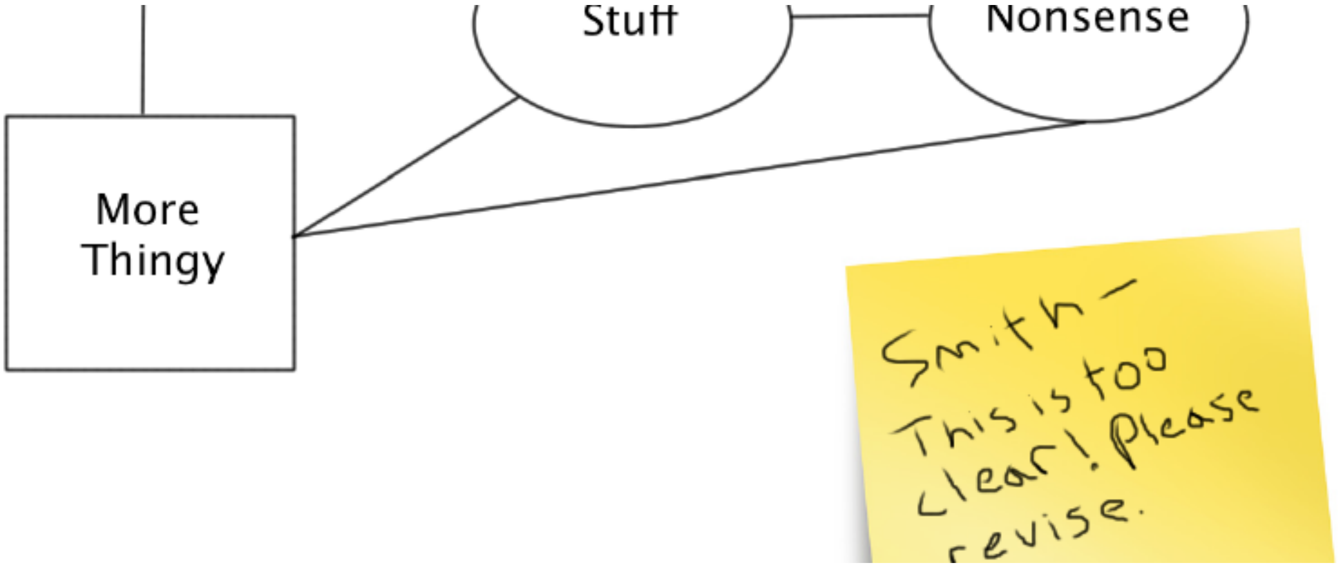
Data Modeling Overview

Data Science: Following the Data



An old adage in detective work is to "follow the money." In data science, one key to success is to "follow the data." In most cases, a data scientist will not help to design an information system from scratch. Instead, there will be several or many legacy systems where data resides; a big part of the challenge to the data scientist lies in integrating those systems.

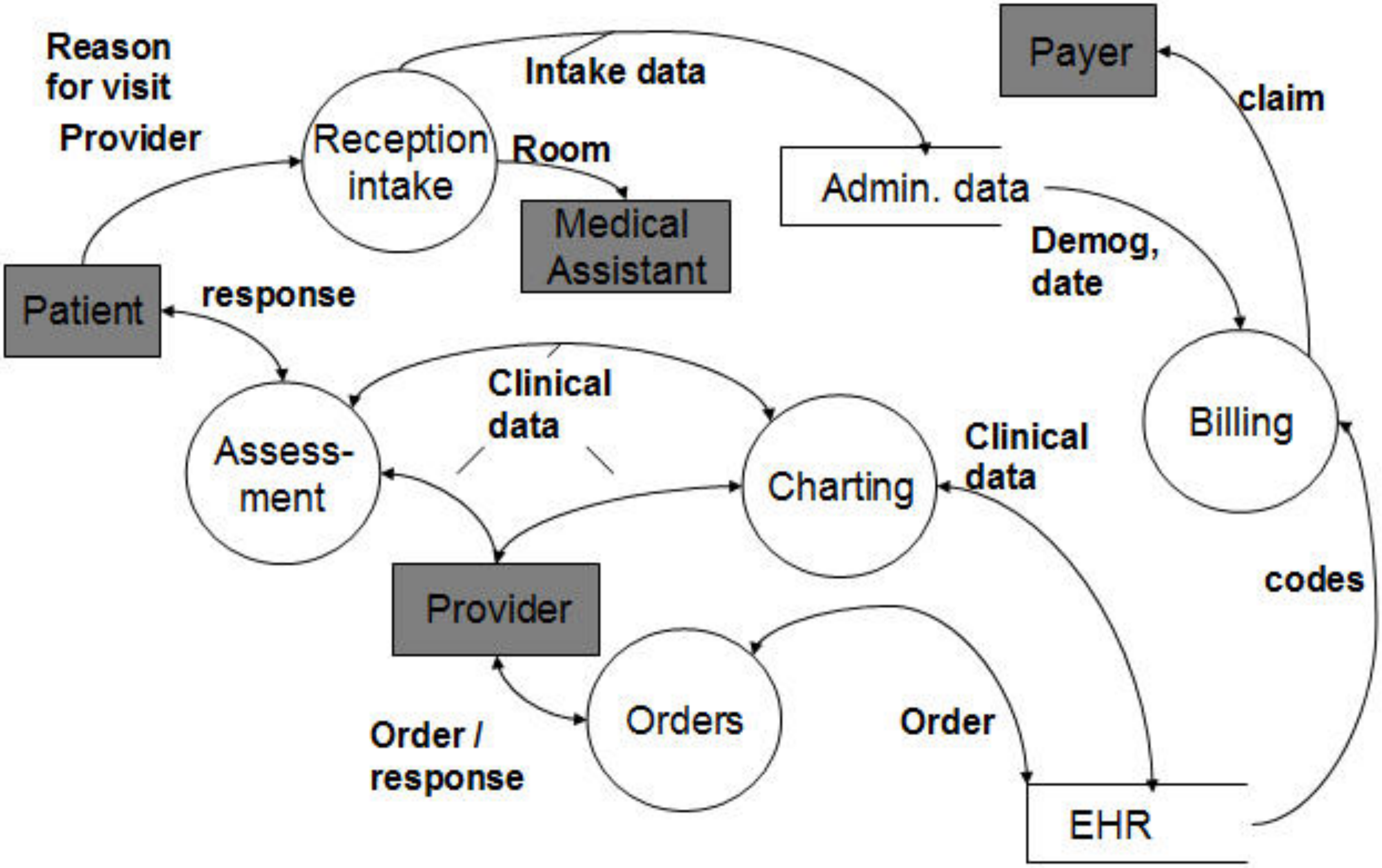
Introduction to Data Models & Systems

- Context for more functional use of R
  - Systems Analysis & Design 101
    - Process model (data flow diagram)
    - Data model (entity relationship diagram)
    - Data model (star schema)
    - Graphical user interface (GUI)

- Reference to the "systems type" pyramid

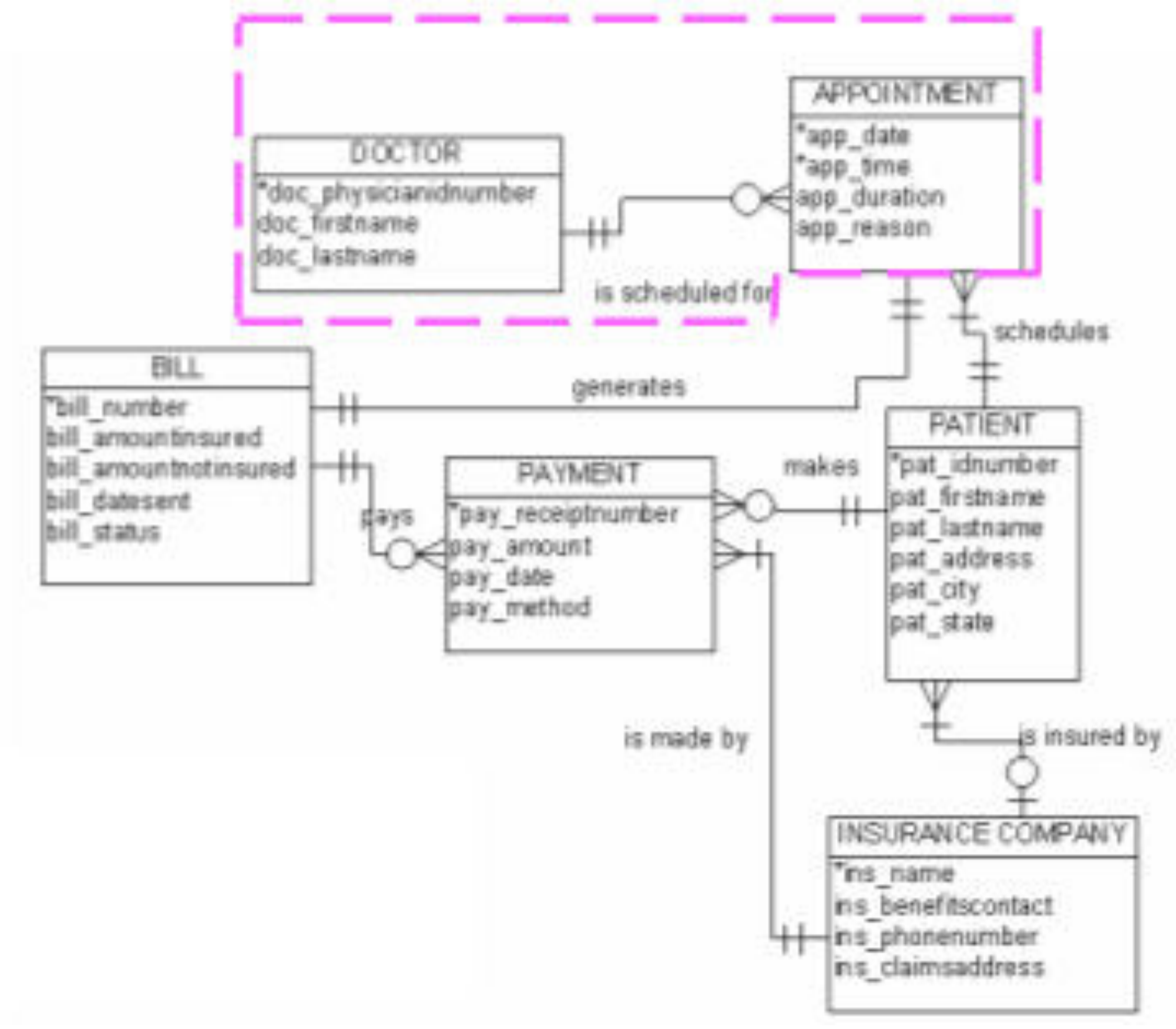
Note that might be a review for many of you

Data Flow Diagram (DFD)

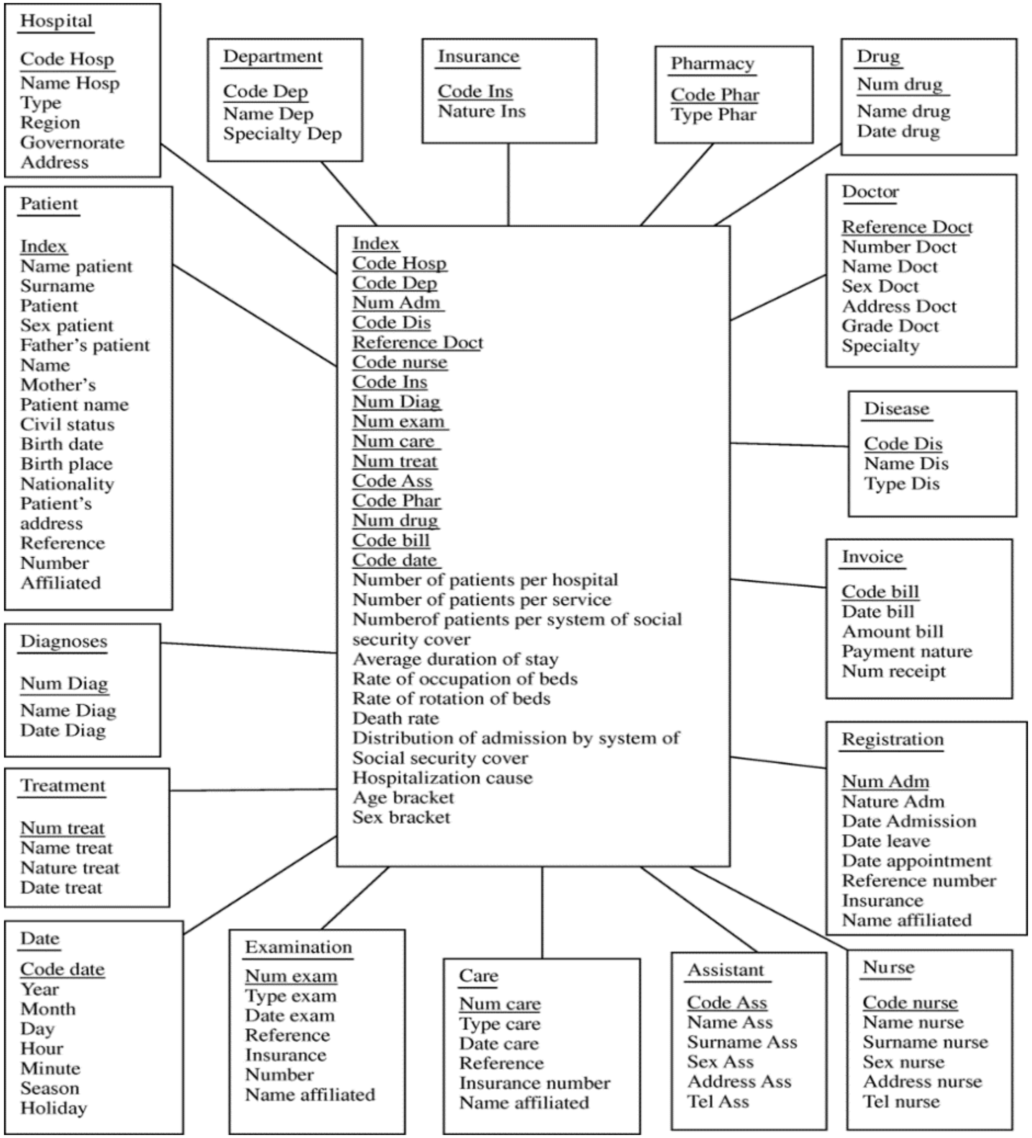


Entity Relationship Diagram (ERD)

A doctor can be scheduled for many appointments but may not have any scheduled at all. Each appointment is scheduled with exactly 1 doctor.



Star Schema



Hospital
Code Hosp
Name Hosp
Type
Region
Governorate
Address

Department
Code Dep
Name Dep
Specialty Dep

Insurance
Code Ins
Nature Ins

Pharmacy
Code Phar
Type Phar

Drug
Num drug
Name drug
Date drug

Patient
Index
Name patient
Surname
Patient
Sex patient
Father's patient
Name
Mother's
Patient name
Civil status
Birth date
Birth place
Nationality
Patient's
address
Reference
Number
Affiliated

Index
Code Hosp
Code Dep
Num Adm
Code Dis
Reference Doct
Code nurse
Code Ins
Num Diag
Num exam
Num care
Num treat
Code Ass
Code Phar
Num drug
Code bill
Code date
Number of patients per hospital
Number of patients per service
Numberof patients per system of social
security cover
Average duration of stay
Rate of occupation of beds
Rate of rotation of beds
Death rate
Distribution of admission by system of
Social security cover
Hospitalization cause
Age bracket
Sex bracket

Doctor
Reference Doct
Number Doct
Name Doct
Sex Doct
Address Doct
Grade Doct
Specialty

Disease
Code Dis
Name Dis
Type Dis

Invoice
Code bill
Date bill
Amount bill
Payment nature
Num receipt

Registration
Num Adm
Nature Adm
Date Admission
Date leave
Date appointment
Reference number
Insurance
Name affiliated

Diagnoses
Num Diag
Name Diag
Date Diag

Treatment
Num treat
Name treat
Nature treat
Date treat

Date
Code date
Year
Month
Day
Hour
Minute
Season
Holiday

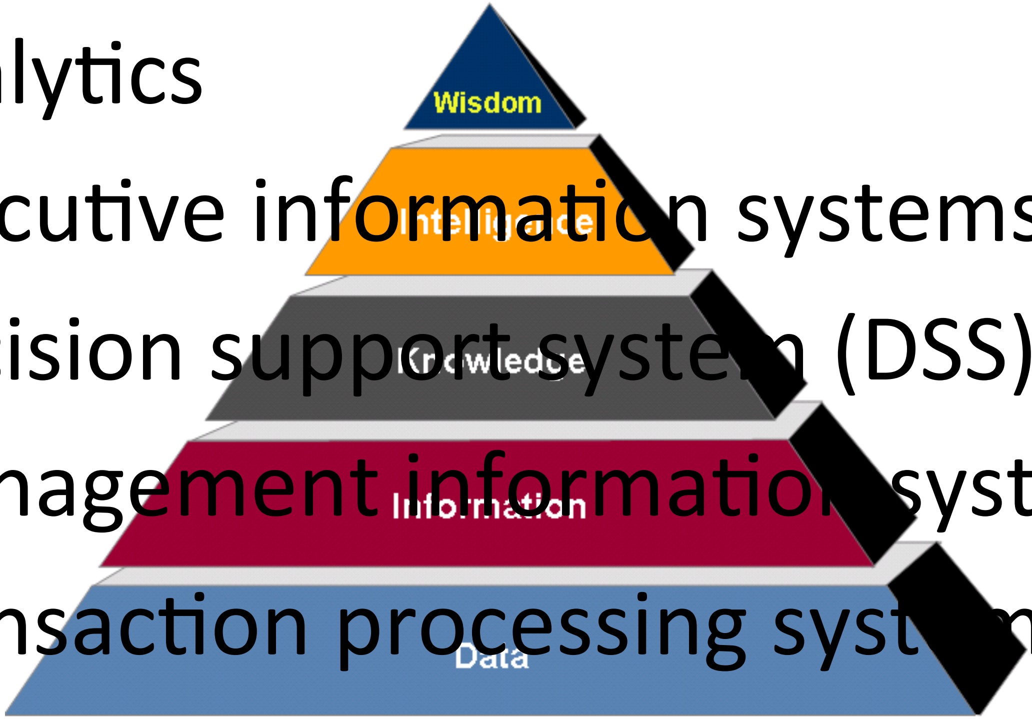
Examination
Num exam
Type exam
Date exam
Reference
Insurance
Number
Name affiliated

Care
Num care
Type care
Date care
Reference
Insurance number
Name affiliated

Assistant
Code Ass
Name Ass
Surname Ass
Sex Ass
Address Ass
Tel Ass

Nurse
Code nurse
Name nurse
Surname nurse
Sex nurse
Address nurse
Tel nurse

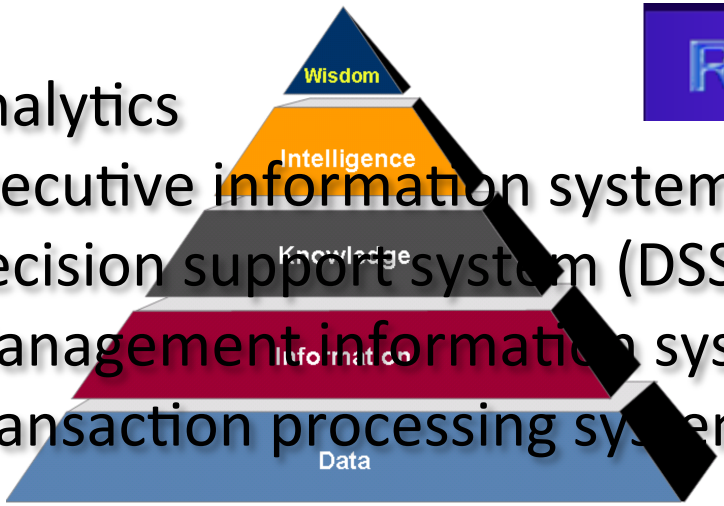
Information System Types



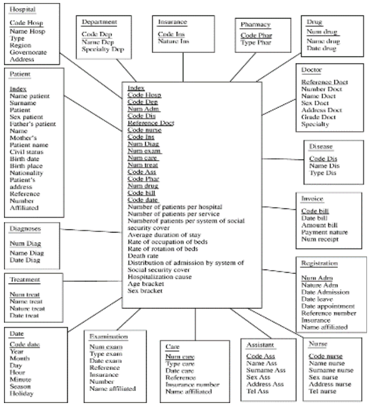
- Analytics
- Executive information systems (EIS)
- Decision support system (DSS)
- Management information system
- Transaction processing system

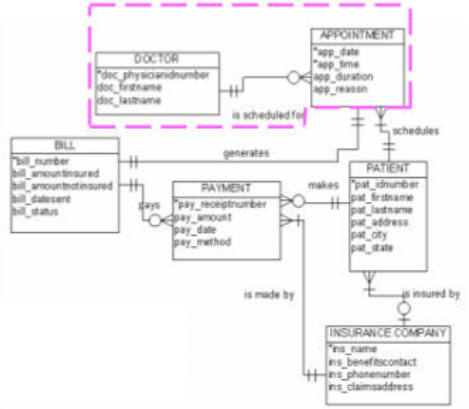
Data Science

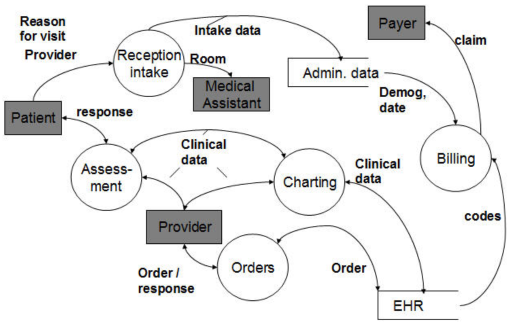
Information Systems Types



- Analytics
- Executive information systems (EIS)
- Decision support system (DSS)
- Management information system
- Transaction processing system







Question

Why do data modeling—why is it useful?

Rows and Columns

Rows and Columns

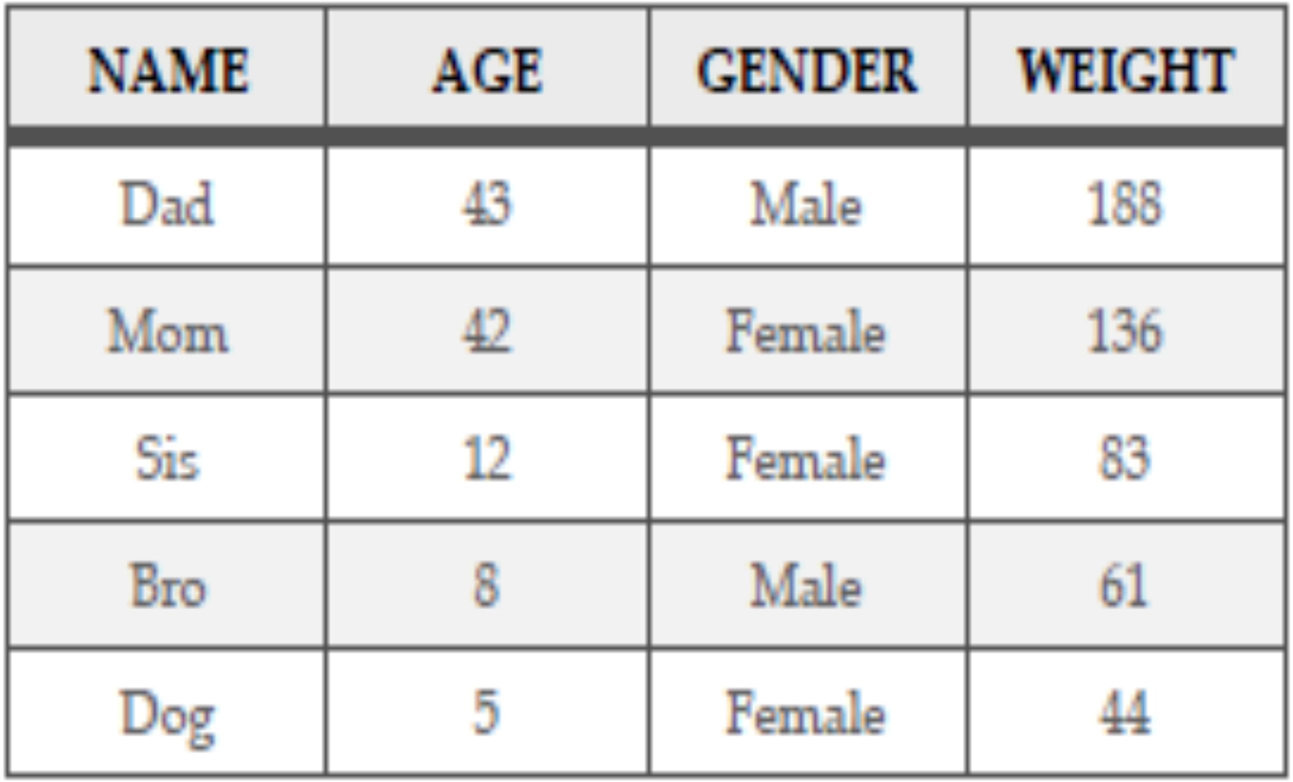
One of the most basic and widely used methods of representing data is to use rows and columns, where each row is a case or instance and each column is a variable and attribute. Most spreadsheets arrange their data in rows and columns, although spreadsheets don't usually refer to these as cases or variables. R represents rows and columns in an object called a *data frame*.

Thinking About Data

- Know your data
  - Context
  - Content
  - Mode

- Data organization to facilitate R analysis
  - Rows and columns
  - Consistent mode type by attribute/variable

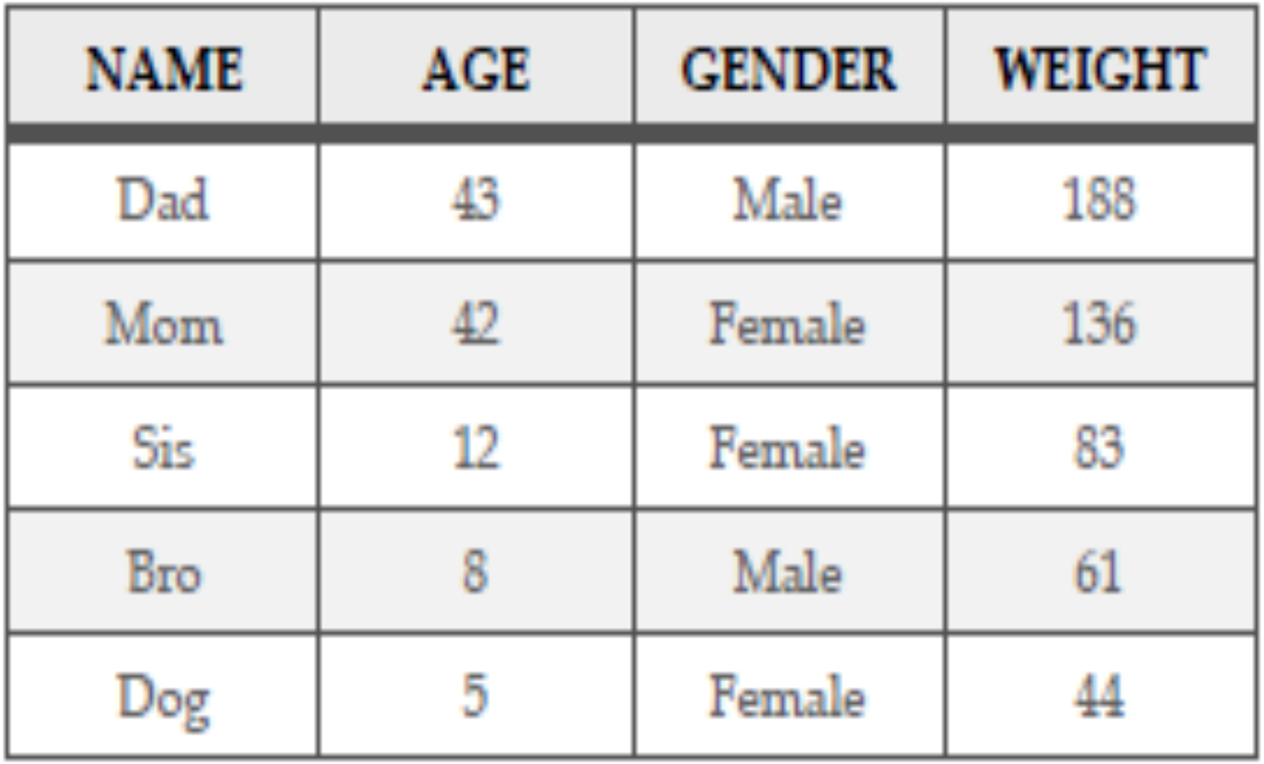
An Example Dataset: Context



NAME	AGE	GENDER	WEIGHT
Dad	43	Male	188
Mom	42	Female	136
Sis	12	Female	83
Bro	8	Male	61
Dog	5	Female	44

An Example Dataset: Characteristics

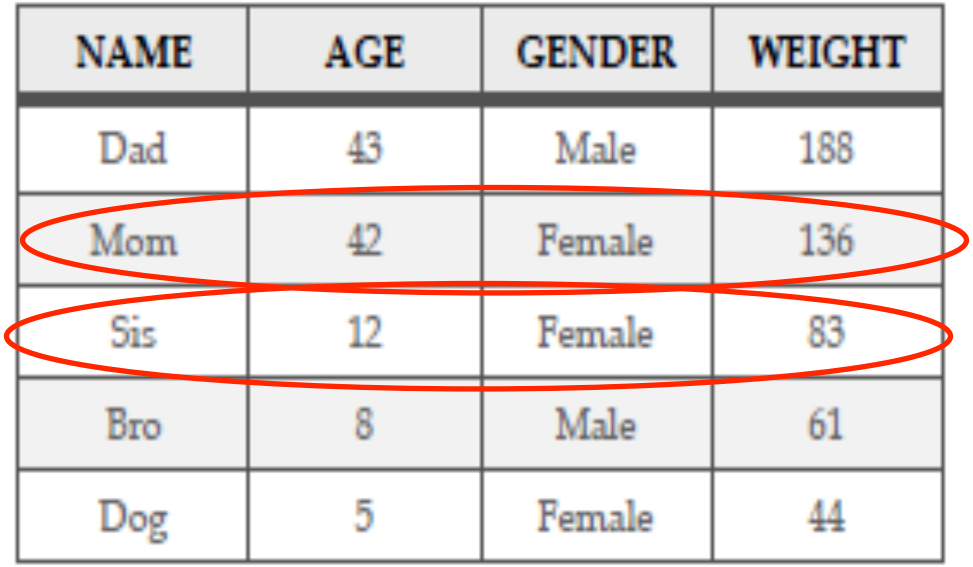
Two-dimensions: rows and columns



NAME	AGE	GENDER	WEIGHT
Dad	43	Male	188
Mom	42	Female	136
Sis	12	Female	83
Bro	8	Male	61
Dog	5	Female	44

An Example Dataset: Characteristics

- Rows (data)
  - Cases
  - Instances
  - Observations

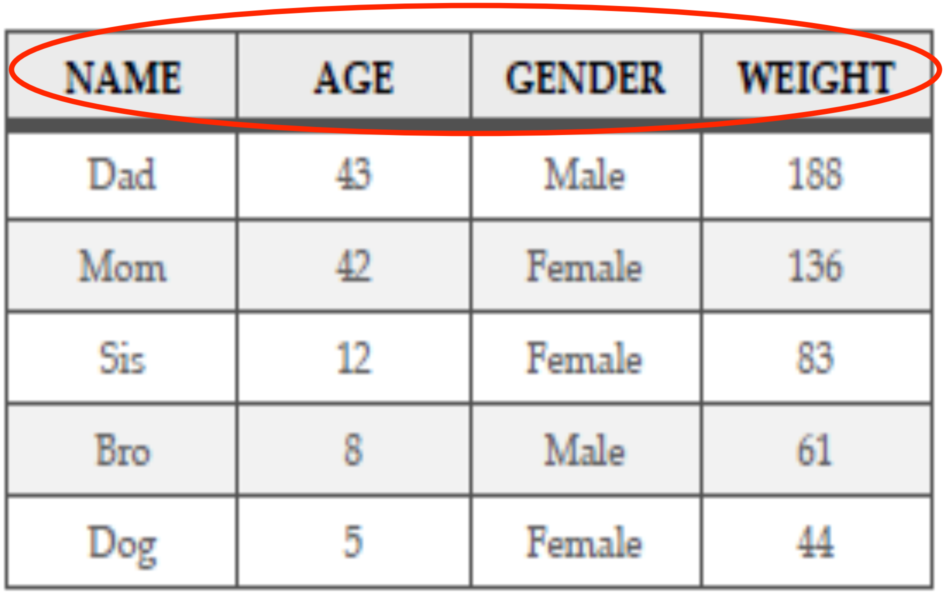


NAME	AGE	GENDER	WEIGHT
Dad	43	Male	188
Mom	42	Female	136
Sis	12	Female	83
Bro	8	Male	61
Dog	5	Female	44

Note: Name Age Gender Weight is **not** a data row.

An Example Dataset: Characteristics

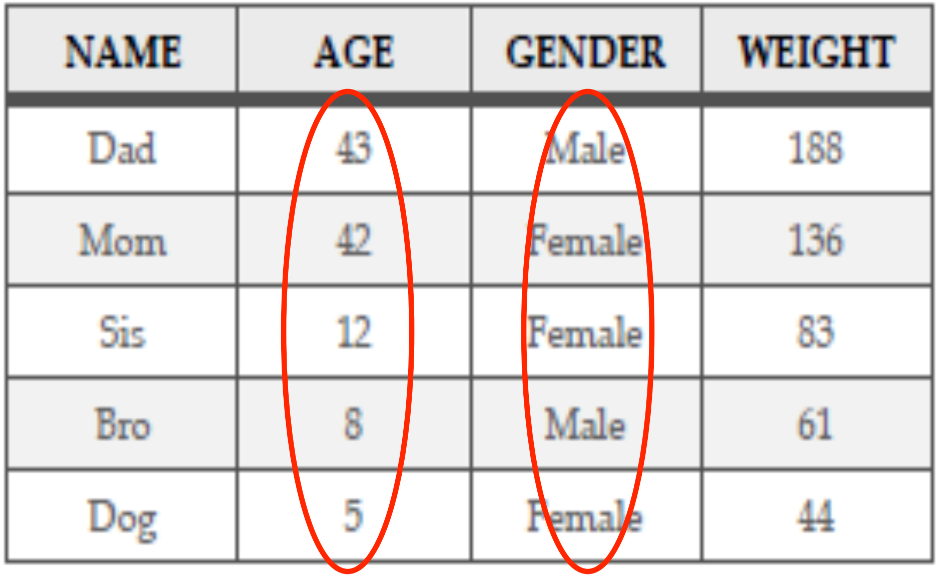
- Columns (data)
  - **Variable name**
  - Attributes
  - Variables



NAME	AGE	GENDER	WEIGHT
Dad	43	Male	188
Mom	42	Female	136
Sis	12	Female	83
Bro	8	Male	61
Dog	5	Female	44

An Example Dataset: Characteristics

- Columns (data)
  - **Attributes**
  - **Variables**
  - Variable name

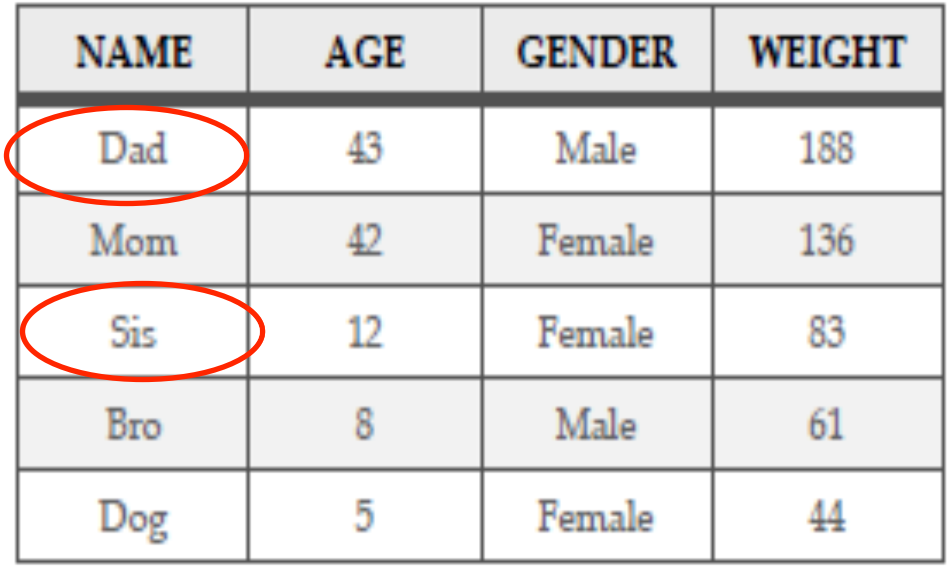


NAME	AGE	GENDER	WEIGHT
Dad	43	Male	188
Mom	42	Female	136
Sis	12	Female	83
Bro	8	Male	61
Dog	5	Female	44

- Note: Name Age Gender Weight are **not** data

An Example Dataset: Characteristics

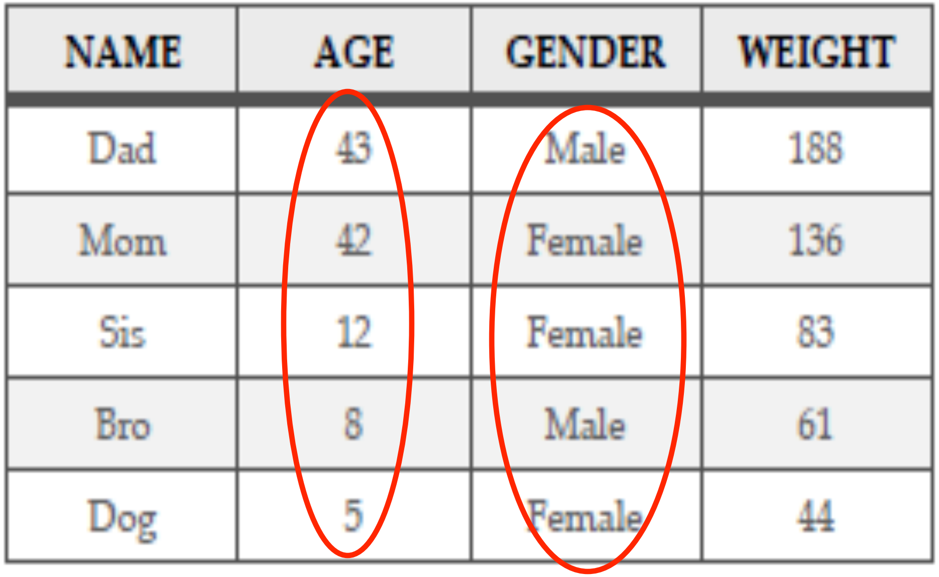
- Each row has a unique identifier (case label).



NAME	AGE	GENDER	WEIGHT
Dad	43	Male	188
Mom	42	Female	136
Sis	12	Female	83
Bro	8	Male	61
Dog	5	Female	44

An Example Dataset: Characteristics

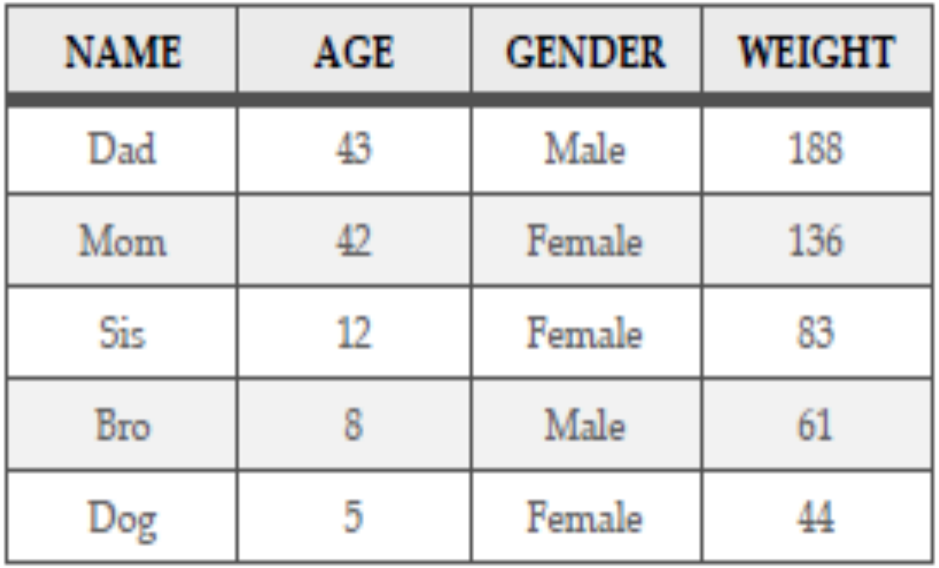
- Each column has the same type/mode of data.
- Each column has the same number of entries.



NAME	AGE	GENDER	WEIGHT
Dad	43	Male	188
Mom	42	Female	136
Sis	12	Female	83
Bro	8	Male	61
Dog	5	Female	44

Creating a dataset in R

- Data set: How does this get built in R?
  - Create a vector for each variable (column).
  - Create a data frame to combine individual vectors.



NAME	AGE	GENDER	WEIGHT
Dad	43	Male	188
Mom	42	Female	136
Sis	12	Female	83
Bro	8	Male	61
Dog	5	Female	44

Question:

- How would you represent the following data in a data frame?

  – Students in a class
    - For each student, we have a student ID and a GPA.
    - Student 1: ID: N1; GPA: 3.8
    - Student 2: ID: N2; GPA: 4.0
    - Student 3: ID: N3; GPA: 3.3
    - Student 4: ID: N4; GPA: 3.5
    - Student 5: ID: N5; GPA: 3.9

→Create a grid (table) to show how you would represent this information. (Submit a simple table as a spreadsheet.)

Answer:

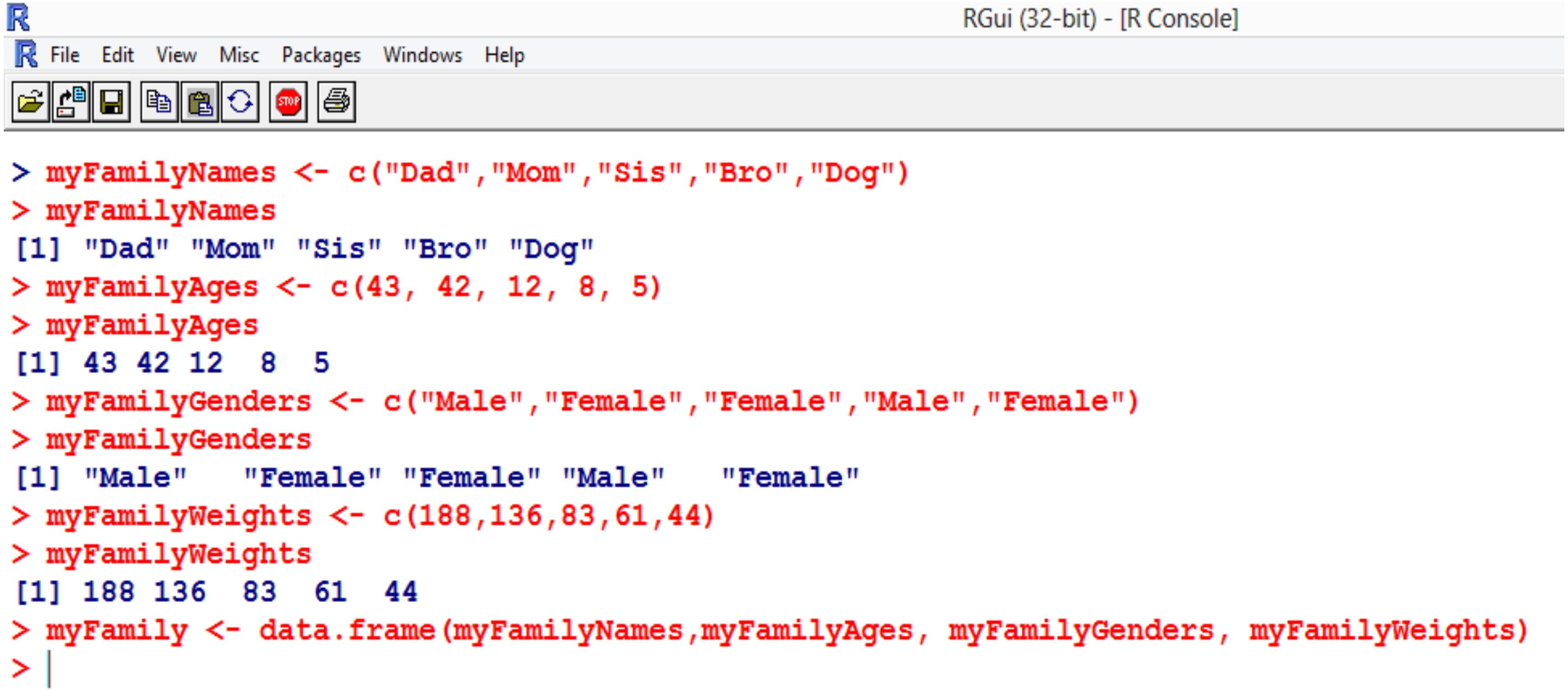
- How would you represent the following data in a data frame?

Student ID	Student GPA
N1	3.8
N2	4.0
N3	3.3
N4	3.5
N5	3.9

Data Frames in R

Creating a Dataframe in R

The respective variable columns have been built as vectors and displayed below.



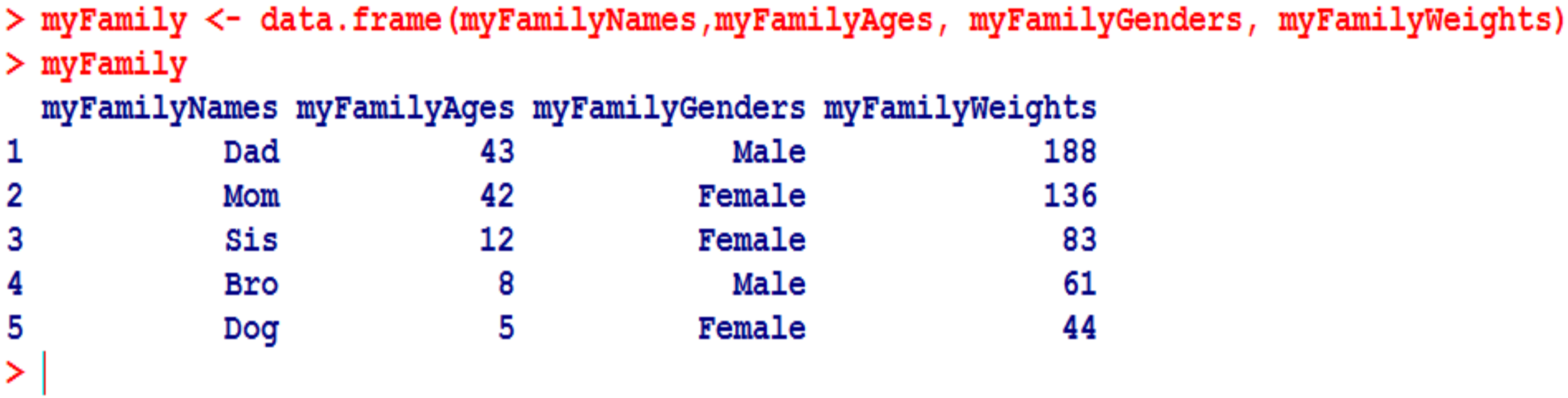
RGui (32-bit) - [R Console]
File Edit View Misc Packages Windows Help

```
> myFamilyNames <- c("Dad","Mom","Sis","Bro","Dog")
> myFamilyNames
[1] "Dad" "Mom" "Sis" "Bro" "Dog"
> myFamilyAges <- c(43, 42, 12, 8, 5)
> myFamilyAges
[1] 43 42 12  8  5
> myFamilyGenders <- c("Male","Female","Female","Male","Female")
> myFamilyGenders
[1] "Male"   "Female" "Female" "Male"   "Female"
> myFamilyWeights <- c(188,136,83,61,44)
> myFamilyWeights
[1] 188 136  83  61  44
> myFamily <- data.frame(myFamilyNames,myFamilyAges, myFamilyGenders, myFamilyWeights)
> 
```

The columns have been combined and assigned a label via the data.frame function.

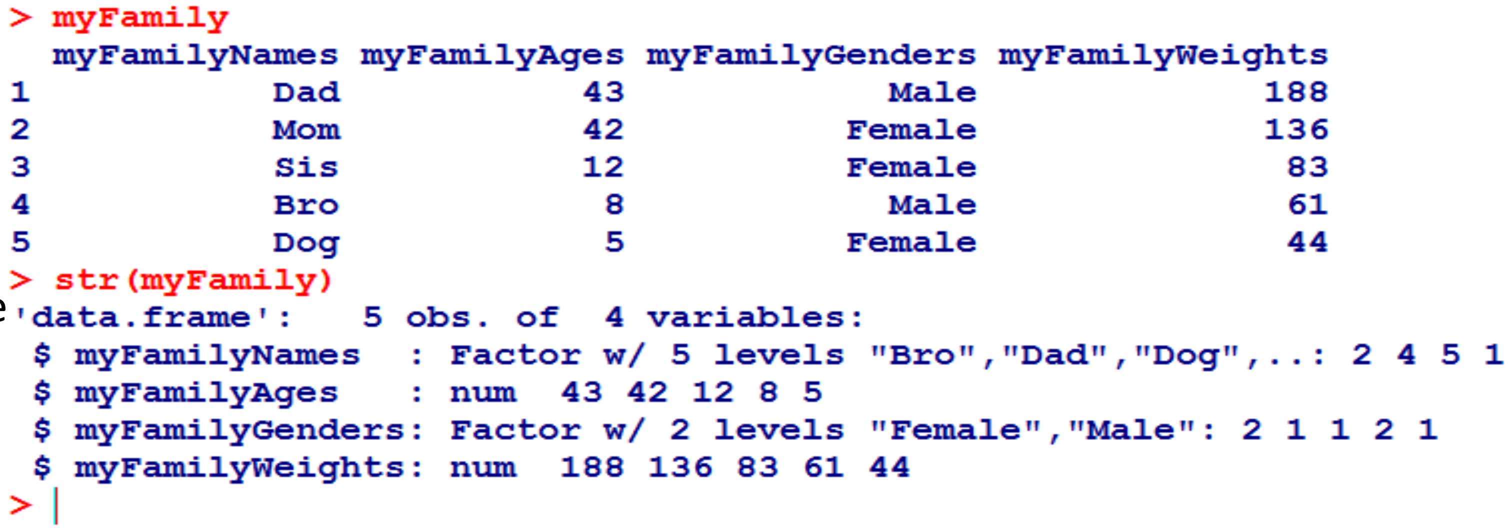
Viewing a Dataframe

Display the contents of the data object MyFamily.



```
> myFamily <- data.frame(myFamilyNames,myFamilyAges, myFamilyGenders, myFamilyWeights)
> myFamily
  myFamilyNames myFamilyAges myFamilyGenders myFamilyWeights
1           Dad           43            Male             188
2           Mom           42          Female             136
3           Sis           12          Female              83
4           Bro            8            Male              61
5           Dog            5          Female              44
> 
```

Using the R "Str" (Structure) Command



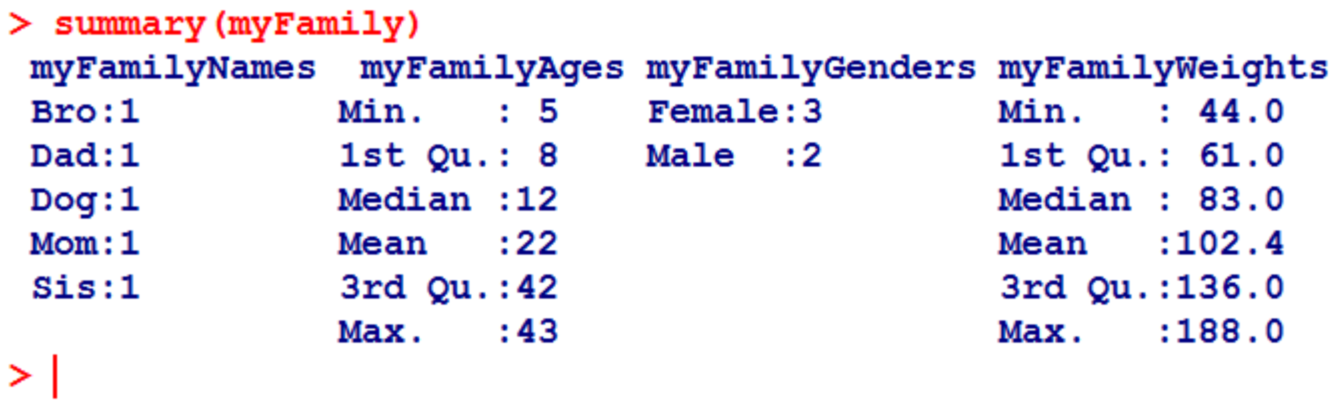
```
> myFamily
  myFamilyNames myFamilyAges myFamilyGenders myFamilyWeights
1           Dad           43            Male             188
2           Mom           42          Female             136
3           Sis           12          Female              83
4           Bro            8            Male              61
5           Dog            5          Female              44
> str(myFamily)
'data.frame':   5 obs. of  4 variables:
 $ myFamilyNames  : Factor w/ 5 levels "Bro","Dad","Dog",..: 2 4 5 1
 $ myFamilyAges   : num  43 42 12 8 5
 $ myFamilyGenders: Factor w/ 2 levels "Female","Male": 2 1 1 2 1
 $ myFamilyWeights: num  188 136 83 61 44
> |
```

What does the structure function tell us about the data object myFamily?

- Confirmation that MyFamily is a data frame;
- MyFamily has five observations (cases/instances) and four variables.
- "$" for each variable/component column with descriptive information.
- Each of the variables has a mode or type (same mode within a variable/column).
- Variable is either a "factor" or "num".
- "Factor" variable has a "level".
- "Level" describes the options within a variable.
- "num" variable indicates "numeric".

Using the R Summary Command

What does the summary function tell us about the data object myFamily?



```
> summary(myFamily)
 myFamilyNames  myFamilyAges myFamilyGenders myFamilyWeights
 Bro:1         Min.   : 5    Female:3        Min.   : 44.0  
 Dad:1         1st Qu.: 8    Male  :2        1st Qu.: 61.0  
 Dog:1         Median :12                    Median : 83.0  
 Mom:1         Mean   :22                    Mean   :102.4  
 Sis:1         3rd Qu.:42                    3rd Qu.:136.0  
               Max.   :43                    Max.   :188.0  
> 
```

- “Factor” variables list variable names (MyFamilyNames, myFamilyGenders, MyFamilyWeights) along with the number of occurrences of cases that are coded within that factor.

- Numeric variables have six different calculated quantities that help summarize the variable:
  - Min—minimum or lowest value of all cases
  - 1st Qu—dividing line at the top of the 1st quartile
  - Median—value of the case that splits the whole group in half
  - Mean—numeric average
  - 3rd Qu—3rd quartile
  - Max—max value of all cases

Accessing Dataframes as a Matrix

```
# returns the data in the first row and first column
> myFamily[1,1]

#Returns the first row
> myFamily[1,]

#Returns the first column
> myFamily[,1]

#Returns everything but the first row (deletes first row)
> myFamily[-1,]

#Returns everything but the first column
> myFamily[,-1]
```

R Takeaways

- A **vector** is a list of elements/things
- All the vectors things are the same type (**mode**)
- Data is in a **rectangular format** (rows & columns)
- A **data frame** stores these rectangular data sets
- **data.frame**() organizes vectors into a data frame
- **str**() and **summary**() provide info on a data frame
- A **factor** organizes groups of observations
- **Quartiles** divide a sorted vector into 4 groups.
- Min() and max() measure "**dispersion**"
- Mean() and median() measure "**central tendency**"