

1. 비행 달이 7, 8, 9월인 행만 추려내시오.

```
> flights %>% filter(month == c(7,8,9))
# A tibble: 28,774 x 19
   year month   day dep_time sched_dep_time dep_delay arr_time
  <int> <int> <int>   <int>         <int>         <dbl>   <int>
1  2013     7     1     29           2245           104     151
2  2013     7     1     46           2051           235     304
3  2013     7     1    100           2146           194     327
4  2013     7     1    111           2359            72     448
5  2013     7     1    538            540           -2     800
6  2013     7     1    547            548           -1     903
7  2013     7     1    551            600           -9     721
8  2013     7     1    557            600           -3     817
9  2013     7     1    557            600           -3     712
10 2013     7     1    559            605           -6     748
# ... with 28,764 more rows, and 12 more variables:
#   sched_arr_time <int>, arr_delay <dbl>, carrier <chr>,
#   flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
#   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
#   time_hour <dtm>
```

2. 목적지(dest)가 "IAH"이거나 "HOU"인 행만 추려내시오.

```
> flights %>% filter(dest == c("IAH", "HOU"))
# A tibble: 4,655 x 19
  year month   day dep_time sched_dep_time dep_delay arr_time
  <int> <int> <int>   <int>         <int>         <dbl>   <int>
1  2013     1     1     517             515           2     830
2  2013     1     1     623             627          -4     933
3  2013     1     1    1028            1026           2    1350
4  2013     1     1    1114             900        134    1447
5  2013     1     1    1208            1158          10    1540
6  2013     1     1    1306            1300           6    1622
7  2013     1     1    1527            1515          12    1854
8  2013     1     1    1620            1620           0    1945
9  2013     1     1    1725            1720           5    2045
10 2013     1     1    1855            1848           7    2203
# ... with 4,645 more rows, and 12 more variables:
#   sched_arr_time <int>, arr_delay <dbl>, carrier <chr>,
#   flight <int>, tailnum <chr>, origin <chr>, dest <chr>,
#   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
#   time_hour <dtm>
```

3. 도착지연 시간(arr_delay)이 60분이고, 출발지연 시간(dep_delay)이 0분인 행만 추려내시오.

```
> flights %>% filter(arr_delay == 60 & dep_delay == 0)
# A tibble: 6 x 19
   year month   day dep_time sched_dep_time dep_delay arr_time
  <int> <int> <int>   <int>         <int>         <dbl>   <int>
1  2013     4    10    1802             1802           0     2238
2  2013     4    12     950             950           0     1352
3  2013     5    23    1130             1130           0     1427
4  2013     6    14    1559             1559           0     2014
5  2013     7    22    1245             1245           0     1635
6  2013     8    29    1720             1720           0     2020
```

4. year, month, day 열만 추려내시오.

```
> flights %>% select(year, month, day)
```

```
# A tibble: 336,776 x 3
```

	year	month	day
	<int>	<int>	<int>
1	2013	1	1
2	2013	1	1
3	2013	1	1
4	2013	1	1
5	2013	1	1
6	2013	1	1
7	2013	1	1
8	2013	1	1
9	2013	1	1
10	2013	1	1

```
# ... with 336,766 more rows
```

5. dep_time부터 arr_delay 열까지 한꺼번에 추려내시오.

```
> flights %>% select(dep_time:arr_delay)
# A tibble: 336,776 x 6
  dep_time sched_dep_time dep_delay arr_time sched_arr_time
  <int>         <int>         <dbl>    <int>         <int>
1     517           515           2      830           819
2     533           529           4      850           830
3     542           540           2      923           850
4     544           545          -1     1004          1022
5     554           600          -6      812           837
6     554           558          -4      740           728
7     555           600          -5      913           854
8     557           600          -3      709           723
9     557           600          -3      838           846
10    558           600          -2      753           745
# ... with 336,766 more rows, and 1 more variable:
#   arr_delay <dbl>
```

6. year, month, day에 따른 dep_delay의 평균을 구하시오. (결측치도 처리할 것)

```
* year > flights %>% filter(!is.na(dep_delay)) %>%  
+ group_by(year) %>%  
+ summarise(delay_year = mean(dep_delay)) %>%  
+ ungroup()  
# A tibble: 1 x 2  
  year delay_year  
  <int>     <dbl>  
1  2013      12.6
```

* day

```
> flights %>% filter(!is.na(dep_delay)) %>%  
+ group_by(day) %>%  
+ summarise(delay_day = mean(dep_delay)) %>%  
+ ungroup()  
# A tibble: 31 x 2  
  day delay_day  
  <int>     <dbl>  
1     1      14.2  
2     2      14.1  
3     3      10.8  
4     4       5.79  
5     5       7.82  
6     6       6.99  
7     7      14.3  
8     8      21.8  
9     9      14.6  
10    10      18.3  
# ... with 21 more rows
```

* month

```
> flights %>% filter(!is.na(dep_delay)) %>%  
+ group_by(month) %>%  
+ summarise(delay_month = mean(dep_delay)) %>%  
+ ungroup()  
# A tibble: 12 x 2  
  month delay_month  
  <int>     <dbl>  
1     1      10.0  
2     2      10.8  
3     3      13.2  
4     4      13.9  
5     5      13.0  
6     6      20.8  
7     7      21.7  
8     8      12.6  
9     9       6.72  
10    10       6.24  
11    11       5.44  
12    12      16.6
```

7. 목적지(dest)에 따른 dep_delay의 평균을 구해 내림차순으로 정리하시오.

```
> flights %>% filter(!is.na(dep_delay)) %>%  
+   group_by(dest) %>%  
+   summarise(delay_dest = mean(dep_delay)) %>%  
+   arrange(desc(delay_dest))  
# A tibble: 104 x 2  
  dest    delay_dest  
  <chr>      <dbl>  
1 CAE         35.6  
2 TUL         34.9  
3 OKC         30.6  
4 BHM         29.7  
5 TYS         28.5  
6 JAC         26.5  
7 DSM         26.2  
8 RIC         23.6  
9 ALB         23.6  
10 MSN        23.6  
# ... with 94 more rows
```