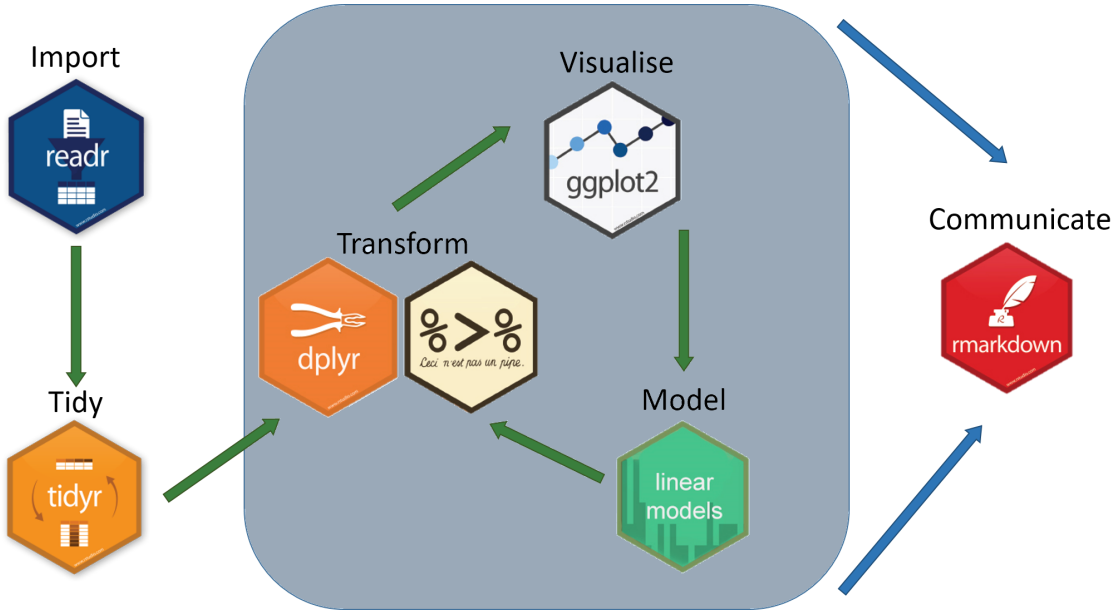
Data Analysis Process



Make Tidy Codes with

Pipes - %>% ... %>%

The pipe operator implemented first in **magrittr** package.

```
library(magrittr)
```

Makes code readable by:

- structuring sequences of data operations
- avoiding nested function calls,
- minimizing the need for local variables and function definitions

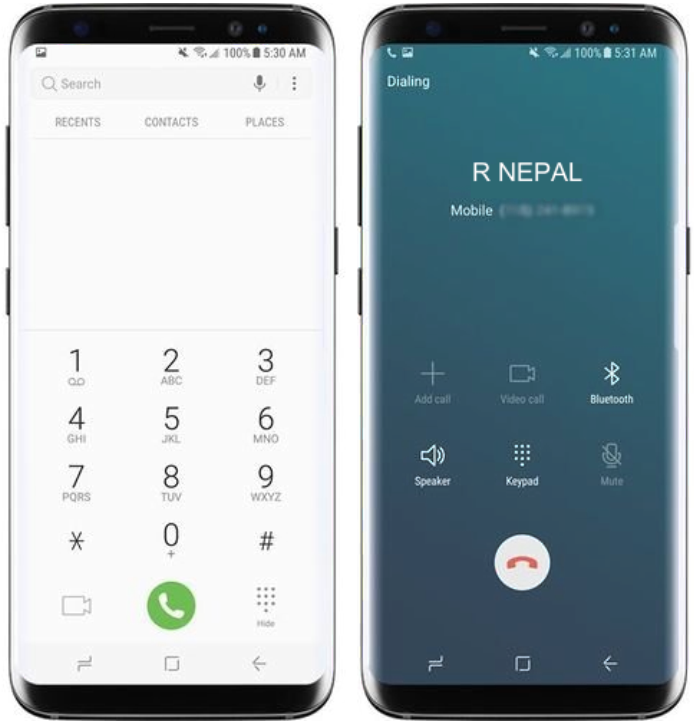
Example: Without using Pipes

Assume, you need to make a call.

- get phone,
- type number,
- make call,
- end call

Code:

```
end_call(
  make_call(
    type_number(
      get_phone , "984344***"
      ), "Hello"
    )
  )
```



How does Pipes work?

```
get_phone %>%
type_number("9843440863") %>%
make_call("Hello") %>%
end_call()
```

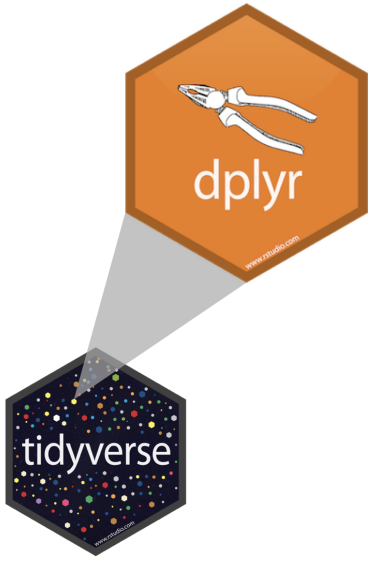
Here,

- **%>%** is pipe operator
- Shortcut: **Ctrl + Shift + m**

Data Manipulation with

Grammar of Data Manipulation

dplyr is a grammar of data manipulation that provides verbs (function) which solves common data manipulation problems.



Function	Description
select()	picks variables based on their names.
filter()	picks cases based on their values.
mutate()	adds new variables that are functions of existing variables
group_by()	groups the variables by columns
summarise()	reduces multiple values down to a single summary.
arrange()	changes the ordering of the rows.

dplyr - Rules for functions

```
library(dplyr)
```

- First argument is always a data frame
- Latter arguments performs as per functions
- Always return a data frame
- Don't modify in place

Load data

```
library(readr)

gapminder <- read_csv("gapminder.csv")
```

country	continent	year	lifeExp	pop	gdpPercap
Afghanistan	Asia	1952	28.801	8425333	779.4453
Afghanistan	Asia	1957	30.332	9240934	820.8530
Afghanistan	Asia	1962	31.997	10267083	853.1007
Afghanistan	Asia	1967	34.020	11537966	836.1971
Afghanistan	Asia	1972	36.088	13079460	739.9811
Afghanistan	Asia	1977	38.438	14880372	786.1134

Select

```
country_gdp_year <- gapminder %>%
  select(year, country, gdpPercap)
```

Here ,

- **select()** is a dplyr function.
- **gapminder** is a data
- **year, country, gdpPercap** are selected data

year	country	gdpPercap
1952	Afghanistan	779.4453
1957	Afghanistan	820.8530
1962	Afghanistan	853.1007
1967	Afghanistan	836.1971
1972	Afghanistan	739.9811
1977	Afghanistan	786.1134

Filter by value

```
gapminder_asia <- gapminder %>%
  filter(continent == "Asia")
```

Here ,

- **filter()** is a dplyr function.
- **gapminder** is a data set
- **continent** is a case which should equal to value "**Asia**"

country <chr>	continent <chr>	year <dbl>	lifeExp <dbl>
Afghanistan	Asia	1952	28.801
Afghanistan	Asia	1957	30.332
Afghanistan	Asia	1962	31.997
Afghanistan	Asia	1967	34.020
Afghanistan	Asia	1972	36.088

5 rows | 1-4 of 6 columns

Logical Operators in R

operator	definition	operator	definition
<	less than	x	y
<=	less than or equal to	x & y	x AND y
>	greater than	is.na(x)	test if x is NA
>=	greater than or equal to	x %in% y	test if x is in y
==	exactly equal to	!(x %in% y)	test if x is not in y
!=	not equal to	!x	not x

02:00

Your Turn - 01

Question:

Display country-wise, annual life expectancy of Europe.

Hints :

1. Filter **continent** by **"Europe"**
2. Use **%>%** to combine two functions
3. Then, select variables - **year, country, lifeExp**

year	country	lifeExp
1952	Albania	55.23
1957	Albania	59.28
1962	Albania	64.82
1967	Albania	66.22
1972	Albania	67.69
1977	Albania	68.93

Answer - 01

```
lifeexp_year_europe <- gapminder %>%
  filter(continent == "Europe") %>%
  select(year, country, lifeExp)
```

Here,

- **%>%** is a pipe operator
- **gapminder** is a data
- **filter()** and **select()** are dplyr function

year	country	lifeExp
1952	Albania	55.23
1957	Albania	59.28
1962	Albania	64.82
1967	Albania	66.22
1972	Albania	67.69

Mutate - Add New Variables

```
gapminder_total_gdp <- gapminder %>%
  mutate(gdp = gdpPercap * pop)
```

country	continent	year	lifeExp	pop	gdpPercap	gdp
Afghanistan	Asia	1952	28.801	8425333	779.4453	6567086330
Afghanistan	Asia	1957	30.332	9240934	820.8530	7585448670
Afghanistan	Asia	1962	31.997	10267083	853.1007	8758855797
Afghanistan	Asia	1967	34.020	11537966	836.1971	9648014150
Afghanistan	Asia	1972	36.088	13079460	739.9811	9678553274

02:00

Your Turn - 02

Find each countries in **Asia** with life expectancy in **2007** and rank of country's **life expectancy**.

Hint:

1. Filter **continent** by "**Asia**" & **year** by **2007**
2. Select variables: **country, lifeExp**
3. Mutate to create **rank** variable using **min_rank()** on **lifeExp**

country	lifeExp	rank
Afghanistan	43.828	1
Bahrain	75.635	25
Bangladesh	64.062	7
Cambodia	59.723	3
China	72.961	20

Answer - 02

```
asia_lifeExp <- gapminder %>%
  filter(
    continent == "Asia",
    year == 2007
  ) %>%
  select(country, lifeExp) %>%
  mutate(rank = min_rank(lifeExp))
```

country	lifeExp	rank
Afghanistan	43.828	1
Bahrain	75.635	25
Bangladesh	64.062	7
Cambodia	59.723	3
China	72.961	20

Group By - One or more variable

```
gapminder_group_by <- gapminder %>%
  filter(year == 2007) %>%
  group_by(continent)
```

country	continent	year	lifeExp	pop	gdpPercap
Afghanistan	Asia	2007	43.828	31889923	974.5803
Albania	Europe	2007	76.423	3600523	5937.0295
Algeria	Africa	2007	72.301	33333216	6223.3675
Angola	Africa	2007	42.731	12420476	4797.2313
Argentina	Americas	2007	75.320	40301927	12779.3796

Arrange - Ascending Order

```
gapminder_asc <- gapminder %>%
  filter(year == 2007) %>%
  arrange(pop)
```

country	continent	year	lifeExp	pop	gdpPercap
Sao Tome and Principe	Africa	2007	65.528	199579	1598.435
Iceland	Europe	2007	81.757	301931	36180.789
Djibouti	Africa	2007	54.791	496374	2082.482
Equatorial Guinea	Africa	2007	51.579	551201	12154.090
Montenegro	Europe	2007	74.543	684736	9253.896

Arrange - Descending Order

```
gapminder_asc <- gapminder %>%
  filter(year == 2007) %>%
  arrange(desc(pop))
```

country	continent	year	lifeExp	pop	gdpPercap
China	Asia	2007	72.961	1318683096	4959.115
India	Asia	2007	64.698	1110396331	2452.210
United States	Americas	2007	78.242	301139947	42951.653
Indonesia	Asia	2007	70.650	223547000	3540.652
Brazil	Americas	2007	72.390	190010647	9065.801

02:00

Your Turn - 03

- Find countries in **Asia** with low ranked **gdpPercap** in **1992**

Hint:

1. Filter **continent** by "**Asia**" & **year** by **1992**
2. Select variables: **country, gdpPercap**
3. Mutate to create **rank** variable using **min_rank()** on **gdpPercap**
4. Arrange **rank** by ascending order

country	gdpPercap	rank
Myanmar	347.0000	1
Afghanistan	649.3414	2
Cambodia	682.3032	3
Bangladesh	837.8102	4
Nepal	897.7404	5

Answer - 03

```
asia_gdpPercap_1992 <- gapminder %>%
  filter(
    continent == "Asia",
    year == 1992
  ) %>%
  select(country, gdpPercap) %>%
  mutate(
    rank = min_rank(gdpPercap)
  ) %>%
  arrange(rank)
```

Asian Countries with low gdpPercap in 1992

country	gdpPercap	rank
Myanmar	347.0000	1
Afghanistan	649.3414	2
Cambodia	682.3032	3
Bangladesh	837.8102	4
Nepal	897.7404	5

Summarize

```
gampinder_sum <- gapminder %>%
  filter(year == 2007) %>%
  group_by(continent) %>%
  summarise(meanlife = mean(lifeExp))
```

Here,

- **group_by** - groups by variables
- **summarise** - reduce multiple variables

continent	meanlife
Africa	54.80604
Americas	73.60812
Asia	70.72848
Europe	77.64860
Oceania	80.71950