

# 실시간 내 주변 운영 약국

## (서울시 한정)

**팀:** 이게왜되조

**팀원:** 2129002 강효은, 2129005 곽지수, 2129027 이수미,  
2129036 차수빈, 2116024 최은빈

## ◎ 개요

### 1. 프로젝트 소개 및 목표

- 기획의도
- 프로젝트 소개
- 프로젝트 방향성

### 2. 데이터 전처리

- 행정구 추출
- 컬럼명 통일
- 일요일/공휴일/야간운영 여부 처리

### 3. 알고리즘

- 필요한 데이터 준비하기
- 입력
- 출력

### 4. 프로그램 사용 방법

- 옵션1: 현재 운영중인 약국 찾기
- 옵션2: 요일별 야간 운영 정보
- 옵션3: 근처(행정구)에 있는 약국 전체 정보 조회

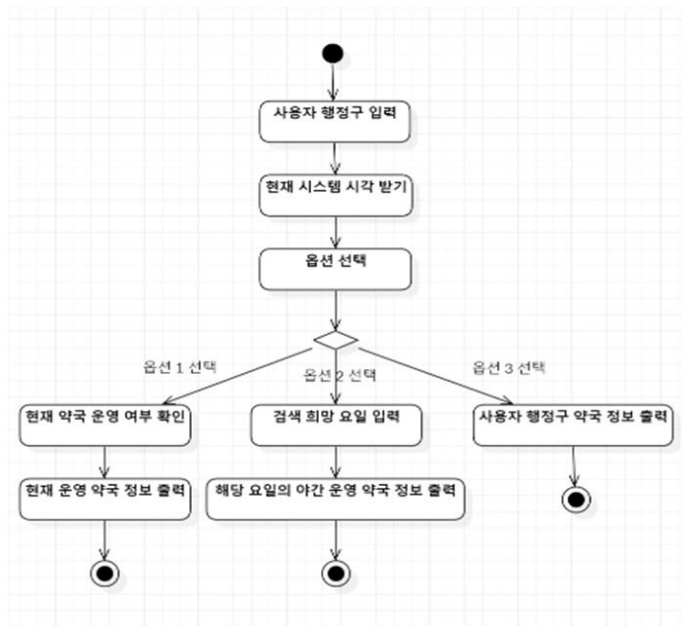
### 5. 기대효과

## 1. 프로젝트 소개 및 목표

- 기획의도

본 프로젝트는 사용자로 하여금 '서울시 약국 운영 정보'를 편리하게 알아볼 수 있도록 도움을 제공하는 것을 목표로 한다. 특히 야간이나 공휴일과 같이 운영 중인 약국을 찾기 어려운 시간대에 편리하게 약국을 찾을 수 있는 기능을 제공하고자 한다.

- 프로젝트 소개



[간략화한 프로그램 순서도 - 각 옵션 별 기능 포함]

본 프로그램은 사용자의 지역을 행정구 단위로 입력받고, 총 3 가지 옵션 중 선택에 따라 필요한 정보를 제공받을 수 있도록 한다.

1 번 옵션 선택 시, 현재 사용자의 접속 시점에 운영하는 약국 정보를 검색할 수 있다.

2 번 옵션 선택 시, 요일 별 야간 운영 약국 정보를 검색할 수 있다.

3 번 옵션 선택 시, 사용자가 입력한 행정구 내 모든 약국 정보를 얻을 수 있다.

- 프로젝트 방향성

해당 프로젝트는 데이터를 수집하고, 수집된 데이터를 전처리하고, 전처리된 데이터를 활용하여 필요한 작업을 수행하도록 알고리즘을 구현하는 순서로 진행되었다. 데이터 수집을 통해 서울시 열린 데이터 광장에서 제공하는 '서울시 약국 운영시간정보' 데이터셋을 수집했으며, 해당 데이터를 원본 데이터로 활용하였다.

(데이터 출처 : [https://www.e-gen.or.kr/egen/search\\_pharmacy.do?searchType=general](https://www.e-gen.or.kr/egen/search_pharmacy.do?searchType=general) )

이 데이터셋을 바탕으로 필요한 정보들만을 얻기 위해 전처리 작업을 완료하고, 사용자가 선택한 옵션에 따라 적절한 작업이 실행될 수 있도록 알고리즘을 구현하였다.

## 2. 데이터 전처리

### 2-1) 행정구 추출 작업

알고리즘 파트에서 사용자는 검색하고자 하는 지역을 행정구 단위로만 입력한다. 사용자가 입력한 행정구와 약국 주소를 대조하기 위해, 약국의 도로명 주소 정보에서 행정구만을 추출하여 별도의 컬럼으로 저장한다.

```
addr_li<- strsplit(df1$주소,split = " ")
head(addr_li,n = 3)

## [[1]]
## [1] "서울특별시" "강남구" "강남대로" "292" "3 층"
## [6] "(도곡동" "행행빌딩)"
##
## [[2]]
## [1] "서울특별시" "동대문구" "전농로" "60-1" "1 층"
## [6] "(답십리동)"
##
## [[3]]
## [1] "서울특별시" "강남구" "봉은사로 114 길" "42"
## [5] "(삼성동)"
```

데이터셋에 저장된 주소 정보 탐색 및 슬라이싱 결과 각 벡터의 두 번째 내부 원소가 추출해야 할 행정구 정보임을 알 수 있었다. 리스트의 경우 구성 벡터의 내부 원소에 접근하려면 [[ ]]을 이용하여야 한다. 따라서, 각 벡터의 두 번째 요소에 접근하는 함수 search()를 정의하고, 각 행마다 적용하였다.

```
search <- function(x){ # 두 번째 요소 찾기
  x[2]
}

# 각 행(각 벡터)마다 적용
df1$행정구 <- sapply(addr_li,search)
View(df1)
```

## 2-2) 컬럼 명 통일

수집한 RAW DATA 에는 프로그램을 구현하는데 불필요한 정보들 역시 담겨있다. 따라서 불필요한 컬럼들을 삭제해주었다.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	
약국ID	주소	약국명	대표전화	진료시간1	진료시간2	진료시간3	진료시간4	진료시간5	진료시간6	진료시간7	진료시간8	진료시간9	진료시간10	진료시간11	진료시간12	진료시간13	진료시간14	진료시간15	진료시간16	진료시간17	평판점수1	평판점수2	행위점수	작업일시	
C1108810	서울특별시가람약국	02-2225-0	2000	2000	2000	2000	2000	2000	1530	2000	1600	800	800	800	800	800	800	900	1030	1000	85	94	126.8867	37.46675	41.58.0
C1106934	서울특별시가람약국	02-518-42	2000	2000	2000	2000	2000	2000	1800	2000	1600	930	930	930	930	930	930	930	1030	1000	60	28	127.0206	37.52292	41.58.0
C1108554	서울특별시가람약국	02-3281-8	1730	1730	1730	1730	1730	1730	1300	2000	1600	900	900	900	900	900	900	900	1030	1000	85	12	126.8864	37.48108	41.58.0
C1104205	서울특별시가람약국	02-838-11	2200	2200	2200	2200	2200	2200	1900	1900	830	830	830	830	830	830	830	830	900	900	85	29	126.8931	37.47608	41.58.0
C1104582	서울특별시가람약국	02-997-45	1800	1800	1800	1800	1800	1800	1300	1900	1900	800	800	800	800	800	800	800	900	900	14	5	127.0457	37.65195	41.58.0
C1108835	서울특별시가람약국	02-3663-8	1930	1930	1930	2000	1930	1600	1900	1900	900	900	900	900	900	900	900	900	900	900	75	32	126.8603	37.56129	41.58.0
C1102136	서울특별시가람약국	02-3661-3	1830	1830	1830	1830	1830	1600	1900	1900	815	815	815	815	815	815	815	815	900	900	75	27	126.852	37.5636	41.58.0
C1103478	서울특별시가람약국	02-2659-2	2030	2030	2030	2030	2030	1730	1900	1900	900	900	900	900	900	900	900	900	900	900	75	75	126.8517	37.56259	41.58.0

[수집한 RAW DATA 의 일부]

약국명	주소	대표전화	시작(월)	마감(월)	~	시작(일)	마감(일)	시작(공휴일)	마감(공휴일)	일운영	공휴일운영	야간운영(월)	~	야간운영(일)	야간운영(공휴일)
이화약국	(도로명주소)	010-1111-1111	800	1900		NA	NA	NA	NA	FALSE	FALSE	TRUE		FALSE	FALSE
character	character	character	numeric	numeric		logical	logical	logical	logical	logical	logical	logical		logical	logical
기존에 있는 컬럼(컬럼명만 변경)										새로 추가할 컬럼					

이후 데이터 프레임을 위와 같이 사용하기 편리한 형태로 컬럼명을 통일시켜주었고, 별도로 필요한 컬럼들은 추가하는 작업을 진행했다.

```
# 불필요한 컬럼 삭제
df2 <- df1[, -c(1, 21:25)]
View(df2)

# 보기 좋은 형태로 컬럼 순서 변경하기
df3 <- df2[, c(2, 1, 3, 20, 12, 4, 13, 5, 14, 6, 15, 7, 16, 8, 17, 9, 18, 10, 19, 11)]
View(df3)

# 컬럼명 변경하기
new_colname <- c("약국명", "주소", "대표전화", "행정구", "시작(월)", "마감(월)", "시작(화)", "마감(화)", "시작(수)", "마감(수)", "시작(목)", "마감(목)", "시작(금)", "마감(금)", "시작(토)", "마감(토)", "시작(일)", "마감(일)", "시작(공휴일)", "마감(공휴일)")
colnames(df3) <- new_colname
View(df3)
```

[컬럼명 통일 작업을 위해 작성한 코드]

## 2-3) 일요일/공휴일/야간운영 여부 처리

### A> 일요일

#### 1. 일요일 운영 여부

```
df3$일운영 <- (!is.na(df3$`시작(일)`)) & (!is.na(df3$`마감(일)`))
View(df3)
```

일요일의 경우 약국 운영을 하지 않는 경우 시작과 마감의 운영정보 데이터가 NA로 들어갈 수 있다. 이러한 경우, 추가한 일요일 운영정보 컬럼에 FALSE가 저장되도록 하였다.

이후 공휴일 운영 여부 컬럼도 일요일 운영여부 컬럼과 동일한 방식을 적용하여 논리 연산을 수행하고, 논리 연산의 결과를 '공휴일운영' 컬럼에 새로이 추가해 주었다.

```
df3$공휴일운영 <- (!is.na(df3$`시작(공휴일)`)) & (!is.na(df3$`마감(공휴일)`))
View(df3)
```

다음으로는 야간운행을 하는 약국들에 대한 처리를 진행하였다. 원본 데이터를 보면 마감 시간에서 새벽 시간대를 표시하는 형식이 통일되어 있지 않음을 확인할 수 있었다.

	약국명	주소	대표전화	행정구	시작(월)	마감(월)
368	경희정문약국	서울특별시 동대문구 경희대로 22-1 1층 (회기동)	02-957-6257	동대문구	0	0
484	국민약국	서울특별시 동대문구 이문로 37 104호 (회기동 회기역 한일...	02-962-5008	동대문구	0	0
849	단우한약국	서울특별시 동대문구 약령중앙로 5 대산빌딩 1층 (제기동)	02-000-0000	동대문구	0	0

	약국명	주소	대표전화	행정구	시작(월)	마감(월)
4033	영국대우약국	서울특별시 영등포구 영등포로137길 34 1층 (영등포)	02-487-8880	영등포구	1000	2300
2	365종로약국	서울특별시 동대문구 전농로 60-1 1층 (답십리동)	02-2215-8704	동대문구	800	2400
6	건강한세상행복한약국	서울특별시 강남구 강남대로 488 (논현동) 1층	02-545-0029	강남구	0	2400

이후 야간운영 여부를 파악할 때 2030 이후에 영업하는 약국을 비교 연산을 통해 추출할 예정이기에, 새벽 시간대가 2030 이상이어야 제대로 추출될 것이기에, 새벽 시간대를 모두 24시 이후 시간대로 표시하는 형식을 택하였다. 또한, 원본 데이터에서 가장 빠른 시작 시간이 5시 30분이기에, 모든 약국의 운영시간이 530에서 2929 사이에 있도록 시간 표기 형식을 통일시키기로 결정하였다.

모든 약국들의 운영 시간대를 통일시키기 위해 먼저 24시 운영 약국과 그렇지 않은 약국을 분리하였다. 현재 약국의 시작시간과 마감시간은 모두 정수형, 즉 numeric으로 저장되어 있기에, 마감시간에서 시작시간을 빼주면 약국의 총 운영 시간을 구할 수 있다. 이때, 총 운영시간이 000또는 2400으로 나오는 약국들이 24시 운영을 하는 약국들이다.

# 함수 선언

```
open_24 <- function(df, day){
  open_filter = paste0('시작(', day, ')') # 해당 요일의 시작 시간 가져오기
  close_filter = paste0('마감(', day, ')') # 해당 요일의 마감 시간 가져오기
  opentime <- df[,open_filter]
  closetime <- df[,close_filter]
  df[,open_filter] <- ifelse(((closetime - opentime == 0)|((closetime - opentime == 2400))), 530, opentime)
  df[,close_filter] <- ifelse(((closetime - opentime == 0)|((closetime - opentime == 2400))), 2929, closetime)
  return(df)
}
```

함수에서는 약국 운영 정보가 저장된 데이터 프레임과 검사를 수행할 요일을 매개변수로 받아온다. 이후, 각 행마다 검사를 하여 24시 약국인 경우 시작시간을 530, 마감시간을 2929로 바꾸어

주었다. 이후 해당 함수를 요일별로 호출하여 해당 작업을 수행하였다.

```
# 함수 호출
df3 <- open_24(df3, "월")
df3 <- open_24(df3, "화")
df3 <- open_24(df3, "수")
df3 <- open_24(df3, "목")
df3 <- open_24(df3, "금")
df3 <- open_24(df3, "토")
df3 <- open_24(df3, "일")
df3 <- open_24(df3, "공휴일")
```

새벽 운영을 하는 약국들 또한 유사한 방식을 적용하였다. 마감시간의 새벽 시간 표기 형식을 통일시켜주기 위해 마감시간이 000에서 529 사이인 경우(새벽 시간대에 마감을 하는 경우) 마감 시간에 2400을 더해주었다.

```
# 함수 선언
midnight <- function(df, day){
  close_filter = paste0('마감(', day, ')')
  closetime <- df[, close_filter]
  df[, close_filter] <- ifelse(((closetime >= 0) & (closetime < 530)), closetime
+ 2400, closetime)
  return(df)
}

# 함수 호출
df3 <- midnight(df3, "월")
df3 <- midnight(df3, "화")
df3 <- midnight(df3, "수")
df3 <- midnight(df3, "목")
df3 <- midnight(df3, "금")
df3 <- midnight(df3, "토")
df3 <- midnight(df3, "일")
df3 <- midnight(df3, "공휴일")
```

새벽 시간대를 표시하는 형식을 통일시킨 후, 별도로 야간 운영 여부 컬럼을 만들어 해당 컬럼에 추가해준다. 위의 24시 운영 약국을 찾아내는 방법이나 새벽 운영을 하는 약국들을 처리해준 방식과 유사한 방법을 적용하여, 약국 마감시간이 2030 이상 2929 이하인 약국들을 골라내어 야간 운영 여부에 TRUE가 저장되도록 하였다. 이후, 요일별로 함수를 반복적으로 호출하여 해당 컬럼들을 처리해주었다.

```
# 함수 선언
open_midnight <- function(df,day){
  close_filter = paste0('마감(',day,')')
  closetime <- df[,close_filter]
  midnight_filter = paste0('야간운영(',day,')')
  df[,midnight_filter] <- ifelse(((closetime >= 2030)&(closetime <= 2929)),TRUE,FALSE)
  return(df)
}

# 함수 호출
df3 <- open_midnight(df3,"월")
df3 <- open_midnight(df3,"화")
df3 <- open_midnight(df3,"수")
df3 <- open_midnight(df3,"목")
df3 <- open_midnight(df3,"금")
df3 <- open_midnight(df3,"토")
df3 <- open_midnight(df3,"일")
df3 <- open_midnight(df3,"공휴일")
```

하지만, 일요일과 공휴일의 경우 운영을 하지 않는 약국들은 마감시간이 NA로 되어 있기에, 단순히 비교 연산자를 사용하여 시간대를 비교하면 연산 결과 NA가 나온다. 따라서, ifelse문을 통과하더라도 결과값이 FALSE가 아닌 NA로 저장된다. 야간운영여부 컬럼에 NA가 포함되어 있는 경우 이후 알고리즘 파트에서 필요한 약국 정보가 제대로 호출되지 않을 가능성이 존재한다. 따라서, 해당 약국들만을 다시 추출하여 기존에 NA로 저장된 야간운영 여부를 FALSE로 변경하였다.

```
df3[which(is.na(df3$`시작(일)`)), "야간운영(일)"] <- FALSE
df3[which(is.na(df3$`시작(공휴일)`)), "야간운영(공휴일)"] <- FALSE
```

이렇게 전처리가 완료된 데이터는 csv 파일로 저장하고, 알고리즘 파트에서 해당 데이터를 불러와 이후의 작업을 수행하게 된다.



### 3. 알고리즘

#### 3-1) 필요한 데이터 준비하기

우선 전처리가 완료된 약국 운영 정보를 불러온다. 이후, 한글로 칼럼명이 설정되었을 경우 글꼴이 깨져 데이터를 불러올 때 오류가 생길 것을 대비해 칼럼명을 재설정 하였다.

```
df1 <- read.csv("seoul_pharmacy_final.csv")

# 칼럼명 재설정
new_colname <- c("약국명", "주소", "대표전화", "행정구", "시작(월)", "마감(월)", "시작(화)", "마감(화)", "시작(수)", "마감(수)", "시작(목)", "마감(목)", "시작(금)", "마감(금)", "시작(토)", "마감(토)", "시작(일)", "마감(일)", "시작(공휴일)", "마감(공휴일)", "일운영", "공휴일운영", "야간운영(월)", "야간운영(화)", "야간운영(수)", "야간운영(목)", "야간운영(금)", "야간운영(토)", "야간운영(일)", "야간운영(공휴일)")
colnames(df1) <- new_colname
```

사용자가 입력한 날짜가 공휴일에 해당되는지 판별하기 위해, 2022년의 공휴일 정보를 담은 csv 파일을 다음과 같이 따로 제작하였다. 이후, 해당 데이터를 불러와 '날짜' 칼럼만을 선택하여 holiday\_2022 벡터로 저장하였다.

	A	B	C
1	날짜	설명	
2	2022-01-01	신정	
3	2022-01-31	설날	
4	2022-02-01	설날	
5	2022-02-02	설날	
6	2022-03-01	삼일절	
7	2022-03-09	대통령 선거일	
8	2022-05-05	어린이날	
9	2022-05-08	부처님오신날	
10	2022-06-01	전국동시지방선거	
11	2022-06-06	현충일	
12	2022-08-15	광복절	
13	2022-09-09	추석	
14	2022-09-10	추석	

```
# 파일 읽어오기
holiday_df <- read.csv("holiday_2022.csv")

# 2022 년의 공휴일 날짜 정보를 벡터로 저장
holiday_2022 <- holiday_df$날짜

View(holiday_df) # 개발 시 데이터 확인을 위한 코드
```

#### 3-2) 입력

해당 프로그램은 입력 파트에서 readline을 통해 사용자에게 행정구를 직접 입력 받는다.

```
print("서울시 약국 운영 정보를 알려드립니다!")

## [1] "서울시 약국 운영 정보를 알려드립니다!"

place <- readline(prompt = "검색하고자 하는 지역을 행정구 단위로 입력해 주세요 : ")

## 검색하고자 하는 지역을 행정구 단위로 입력해 주세요 :

cat("입력하신 지역은 [", place, "] 입니다.")

## 입력하신 지역은 [ ] 입니다.
```

place	"강남구"
-------	-------

현재 시간 및 요일은 R의 함수를 사용해 자동으로 얻어온다.

```
curtime = Sys.time() # "2022-05-29 02:38:40 KST"
curtime = as.character(curtime) # "2022-05-29 02:38:40"
curtime = substr(curtime,12,16) # "02:38"
curtime = gsub(":", "", curtime) # "0238"
curtime = as.numeric(curtime) # 238

# 530 <= (현재 시각) <= 2929 가 되도록 형식 맞춰주기
curtime <- ifelse((curtime<530),curtime + 2400,curtime)
```

현재 시간을 Sys.time()을 이용하여 얻어오면 불필요한 정보들도 함께 저장된다. 따라서, curtime을 문자로 형변환을 해주고, substr( )을 이용하여 시간과 분에 해당하는 곳만 가져온다. 이후 gsub( )를 이용하여 콜론(:)을 제거한 후 숫자형으로 변환한다. 이후 전처리된 데이터와 curtime의 시간 형식을 통일해주면 사용자의 현재 시각을 입력받는 과정이 마무리된다.

curtime	2404
---------	------

현재 요일은 Sys.Date를 통해 불러온다.

```
# 현재 날짜
curdate <- ifelse((curtime >= 2400) & (curtime <= 2929),
                 as.character(Sys.Date() - 1), as.character(Sys.Date())) # 새벽
# 새벽 시간대 요일 처리

# 현재 요일
curday <- ifelse((curdate%in% holiday_2022),"공휴일",weekdays(as.Date(curdate))) # 공휴일 처리
curday <- ifelse(curday != "공휴일",substr(curday,1,1),"공휴일") # 요일 저장 형식 처리
```

다만 앞서 새벽 시간 대신 00:00~05:29분을 24:00~29:29로 변환을 시켜주면 하루를 더한 셈이 되기에(화요일 새벽 2시의 경우 월요일 26시로 처리된다.), 이 시간대에 해당하는 부분들은 Sys.Date()로 얻은 날짜에서 하루를 빼 주어야 한다.

또한, 공휴일 정보가 저장된 holiday\_2022 벡터를 이용하여, 만약 해당 날짜가 holiday\_2022내의 공휴일에 해당된다면, 해당 요일이 공휴일로 지정될 수 있도록 하였다.

이렇게 필요한 입력값들을 처리한 후, 사용자에게 행정구를 입력 받을 때와 마찬가지로 readline 을 이용해 사용자가 원하는 옵션 번호를 선택하도록 한다.

```
cat("옵션에 따라 필요한 정보를 선택적으로 얻으실 수 있습니다.", "\n",
    "1) 현재 운영 중인 약국 정보", "\n",
    "2) 요일 별 야간 운영 약국 정보 (20:30 분 이후 운영 약국)", "\n",
    "3) 사용자 주변 약국 정보", "\n")

## 옵션에 따라 필요한 정보를 선택적으로 얻으실 수 있습니다.
## 1) 현재 운영 중인 약국 정보
## 2) 요일 별 야간 운영 약국 정보 (20:30 분 이후 운영 약국)
## 3) 사용자 주변 약국 정보

option = readline(prompt = "원하시는 옵션의 번호를 선택해주세요>> ")

## 원하시는 옵션의 번호를 선택해주세요>>

cat("선택하신 옵션은", option, "입니다.")

## 선택하신 옵션은 1입니다.
```

option	"1"
--------	-----

### 3-3) 출력

사용자가 옵션을 선택하고 나면 이제 그에 맞는 알맞은 정보를 제공하게 된다. 먼저 모든 옵션에서 공통적으로 필요한 정보들을 가져온다. 먼저 사용자가 입력한 행정구 내의 약국 정보를 가져와 새로운 데이터 프레임 df\_gu에 저장한다.

```
df_gu <- df1[c(which(df1$행정구 == place)),]
```

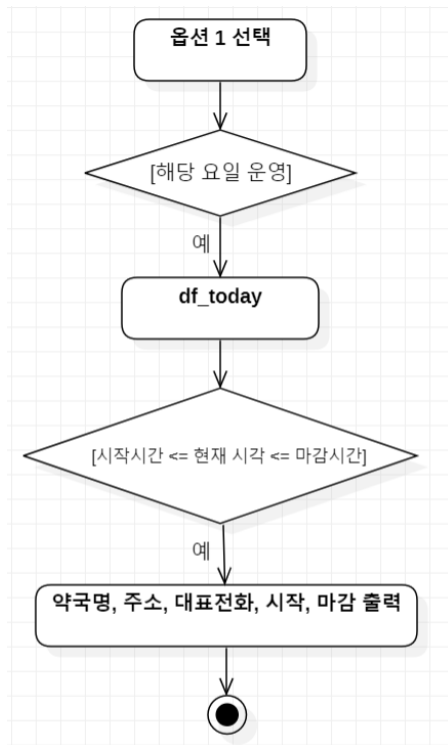
다음으로 약국의 운영 여부를 확인한다. 일요일이나 공휴일의 경우 원본 데이터에서 운영 시간에 NA가 포함되어 있기 때문에 이후 현재 운영 중인 약국 정보를 가져올 때 비교 연산이 제대로 이루어지지 않을 위험성이 존재한다. 따라서 사용자가 접속한 요일에 해당 약국의 운영 여부부터 확인해야 한다.

```
openToday <- function(df_gu, day){
  if((day == "일") | (day == "공휴일")){
    run_filter = paste0(day, "운영") # "일운영" 또는 "공휴일운영"
    df_run <- df_gu[c(which(df_gu[,run_filter])),] # 해당 요일에 운영하는 약국
  }
  else{ # 요일: 월 ~ 토
    df_run <- df_gu
  }
  return(df_run)
}
```

데이터 전처리 단계에서 추가해 둔 '일운영' 컬럼과 '공휴일운영' 컬럼을 이용하여 약국의 운영 여부를 확인하고, 해당 요일에 운영하는 약국만을 df\_run에 저장한다.

이제 선택한 옵션에 따른 필터링을 실행한다. 각각의 옵션에 대한 기능을 수행하는 함수를 만들어 둔 후, 최종 알고리즘에서 switch문을 활용하여 해당하는 함수를 호출하는 방식을 사용하였다.

첫번째 옵션은 '현재 운영 중인 약국 찾기'이다.

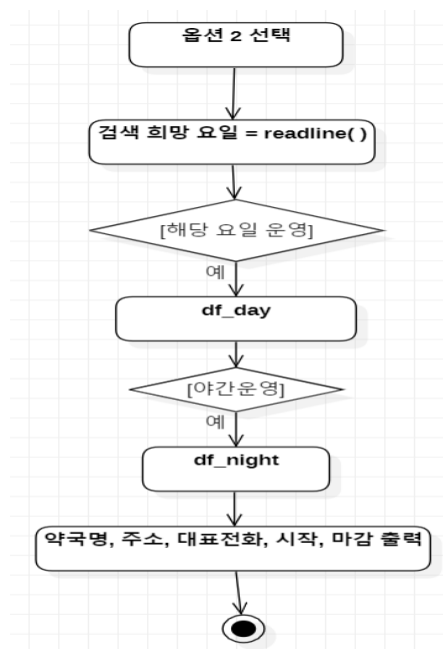


앞서 만들어 둔 openToday( ) 함수를 이용해 해당 요일의 약국 운영 여부를 확인하고, 운영 중인 약국 정보를 새로운 데이터 프레임 df\_today에 저장한다. 이후, 시작시간 <= 운영시간 <= 마감 시간인 약국들의 "약국명", "주소", "대표전화", 그리고 start(시작 시간)과 end(마감 시간) 정보를 각 약국별로 저장한 후, 해당 약국 정보들을 보여준다.

```

nowRunning <- function(df_gu, curday){
  df_today <- openToday(df_gu, curday) # 운영 여부 확인
  start = paste0("시작(", curday, ")") # 약국 시작
  end = paste0("마감(", curday, ")") # 약국 마감
  df_current <- df_today[c(which((df_today[,start] <= curtime) & (df_today[,end] >= curtime))), c("약국명", "주소", "대표전화", start, end)]
  View(df_current)
}
  
```

두번째 옵션은 '요일별 야간 운영 정보'이다.



사용자에게 검색하고자 하는 요일을 입력받은 후, 사용자가 입력한 요일을 다시 한번 알려줌으로써 사용자가 본인이 검색하고자 한 요일을 올바르게 입력했는지 재확인할 수 있도록 하였다.

```

openByDay <- function(df_gu){
  # 사용자에게 검색하고자 하는 요일을 입력받음
  cat("검색하고자 하는 요일을 입력해 주세요","\n",
      "(예: 월, 화, ... ,일, 공휴일)>> ")
  searchday <- readline()
  cat("입력하신 요일은",searchday,"입니다.")

  # 해당 요일에 여는 약국 찾기
  df_day <- openToday(df_gu,searchday)

  # 약국 운영시간 가져오기
  start = paste0("시작(",searchday,")") # 약국 시작
  end = paste0("마감(",searchday,")") # 약국 마감

  # 해당 요일에 여는 약국들 중 야간 운영을 하는 약국을 찾기
  night_filter = paste0("야간운영(",searchday,")")
  df_night <- df_day[c(which(df_day[,night_filter])),]
  df_openday <- df_night[,c("약국명","주소","대표전화",start,end)]

  View(df_openday)
}
  
```

앞서 만든 openToday() 함수를 이용해 사용자가 입력한 요일에 여는 약국을 찾고, 이를 새로운 데이터 프레임 df\_day에 저장한다. 약국의 운영시간을 가져와 야간운업을 하는 약국들만을 새로운 데이터 프레임 df\_night에 저장하고, 이중 "약국명", "주소", "대표전화", start(시작 시간), end(마감 시간)의 필요한 정보들만 뽑아와 df\_openday에 저장한다. 최종적으로 df\_openday를 출력함으로써 사용자가 검색하고자 하는 요일에 야간 운업을 하는 약국의 필요한 정보를 제공해준다.

세번째 옵션은 '근처에 있는 약국의 전체 정보 조회'이다. 앞에서 이미 사용자가 입력한 행정구(df\_gu)에 대한 처리가 되어있었기 때문에, 이때는 View를 이용해 처리된 data frame을 보여주기만 하면 된다.

알고리즘의 마지막 단계는 'switch문으로 옵션 묶기'이다. 옵션 선택에 따라 앞서 나온 옵션1, 2, 3 중 해당하는 함수를 실행하도록 switch문을 활용해 하나의 조건문으로 묶어낸다. 만약 사용자가 지정되어 있지 않은 옵션을 선택할 경우 "옵션을 잘못 입력하셨습니다."를 출력하도록 한다. 이로써 전체 알고리즘 구현을 완료하였다.

```
switch(option,
  "1" = {nowRunning(df_gu, curday)},
  "2" = {openByDay(df_gu)},
  "3" = {View(df_gu)},
  {print("옵션을 잘못 입력하셨습니다.")})

## [1] "옵션을 잘못 입력하셨습니다."
```

#### 4. 프로그램 사용방법

##### - 지역 입력

안내 문구를 확인한 후, 검색하고자 하는 지역을 행정구 단위로 입력한다. 입력 후 확인 문구를 확인해준다. (예: 강남구)



```
R 4.1.2 - C:/Users/smlee/Desktop/팀플/
> # view(df1) # 개발 시 데이터 확인을 위한 코드
> # 파일 읽어오기
> holiday_df <- read.csv("C:/Temp/holiday_2022.csv")
>
> # 2022년의 공휴일 날짜 정보를 벡터로 저장
> holiday_2022 <- holiday_df$날짜
>
> # view(holiday_df) # 개발 시 데이터 확인을 위한 코드
> print("서울시 약국 운영 정보를 알려드립니다!")
[1] "서울시 약국 운영 정보를 알려드립니다!"
> place <- readline(prompt = "검색하고자 하는 지역은 행정구 단위로 입력해 주세요 : ")
검색하고자 하는 지역을 행정구 단위로 입력해 주세요 : 강남구
> cat("입력하신 지역은 [", place, "] 입니다.")
입력하신 지역은 [ 강남구 ] 입니다.

> curtime = Sys.time() # 2022-05-29 02:38:40 KST
> curtime = as.character(curtime) # "2022-05-29 02:38:40"
```

① 지역 입력  
② 입력 확인

## - 옵션 선택

주어진 옵션 중 원하는 옵션의 번호를 입력한다.

```

> #31. 상용선 옵션 목록 읽기와 쓰기
옵션에 따라 필요한 정보를 선택적으로 얻으실 수 있습니다.
1) 현재 운영 중인 약국 정보
2) 요일 별 야간 운영 약국 정보 (20:30분 이후 운영 약국)
3) 사용자 주변 약국 정보
> option = readline(prompt = "원하시는 옵션의 번호를 선택해주세요>> ")
원하시는 옵션의 번호를 선택해주세요>>

```

a> 옵션 1

### 1) 옵션 선택 및 확인

1를 입력 후 제대로 입력되었나 확인한다.

```

옵션에 따라 필요한 정보를 선택적으로 얻으실 수 있습니다.
1) 현재 운영 중인 약국 정보
2) 요일 별 야간 운영 약국 정보 (20:30분 이후 운영 약국)
3) 사용자 주변 약국 정보
> option = readline(prompt = "원하시는 옵션의 번호를 선택해주세요>> ")
원하시는 옵션의 번호를 선택해주세요>> 1
> cat("선택하신 옵션은", option, "입니다. ")
선택하신 옵션은 1 입니다.
> df_qu <- df1[which(df1$행정구 == place),]

```

① 옵션 선택  
② 선택 확인

### 2) 결과 확인

(22-06-10 금요일 20:55 기준) 현재 강남구에서 운영중인 약국 목록을 보여준다.

약국명	주소	대표전화	시작 (금)	마감 (금)
6 건강한세상행복한약국	서울특별시 강남구 강남대로 488 (논현동) 1층	02-545-0029	530	2929
16 논현힐리아약국	서울특별시 강남구 선릉로 641	02-540-7582	900	2200
169 갈마약국	서울특별시 강남구 압구정로 336 (신사동)	02-547-8990	900	2100
179 1층엔약국	서울특별시 강남구 강남대로 624 ICT TOWER 1층 일부포 (신...	02-518-1080	830	2030
184 강남구청약국	서울특별시 강남구 학동로 지하 346 강남구청역 B1층 111호 ...	02-511-6252	800	2100
189 강남미소약국	서울특별시 강남구 논현로86길 22 1층 (역삼동)	02-555-5640	900	2100
190 강남미약국	서울특별시 강남구 강남대로 376 1층 1호 (역삼동)	02-501-2105	830	2100
192 강남보선약국	서울특별시 강남구 도곡로 516 (대치동)	02-568-2310	1000	2200
195 강남새천년은누리약국	서울특별시 강남구 도곡로 523 1층 (대치동)	02-552-3524	1000	2230
197 강남세계로약국	서울특별시 강남구 논현로 52 남현빌딩 1층 (개포동)	02-572-2720	900	2130
199 강남씨티약국	서울특별시 강남구 강남대로 416 1층 (역삼동 창림빌딩)	02-553-1608	830	2130
204 강남역스타약국	서울특별시 강남구 테헤란로4길 6 지하1층 b106호 (역삼동)	02-6326-8908	900	2100
205 강남은누리약국	서울특별시 강남구 선릉로 406 102호 (대치동)	02-566-6480	800	2100
207 강남힐약국	서울특별시 강남구 강남대로 432 점프빌라노 지하1층 101호...	02-6440-5656	900	2230
211 강남자자약국	서울특별시 강남구 강남대로94길 27 1층 105호 (역삼동)	02-6465-4152	830	2100
213 강남퍼스트약국	서울특별시 강남구 도산대로 104 (논현동 1st Avenue)	02-515-7292	1000	2100
214 강남프라자약국	서울특별시 강남구 학동로 323 (논현동 한미빌딩)	02-514-4222	900	2100
215 강남하브약국	서울특별시 강남구 학동로 342	02-3014-9272	830	2030
267 개포메디칼약국	서울특별시 강남구 개포로 303	02-574-5410	930	2030
271 갤러리아온누리약국	서울특별시 강남구 선릉로 643 107호	02-517-0129	900	2100
272 갤러리아약국	서울특별시 강남구 학동로20길 55 106호	02-6227-5578	900	2230
493 국풍당약국	서울특별시 강남구 선릉로 654 1층 (삼성동 예인빌딩)	02-540-4556	830	2100

Showing 1 to 22 of 179 entries, 5 total columns

## b> 옵션 2

### 1) 옵션 선택 및 확인

2를 입력 후 제대로 입력되었나 확인한다.

```
옵션에 따라 필요한 정보를 선택적으로 얻으실 수 있습니다.  
1) 현재 운영 중인 약국 정보  
2) 요일별 야간 운영 약국 정보 (20:30분 이후 운영 약국)  
3) 사용자 주변 약국 정보  
> option = readline(prompt = "원하시는 옵션의 번호를 선택해주세요>> ")  
원하시는 옵션의 번호를 선택해주세요>> 2  
> cat("선택하신 옵션은", option, "입니다. ")  
선택하신 옵션은 2 입니다.  
> df_gu <- df1[c(which(df1$행정구 == place),)]  
> # view(df_gu)
```

① 옵션 선택  
② 선택 확인

### 2) 요일 선택 및 확인

검색하고 싶은 요일을 입력한다. (예: 월)

```
> switch(option,  
+ "1" = {nowRunning(df_gu, curday)},  
+ "2" = {openByday(df_gu)},  
+ "3" = {view(df_gu)},  
+ {print("옵션을 잘못 입력하셨습니다.")})  
검색하고자 하는 요일을 입력해 주세요  
                                     (예: 월, 화, ... ,일, 공휴일)>> 월  
입력하신 요일은 월 입니다.  
> |
```

### 3) 결과

입력한 요일에 야간 운영(마감시간 >= 2030) 하는 약국의 목록이 출력된다.

약국명	주소	대표전화	시작 (월)	마감 (월)
6 건강한세상행복한약국	서울특별시 강남구 강남대로 488 (논현동) 1층	02-545-0029	530	2929
16 논현필리아약국	서울특별시 강남구 선릉로 641	02-540-7582	900	2200
169 감마약국	서울특별시 강남구 압구정로 336 (신사동)	02-547-8990	900	2100
179 1층엔약국	서울특별시 강남구 강남대로 624 ICT TOWER 1층 일부부 (신...	02-518-1080	830	2030
184 강남구청역약국	서울특별시 강남구 학동로 지하 346 강남구청역 B1층 111호 ...	02-511-6252	800	2100
189 강남미소약국	서울특별시 강남구 논현로86길 22 1층 (역삼동)	02-555-5640	900	2100
190 강남미약국	서울특별시 강남구 강남대로 376 1층 (역삼동)	02-501-2105	830	2100
192 강남보선약국	서울특별시 강남구 도곡로 516 (대치동)	02-568-2310	1000	2200
195 강남새천년논누리약국	서울특별시 강남구 도곡로 523 1층 (대치동)	02-552-3524	1000	2230
197 강남세계로약국	서울특별시 강남구 논현로 52 남현빌딩 1층 (개포동)	02-572-2720	900	2130
199 강남씨티약국	서울특별시 강남구 강남대로 416 1층 (역삼동 창림빌딩)	02-553-1608	830	2130
204 강남역스타약국	서울특별시 강남구 테헤란로4길 6 지하1층 b106호 (역삼동)	02-6326-8908	900	2100
205 강남은누리약국	서울특별시 강남구 선릉로 406 102호 (대치동)	02-566-6480	800	2100
207 강남힐약국	서울특별시 강남구 강남대로 432 점프빌라노 지하1층 101호...	02-6440-5656	900	2230
211 강남자자약국	서울특별시 강남구 강남대로94길 27 1층 105호 (역삼동)	02-6465-4152	830	2100
213 강남파스트약국	서울특별시 강남구 도산대로 104 (논현동 1st Avenue)	02-515-7292	1000	2100
214 강남프라자약국	서울특별시 강남구 학동로 323 (논현동 한미빌딩)	02-514-4222	900	2100
215 강남하브약국	서울특별시 강남구 학동로 342	02-3014-9272	830	2030
267 개포메디칼약국	서울특별시 강남구 개포로 303	02-574-5410	930	2030
271 갤러리아온누리약국	서울특별시 강남구 선릉로 843 107호	02-517-0129	900	2100
272 갤러리아약국	서울특별시 강남구 학동로20길 55 106호	02-6227-5578	900	2230
493 국동당약국	서울특별시 강남구 선릉로 654 1층 (삼성동 예인빌딩)	02-540-4556	830	2100

Showing 1 to 23 of 176 entries, 5 total columns



## c> 옵션 3

### 1) 옵션 선택 및 확인

3 를 입력 후 제대로 입력되었나 확인해준다.

```

옵션에 따라 필요한 정보를 선택적으로 얻으실 수 있습니다.
1) 현재 운영 중인 약국 정보
2) 요일별 야간 운영 약국 정보 (20:30분 이후 운영 약국)
3) 사용자 주변 약국 정보
> option = readline(prompt = "원하시는 옵션의 번호를 선택해주세요>> ")
원하시는 옵션의 번호를 선택해주세요>> 3
> car("선택하신 옵션은", option, "입니다.")
선택하신 옵션은 3 입니다.
> df_gu = df[df["구"] == place][1]

```

① 옵션 선택  
② 선택 확인

### 2) 결과

사용자가 선택한 지역(예: 강남구)에서 운영하는 모든 약국 정보가 출력된다.

약국명	주소	대표전화	행정구	시작 (월)	마감 (월)	시작 (화)	마감 (화)	시작 (수)	마감 (수)	시작 (목)	마감 (목)
1 강남밝은약국	서울특별시 강남구 강남대로 292 3층 (도곡동 병행빌딩)	02-573-8796	강남구	900	1830	900	1830	900	1830	900	1830
3 강남소화약국	서울특별시 강남구 동은사로114길 42 (삼성동)	02-556-8357	강남구	900	1200	900	1200	900	1200	900	1200
6 건강한국세상병원약국	서울특별시 강남구 강남대로 488 (논현동) 1층	02-545-0029	강남구	530	2929	530	2929	530	2929	530	2929
16 논현필리아약국	서울특별시 강남구 선릉로 641	02-540-7582	강남구	900	2200	900	2200	900	2200	900	2200
17 돌봄약국	서울특별시 강남구 논현로159길 17 101호 (신사동)	070-4246-6066	강남구	800	1800	800	1800	800	1800	800	1800
34 만약국	서울특별시 강남구 도산대로53길 13 1층 (신사동)	02-2039-7662	강남구	930	1830	930	1830	930	1830	930	1830
49 7층애약국	서울특별시 강남구 테헤란로78길 8 7층 일부(대치동 현빌딩)	02-567-0409	강남구	1000	1900	1000	1900	1000	1900	1000	1900
61 선봉약국	서울특별시 강남구 봉은사로54길 9	02-555-8475	강남구	830	1900	830	1900	830	1900	830	1900
84 역삼라이프약국	서울특별시 강남구 테헤란로25길 59 지하1층(역삼동)	02-554-6881	강남구	930	1930	930	1930	930	1930	930	1930
89 E-삼성약국	서울특별시 강남구 일원로 53 (일원동)	02-3412-1254	강남구	900	1900	900	1900	900	1900	900	1900
140 가로수길약국	서울특별시 강남구 압구정로 108 (신사동) 1층 7호	02-518-4201	강남구	930	2000	930	2000	930	2000	930	2000
149 가온약국	서울특별시 강남구 학동로 413 청담빌딩 2층 (청담동)	02-545-0546	강남구	830	1830	830	1830	830	1830	830	1830
169 감마약국	서울특별시 강남구 압구정로 336 (신사동)	02-547-8990	강남구	900	2100	900	2100	900	2100	900	2100
179 1층엔약국	서울특별시 강남구 강남대로 624 ICT TOWER 1층 일부호 (신사동)	02-518-1080	강남구	830	2030	830	2030	830	2030	830	2030
184 강남구청약국	서울특별시 강남구 학동로 지하 346 강남구청역 81층 111호 (신사동)	02-511-6252	강남구	800	2100	800	2100	800	2100	800	2100
185 강남굿모닝약국	서울특별시 강남구 남부순환로 2615 (도곡동) 극동스타플래...	02-529-4155	강남구	940	2000	940	2000	940	2000	940	2000
186 강남리더스약국	서울특별시 강남구 현릉로 569 1층 105호 (세곡동)	02-3414-8001	강남구	900	1930	900	2000	900	1930	900	1930
187 강남메디칼약국	서울특별시 강남구 삼성로 233 (대치동)	02-568-6644	강남구	900	1900	900	1900	900	1900	900	1900
188 가나미스약국	서울특별시 강남구 논현로86길 22 1층 (역삼동)	02-555-5640	강남구	900	2100	900	2100	900	2100	900	2100

## d> 다른 옵션 선택

'옵션을 잘못 입력하셨습니다'라는 문구가 출력되며 결과는 나오지 않는다.

```

> switch(option,
+       "1" = {nowRunning(df_gu, curday)},
+       "2" = {openByDay(df_gu)},
+       "3" = {view(df_gu)},
+       {print("옵션을 잘못 입력하셨습니다.")})
[1] "옵션을 잘못 입력하셨습니다."
>

```

## 5. 기대효과

이를 통해 운영 중인 약국을 쉽게 찾고, 필요한 처치를 빠르게 할 수 있게 되어 시민들의 건강 증진에 도움이 될 수 있을 것이다.