

GOOGLE DIRECTION API USER GUIDE

Reference: <https://developers.google.com/maps/documentation/directions/get-directions#TextValueObject>

Input	Option
Origin, Destination	Coordinates (latitude/longitude) OR string type address
Date, time, time zone	
Mode	driving, walking, bicycling, transit
Transit mode	Only when mode is transit <ul style="list-style-type: none"> ▪ Bus ▪ Subway ▪ Train ▪ Tram ▪ Rail

Output is different by mode you selected:

Driving	Walking	Bicycling	Transit
Duration (minute)	Duration(minute)	Duration(minute)	Total distance(mile)
Duration with traffic (minute)	Distance(mile)	Distance(mile)	Total travel time (minute)
Distance (mile)			Total walk distance(mile)
			Total walk duration(minute)
			Total transit distance(mile)
			Total transit duration(minute)
			Distance origin--transit(mile)
			Duration origin--transit(minute)
			Distance transit—destination(mile)
			Duration transit—destination(minute)
			# Of transfer
			Routes used
			Agency
			Transit mode
			Wait time (minute)
			Transport time (minute)

Step 1 – input.xlsx

Insert data in input.xlsx file

If input data is **string type**, insert the data in Origin_address and Dest_address columns

LabelID	Origin_address	Dest_address	Origin_lon	Origin_lat	Dest_lon	Dest_lat
1	1222 22nd NW, DC	621 Morrison St, Portland, OR				
2	1223 22nd NW, DC	622 Morrison St, Portland, OR				
3	1224 22nd NW, DC	623 Morrison St, Portland, OR				
4	1225 22nd NW, DC	624 Morrison St, Portland, OR				
5	1226 22nd NW, DC	625 Morrison St, Portland, OR				

If input data is **coordinates**, insert data in Origin_lon, Origin_lat, Dest_lon and Dest_lat columns

LabelID	Origin_address	Dest_address	Origin_lon	Origin_lat	Dest_lon	Dest_lat
1			-122.476689	37.756221	-122.384182	37.662305
3			-122.425277	37.801098	-122.384182	37.662305
8			-122.423173	37.777596	-122.384182	37.662305
9			-122.424686	37.784876	-122.384182	37.662305
10			-122.4752	37.72587	-122.384182	37.662305

Note: If you have both types inserted, **string type** will be process.

Step 2 – config.R

Define the config in config.R

- **gKey:** valid google API key
- **departure_datetime:** departure date and local time (**Note: ONLY future date is working**)
- **timezone:** default time zone is EDT. `as.POSIXct` function sometimes fail to recognize time zone. You can run `OlsonNames()` line and look up the list of time zone. Also make sure if your time zone is recognized correctly by running `print(as.POSIXct(departure_datetime, tz = timezone))`

```

1
2 # Change API-----
3 gkey = ""
4
5 # Define departure date and local time-----
6 departure_datetime <- "2022-05-20 07:30:00"
7 timezone = 'America/Los_Angeles'
8
9 ## look up the list of time zone
10 olsonNames() → run this line to lookup time zone list
11 ## make sure if your time zone is recognized correctly
12 print(as.POSIXct(departure_datetime, tz = timezone))
13
14
15 # Define mode-----
16 mode_sel <- "transit"
17
18 ## define transit mode if your selected mode is "transit" above
19 transit_mode <- 'bus'
20 ## mode options:
21 ## - driving
22 ## - walking
23 ## - bicycling
24 ## - transit
25
26

```

- **mode_sel:** mode to measure trip duration and time (options: driving, walking, bicycling, transit)

Step 3 – main.R

Run **main.R** script and find the output in **output** folder.