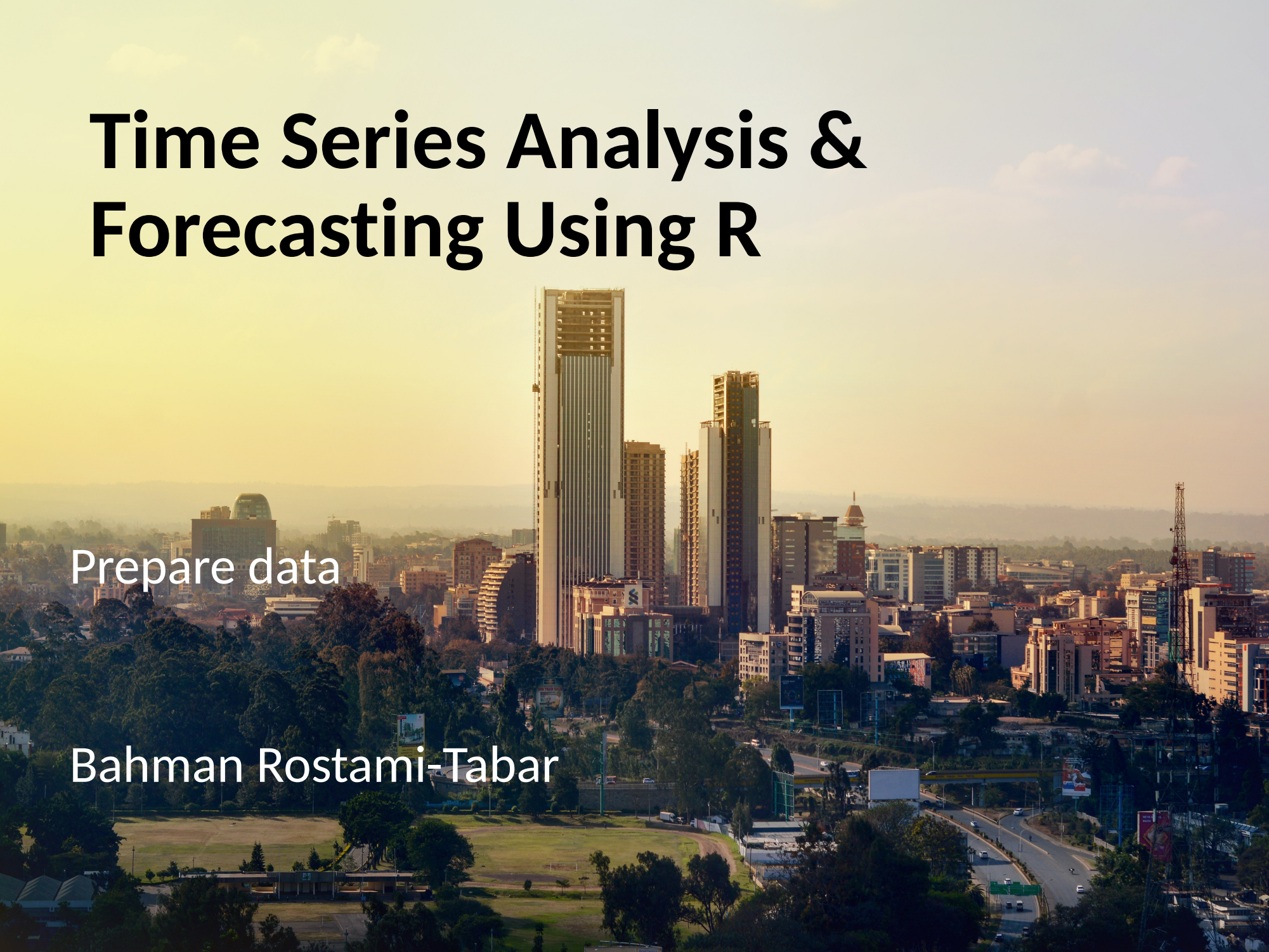
Time Series Analysis & Forecasting Using R



Prepare data

Bahman Rostami-Tabar

Outline

1. Learning outcomes
2. Time series in R
3. Example: create and work with `tsibble`
4. Lab Session 1

Outline

1. Learning outcomes
2. Time series in R
3. Example: create and work with `tsibble`
4. Lab Session 1

Learning outcomes

You should be able to:

1. Create `tsibble` objects in R to work with time series data
2. Use `tsibble` functions to prepare data for time series analysis & forecasting
3. Work with `tsibble` and tidyverse functions

Outline

Time series

A `time series` can be thought of as a list of numbers (the measurements), along with some information about what times those numbers were recorded (the index). This information can be stored as an object in R.

Time series data

- Four-yearly Olympic winning times
- Annual Google profits
- Quarterly Australian beer production
- Monthly rainfall
- Weekly retail sales
- Daily IBM stock prices
- Hourly electricity demand
- 5-minute freeway traffic counts
- Time-stamped stock transaction data

Class packages

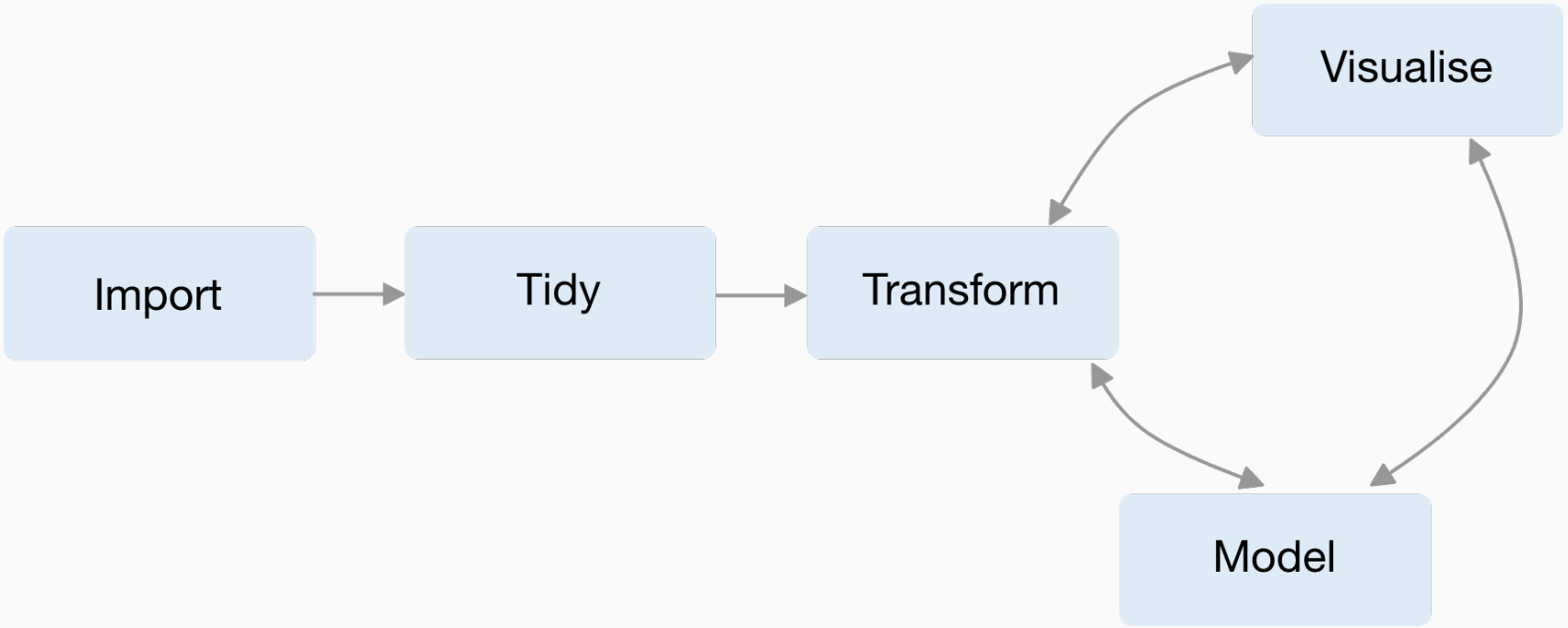
```
# Data manipulation and plotting functions
library(tidyverse)
# Time series manipulation
library(tsibble)
# Forecasting functions
library(fable)
# Time series graphics and statistics
library(feasts)
# Tidy time series data
library(tsibbledata)
```

Class packages

```
# Data manipulation and plotting functions
library(tidyverse)
# Time series manipulation
library(tsibble)
# Forecasting functions
library(fable)
# Time series graphics and statistics
library(feasts)
# Tidy time series data
library(tsibbledata)
```

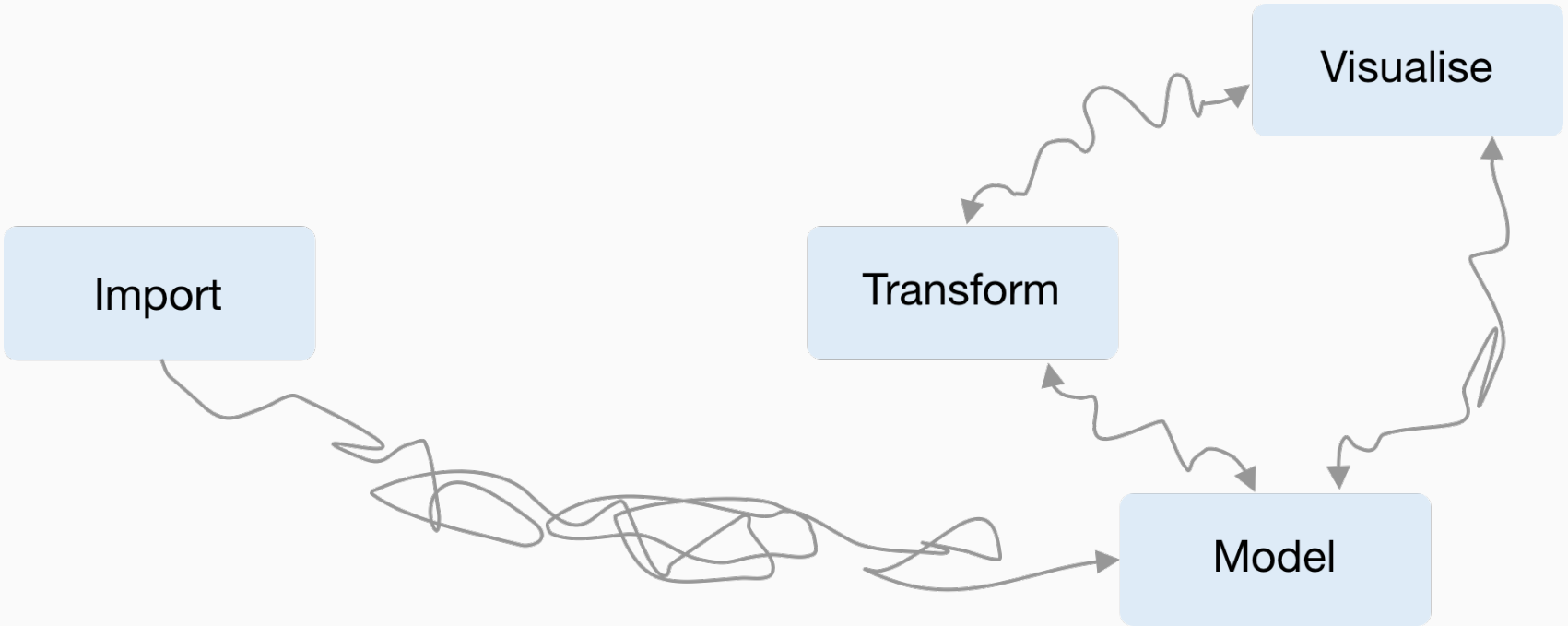
```
# All of the above and more
library(fpp3)
```

Tidyverse



Time series objects in R for forecasting

- does not work with `ts()`, `zoo()`, `xts()`, `etc`
- difficult to work with tidyverse



Features of data

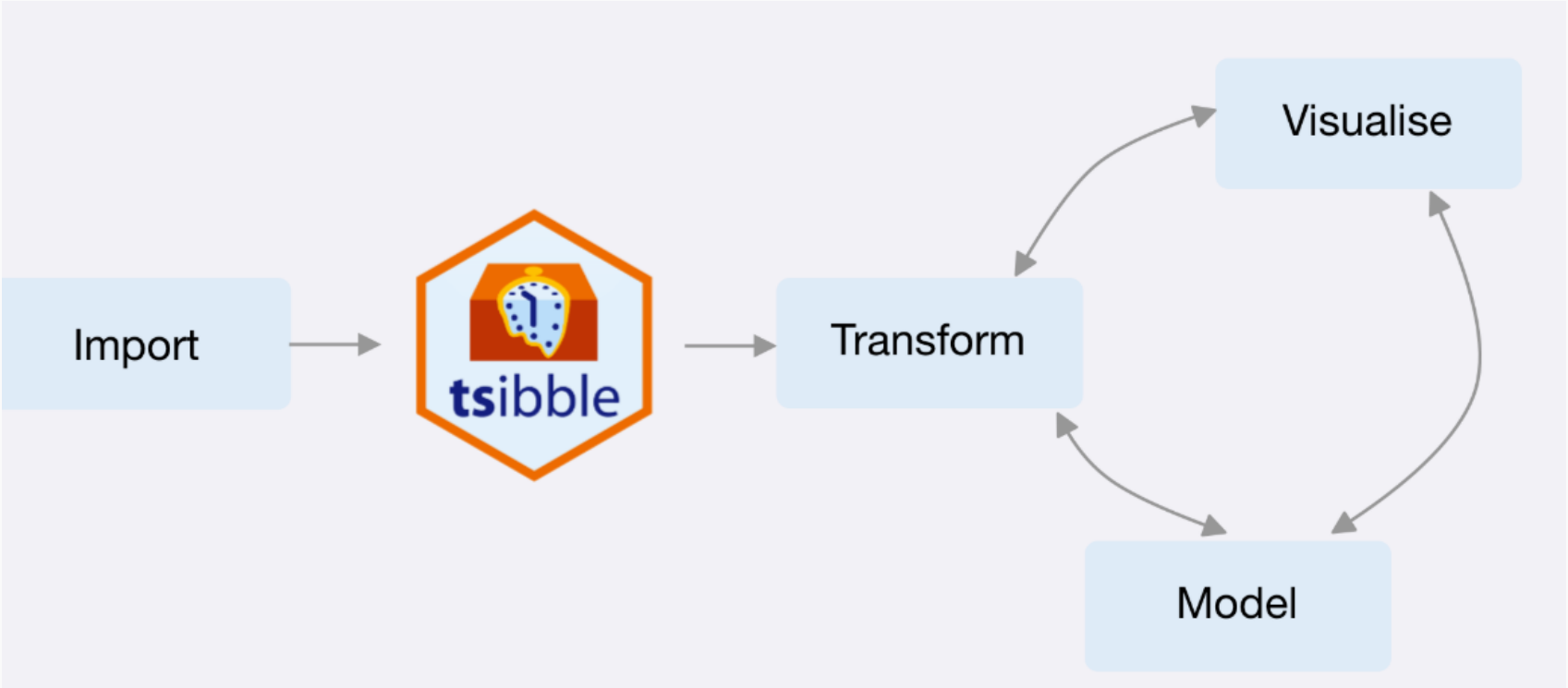
- heterogeneous data types
- irregular time interval
- multiple measured variables
- multiple grouping variables

The key to many time series

- Most time series can be naturally disaggregated using a series of factors known as keys
- These keys are used to uniquely identify separate time series, each of which can be modelled separately.
- This structure allows batch time series analysis & forecasting to be applied across many time series.
- Estimating multiple models is a key feature

Tsibble package

It defines tidier data for temporal analysis



`tsibble` objects

In tsibble:

- An index: time information about the observation
- Measured variable(s): numbers of interest
- Key variable(s): set of variables that define observational units over time
- It works with tidyverse functions.

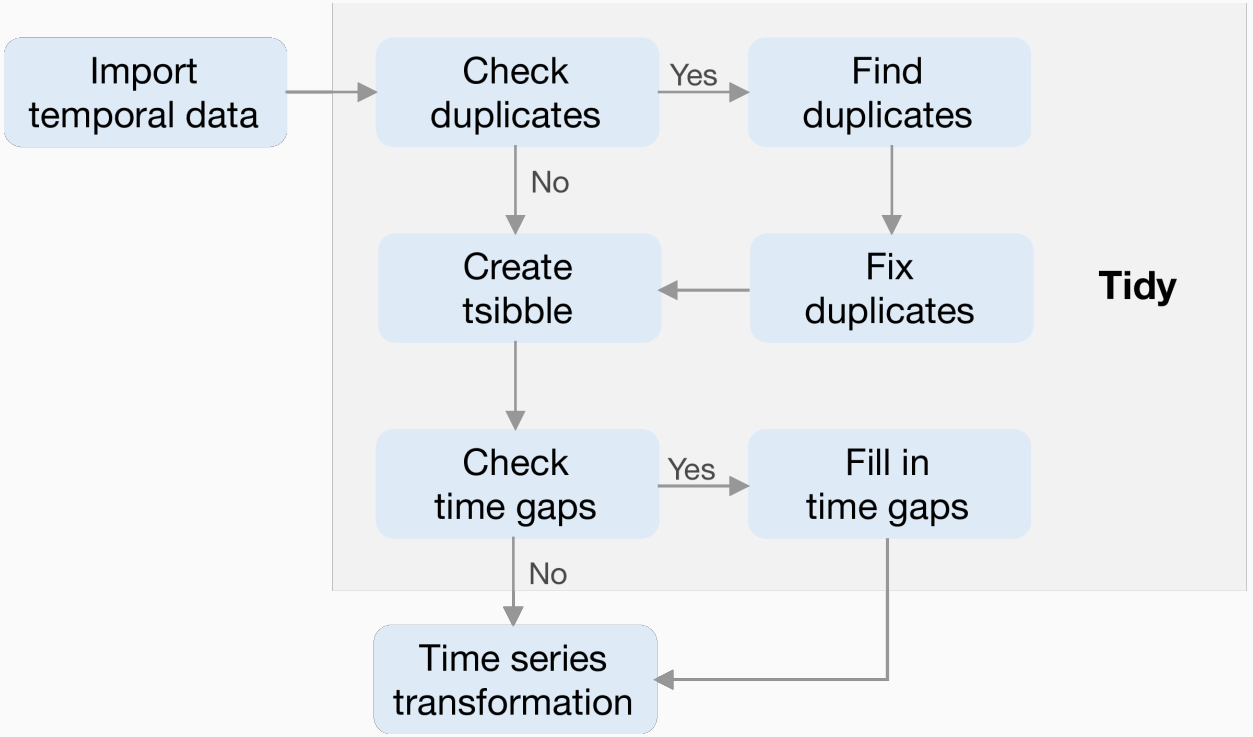
The `tsibble` index

Common time index variables can be created with these functions:

Frequency	Function
Annual	start:end
Quarterly	yearquarter()
Monthly	yearmonth()
Weekly	yearweek()
Daily	as_date(), ymd()
Sub-daily	as_datetime()

Outline

Steps to create a tsibble



Read a csv file

quarterly overnight trips across Australia

```
tourism <- readxl::read_excel("data/tourism.xlsx")
```

```
## # A tibble: 24,320 x 5
##    Quarter    Region   State          Purpose Trips
##    <chr>      <chr>    <chr>          <chr>   <dbl>
##  1 1998-01-01 Adelaide South Austra~ Busine~  135.
##  2 1998-04-01 Adelaide South Austra~ Busine~  110.
##  3 1998-07-01 Adelaide South Austra~ Busine~  166.
##  4 1998-10-01 Adelaide South Austra~ Busine~  127.
##  5 1999-01-01 Adelaide South Austra~ Busine~  137.
##  6 1999-04-01 Adelaide South Austra~ Busine~  200.
##  7 1999-07-01 Adelaide South Austra~ Busine~  169.
##  8 1999-10-01 Adelaide South Austra~ Busine~  134.
##  9 2000-01-01 Adelaide South Austra~ Busine~  154.
## 10 2000-04-01 Adelaide South Austra~ Busine~  169.
## # ... with 24,310 more rows
```

Check duplicates

```
tourismd <- tourism %>% duplicated()
sum(tourismd)
```

```
## [1] 0
```

```
#are_duplicated()
#tourism %>% distinct()
```

Change index to yearquarter

```
tourism <- tourism %>%
  mutate(Quarter = yearquarter(Quarter))
```

```
## # A tibble: 24,320 x 5
##    Quarter Region   State           Purpose  Trips
##      <qtr> <chr>    <chr>           <chr>    <dbl>
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
## 10 2000 Q2 Adelaide South Australia Business  169.
## # ... with 24,310 more rows
```

Craete a tsibble

```
tourism <- tourism %>%
  as_tsibble(
    index = Quarter,
    key = c(Region, State, Purpose)
  )
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State           Purpose  Trips
##      <qtr> <chr>    <chr>           <chr>    <dbl>
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
```

Check gaps

```
tourism %>% has_gaps()
tourism %>% count_gaps()
tourism %>% scan_gaps()
tourism %>% fill_gaps(Trips=0L)
```

tsibble objects

```
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State           Purpose  Trips
##      <qtr> <chr>    <chr>           <chr>    <dbl>
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
## 10 2000 Q2 Adelaide South Australia Business  169.
## # ... with 24,310 more rows
```

tsibble objects

```
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State           Purpose  Trips
##      Index <chr>    <chr>           <chr>    <dbl>
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
## 10 2000 Q2 Adelaide South Australia Business  169.
## # ... with 24,310 more rows
```

tsibble objects

```
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State           Purpose  Trips
##      Index    Keys                            <dbl>
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
## 10 2000 Q2 Adelaide South Australia Business  169.
## # ... with 24,310 more rows
```

`tsibble` objects

```
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State           Purpose  Trips
##      Index    Keys                            Measure
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
## 10 2000 Q2 Adelaide South Australia Business  169.
## # ... with 24,310 more rows
```

`tsibble` objects

```
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State           Purpose  Trips
##      Index    Keys                           Measure
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Busin
##  4 1998 Q4 Adelaide South Australia Busin
##  5 1999 Q1 Adelaide South Australia Busin
##  6 1999 Q2 Adelaide South Australia Busin
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
## 10 2000 Q2 Adelaide South Australia Business  169.
## # ... with 24,310 more rows
```

Domestic visitor nights in thousands by state/region and purpose.

Working with `tsibble` objects

We can use the `filter()` function to select rows.

```
tourism %>%
  filter(Purpose == "Business")
```

```
## # A tsibble: 6,080 x 5 [1Q]
## # Key:       Region, State, Purpose [76]
##    Quarter Region   State           Purpose  Trips
##      <qtr> <chr>    <chr>           <chr>    <dbl>
##  1 1998 Q1 Adelaide South Australia Business  135.
##  2 1998 Q2 Adelaide South Australia Business  110.
##  3 1998 Q3 Adelaide South Australia Business  166.
##  4 1998 Q4 Adelaide South Australia Business  127.
##  5 1999 Q1 Adelaide South Australia Business  137.
##  6 1999 Q2 Adelaide South Australia Business  200.
##  7 1999 Q3 Adelaide South Australia Business  169.
##  8 1999 Q4 Adelaide South Australia Business  134.
##  9 2000 Q1 Adelaide South Australia Business  154.
## 10 2000 Q2 Adelaide South Australia Business  169.
## # … with 6,070 more rows
```

Working with `tsibble` objects

We can use the `select()` function to select columns.

```
tourism %>%
  filter(Purpose == "Business") %>%
  select(Region,Trips)
```

```
## # A tsibble: 6,080 x 5 [1Q]
## # Key:       Region, State, Purpose [76]
##    Region   Trips Quarter State           Purpose
##    <chr>    <dbl>   <qtr> <chr>           <chr>
##  1 Adelaide  135. 1998 Q1 South Australia Business
##  2 Adelaide  110. 1998 Q2 South Australia Business
##  3 Adelaide  166. 1998 Q3 South Australia Business
##  4 Adelaide  127. 1998 Q4 South Australia Business
##  5 Adelaide  137. 1999 Q1 South Australia Business
##  6 Adelaide  200. 1999 Q2 South Australia Business
##  7 Adelaide  169. 1999 Q3 South Australia Business
##  8 Adelaide  134. 1999 Q4 South Australia Business
##  9 Adelaide  154. 2000 Q1 South Australia Business
```

Working with `tsibble` objects

- We can use `group_by()` function to group over keys.
  - We can also do it with: `group_by_key()`
- We can use the `summarise()` function to summarise over keys.

```
tourism %>%
  group_by(Region, Purpose) %>%
  summarise(Trips = mean(Trips)) %>%
  ungroup()
```

```
## # A tsibble: 24,320 x 4 [1Q]
## # Key:       Region, Purpose [304]
##    Region   Purpose  Quarter Trips
##    <chr>    <chr>      <qtr> <dbl>
##  1 Adelaide Business 1998 Q1  135.
##  2 Adelaide Business 1998 Q2  110.
##  3 Adelaide Business 1998 Q3  166.
##  4 Adelaide Business 1998 Q4  127.
```

Working with `tsibble` objects

- We can use `index_by()` function to group over index
- We can use the `summarise()` function to summarise over index.

```
tourism %>%
index_by(Quarter) %>%
  summarise(total_trips = sum(Trips))
```

```
## # A tsibble: 80 x 2 [1Q]
##    Quarter total_trips
##      <qtr>       <dbl>
##  1 1998 Q1      23182.
##  2 1998 Q2      20323.
##  3 1998 Q3      19827.
##  4 1998 Q4      20830.
##  5 1999 Q1      22087.
##  6 1999 Q2      21458.
##  7 1999 Q3      19914.
##  8 1999 Q4      20028.
```

Working with `tsibble` objects

We can use the `mutate()` function to create new variables.

```
tourism %>%
  mutate(year = year(Quarter)) -> m1
```

```
## # A tsibble: 24,320 x 6 [1Q]
## # Key:       Region, State, Purpose [304]
##    Quarter Region   State      Purpose Trips  year
##      <qtr> <chr>    <chr>      <chr>   <dbl> <dbl>
##  1 1998 Q1 Adelaide South Aus~ Busine~  135.  1998
##  2 1998 Q2 Adelaide South Aus~ Busine~  110.  1998
##  3 1998 Q3 Adelaide South Aus~ Busine~  166.  1998
##  4 1998 Q4 Adelaide South Aus~ Busine~  127.  1998
##  5 1999 Q1 Adelaide South Aus~ Busine~  137.  1999
##  6 1999 Q2 Adelaide South Aus~ Busine~  200.  1999
##  7 1999 Q3 Adelaide South Aus~ Busine~  169.  1999
##  8 1999 Q4 Adelaide South Aus~ Busine~  134.  1999
##  9 2000 Q1 Adelaide South Aus~ Busine~  154.  2000
## 10 2000 Q2 Adelaide South Aus~ Busine~  169.  2000
## # ... with 24,310 more rows
```

Outline

Lab Session 1

1. Read [`ae_uk.csv`] into R
2. Check duplication
3. Create a tsibble object! Is the `index` a regular interval? If the answer is no, Which argument do you need to specify?
4. Create a new tsibble which has a regular interval of `30 minutes`, and has total admissions per hour for each day and type of gender and injury_type.
5. Is there any gap in data? you can use `has_gaps()`, `count_gaps()` and `scap_gaps()`
6. Create total hourly, daily, weekly, monthly and quarterly admissions (ignoring keys)