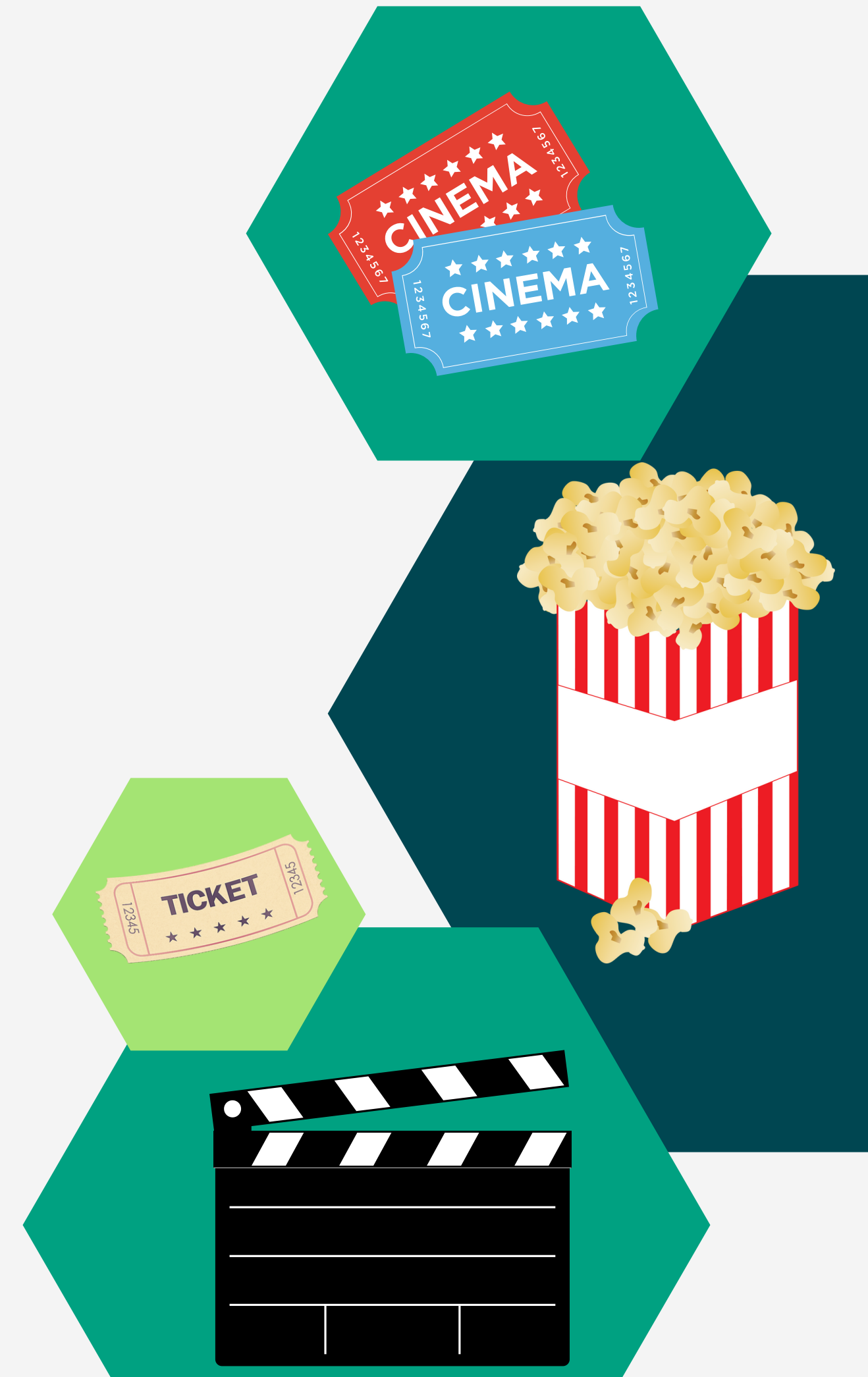# Movie Theater Simulation

## Intro to R Project

Insight group: Amal Almutairi, Nouf Aljohani,
Rahaf Alzahrani, Rawan Alsudias & Salha Nasser

# Agenda

- Overview
- Movie theater simulation function
- Discount function
- VIP screens simulation function
- Snacks Function

# Our Movie Theater

**Offers:**

1. Snacks
2. Discount on certain weeks
3. Regular and VIP Screens.
4. FREE tickets for lucky visitors.

# Our Movie Theater

- **5 Trending Movies!**
- **8 Screens (5 Regular, 3 VIP)**
- **50 Seats per Regular Screen**
- **25 Seats per VIP Screen .**

NOW SHOWING

# Movie Theater Simulation Function

# Function 1: Movie theater simulation function

```r
#this function simulates a movie theater by randomizing the number of visitor, to get the revenue of a
week
movie_theater_simu <- function() {

  week_days <<- rep(0, 7)  # Store totals for each day
  r_mat <<- matrix(nrow=7, ncol=5) # to keep track of all revenue (by movie and day)
  m_mat <<- rep(0,5) # matrix to keep track of total movies revenue
  adult_m <<- rep(0,5)
  child_m <<- rep(0,5)

  adult_mat <<- matrix(nrow=7, ncol=5) # to keep track of adults visitors (by movie and day)
  child_mat <<- matrix(nrow=7, ncol=5) # to keep track of adults visitors (by movie and day)

  total_visitors <<- rep(0,7)

  lucky_visitor <<- lucky_visitors()
```

4

# Function 1: Movie theater simulation function

```r
# iterate through the week
for (i in 1:length(week_days)) {

  # Keep track of total revenue for the day
  #total_revenue <- 0
  # iterate through the amount of screens on a particular day
  for (j in 1:screens) {
    if (i == 6  || i == 7 ){#here we increase the probability of having more people in the weekends
      # Calculate  how many adults and children are watching the movie
      visitors_adults <<- sample(1:seats, size=1,prob = c(rep(0,2,seats-5),0.9,0.9,0.9,0.9,0.9))

      visitors_children <<- sample(1:(seats-(visitors_adults)),size=1)
    }
    else{
      # Calculate  how many adults and children are watching the movie
      visitors_adults <<- sample(1:seats, 1)
      visitors_children <<- sample(1:(seats-(visitors_adults)),1)
    }

    # Calculate the revenue for adults and children
     # if there are lucky visitors -- > only TWO TICKETs are FREE
      if (i * j == lucky_visitor) {
        l_visit <- visitors_adults - 2

        revenue <- (ticket_cost * l_visit ) + (ticket_cost_child * visitors_children)
      }
    else {
      # calculate revenue like usual.
      revenue <- (ticket_cost * visitors_adults ) + (ticket_cost_child * visitors_children)
      }
```

# Function 1: Movie theater simulation function

```r
# Calculate the revenue for adults and children
        revenue_of_adults <- visitors_adults * ticket_cost
        revenue_of_children <- visitors_children * ticket_cost_child

    # Calculate revenue, and add to running total for the day
    r_mat[i,j] <<- revenue # for each screen
    adult_mat[i, j] <<- visitors_adults
    child_mat[i,j] <<- visitors_children
    visitor <<- visitors_adults + visitors_children
  }
  # Save total to the corresponding day
  week_days[i] <<- sum(r_mat[i, ])
  total_visitors[i] <<- sum(visitor)
  # total_revenue <- week_days + snacks
}
# to get the sum of movies columns. [revenue, and attendes].
for (i in 1:5) {
  m_mat[i] <<- sum(r_mat[ , i])
  adult_m[i] <<- sum(adult_mat[ , i])
  child_m[i] <<- sum(child_mat[ , i])
}

}
```

# Graph 1:



Total Tickets Revenue per Day
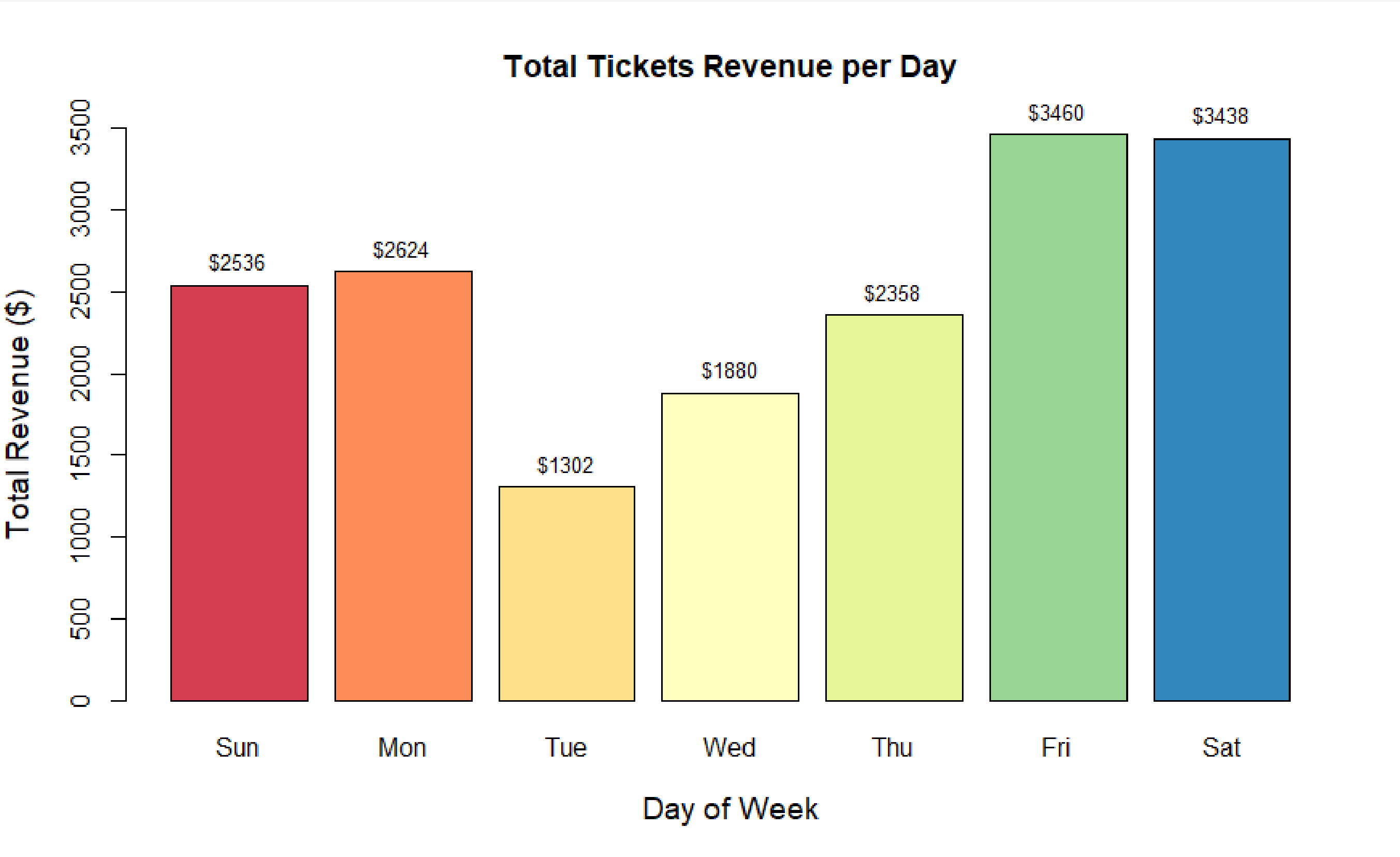
# Graph 2:
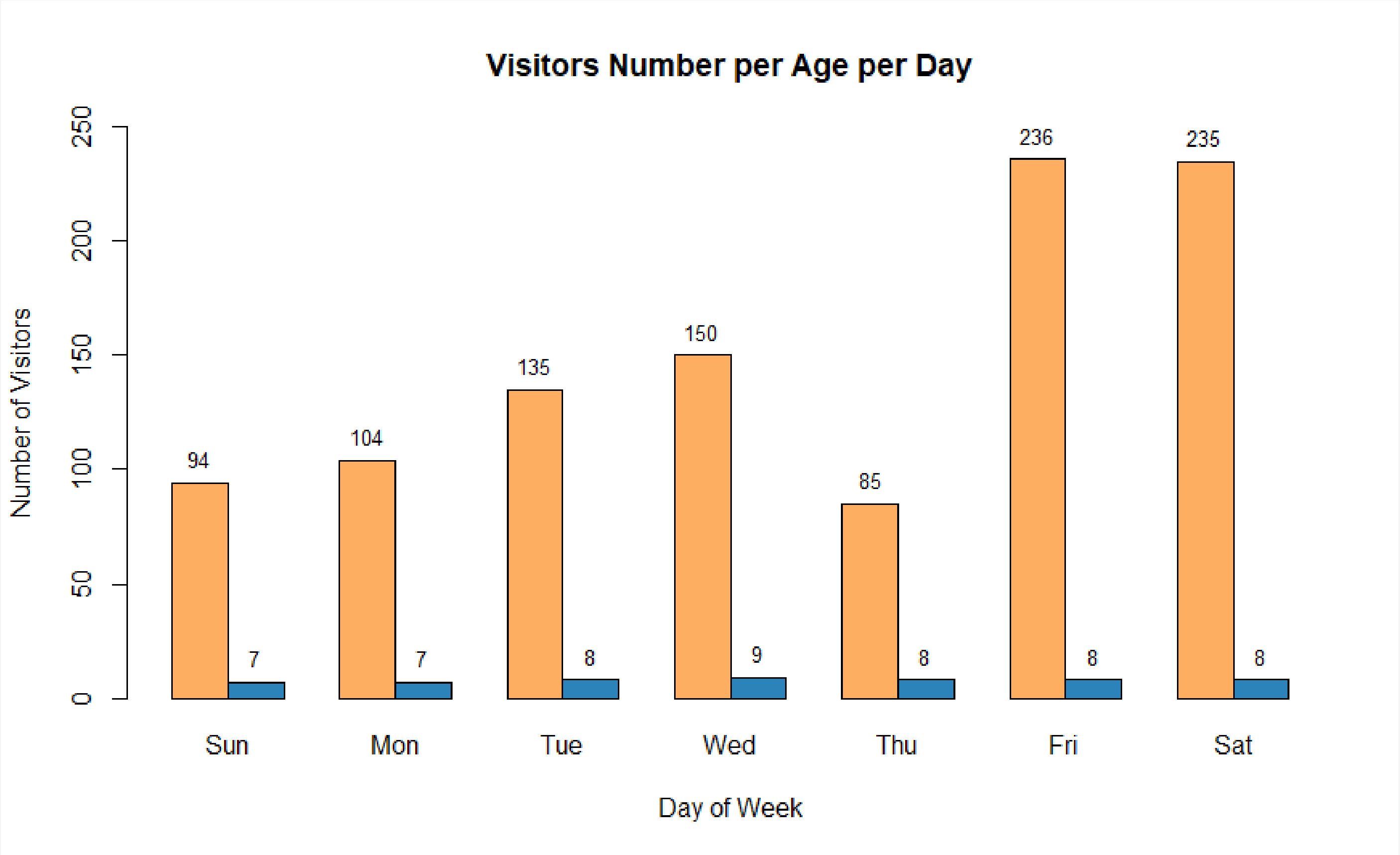
# Discount Function

**10% OFF**

# Function 2 : Discount function

```r
discount_ticket <- function() {
  for (i in 1:length(week_days_discount)) {

    # Keep track of total revenue for the day
    total_revenue_discount <-0
    total_visitors_adults_a_day <- 0
    #total children visitors for each day
    total_visitors_children_a_day <- 0

    # iterate through the amount of screens on a particular day
    for (j in 1:screens) {
      if (i == 6  || i == 7 ){#here we increase the probability of having more people in the weekends
        # Calculate  how many adults and children are watching the movie
        visitors_adults_discount <<- sample(1:seats, size=1,prob = c(rep(0,2,seats-5),0.9,0.9,0.9,0.9,0.9))

        visitors_children_discount <<- sample(1:(seats-(visitors_adults)),size=1)
      }
      else{
        visitors_adults_discount <<- sample(1:seats, 1)
        visitors_children_discount <<- sample(1:(seats-(visitors_adults)),1)
      }

      # Calculate the revenue for adults and children
      revnue_adults <-((ticket_cost - (ticket_cost*discount)) * visitors_adults_discount)
      revnue_children <- ((ticket_cost_child - (ticket_cost_child*discount)) * visitors_children_discount)

      # Calculate revenue, and add to running total for the day
      total_revenue_discount <- total_revenue_discount+revnue_adults+revnue_children
      total_visitors_adults_a_day <- total_visitors_adults_a_day + visitors_adults_discount
      total_visitors_children_a_day <- total_visitors_children_a_day + visitors_children_discount
    }
    # Save total to the corresponding day
    week_days_discount[i] <<- total_revenue_discount

    visitors_adults_in_week_days[i] <<- total_visitors_adults_a_day
    visitors_children_in_week_days[i] <<- total_visitors_children_a_day
  }

}
```

10

# Graph 3



Total Revenue per Day with Discount
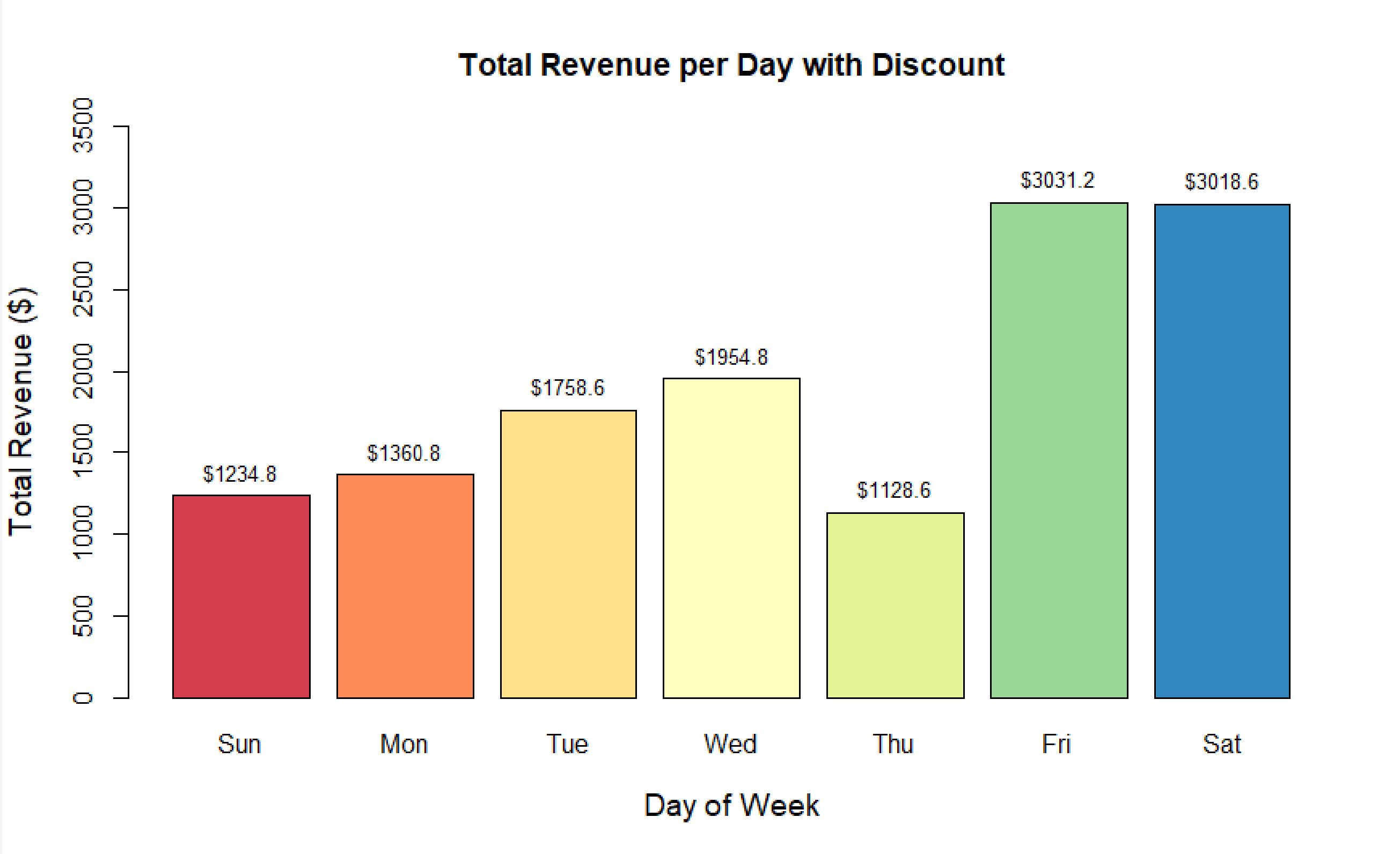
# VIP Screens Simulation Function

# VIP Screens Simulation Function

```r
#function 4:
#this function is to simulate a VIP revenue and visitors

vip_movie_theater_simu <- function() {

  # Cost for VIP adults and children
  vip_ticket_cost <<- 20
  vip_ticket_cost_child <<- 14
  movies <- c('movie1', 'movie2', 'movie3', 'movie4', 'movie5')  # List 5 of your favorite movies
  vip_screens <- 3   # How many VIP screens does the theater have? (assume 1 per movie)
  vip_seats <<- 25 # How many VIP seats does each theater hold
  vip_week_days <<- rep(0, 7)  # Store totals for each day

  #vars to use in the second graph (VIP)
  vip_visitors_adults_in_week_days <<- rep(0, 7)  # Store adults visitors totals for each day
  vip_visitors_children_in_week_days <<- rep(0, 7)  # Store children visitors totals for each day
  ###############
```

# VIP Screens Simulation Function

```
# It's handy to create dataframe, it's great to understand the business as well to represent data in graph
regular_data <- data.frame(total_revenue_per_day = week_days,
                           adults_visitors = visitors_adults_in_week_days,
                           children_visitors = visitors_children_in_week_days,
                           adults_revenue = visitors_adults_in_week_days * ticket_cost,
                           children_revenue = visitors_children_in_week_days * ticket_cost_child)

#dataframe for Vips
vip_data <- data.frame(vip_total_revenue_per_day = vip_week_days,
                       vip_adults_visitors = vip_visitors_adults_in_week_days,
                       vip_children_visitors = vip_visitors_children_in_week_days,
                       vip_adults_revenue = vip_visitors_adults_in_week_days * vip_ticket_cost,
                       vip_children_revenue = vip_visitors_children_in_week_days * vip_ticket_cost_child)
```
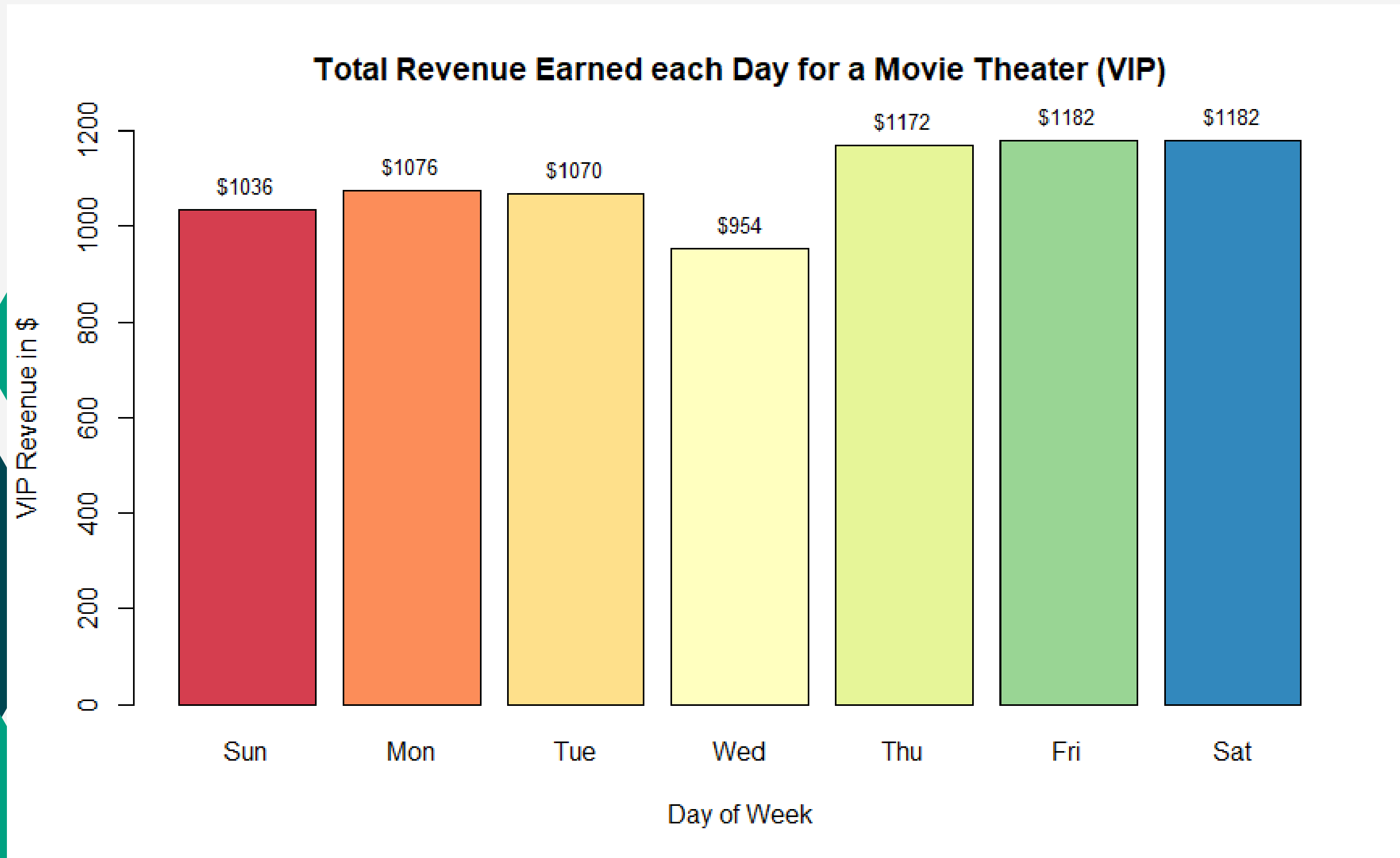
# VIP Screens Simulation Function

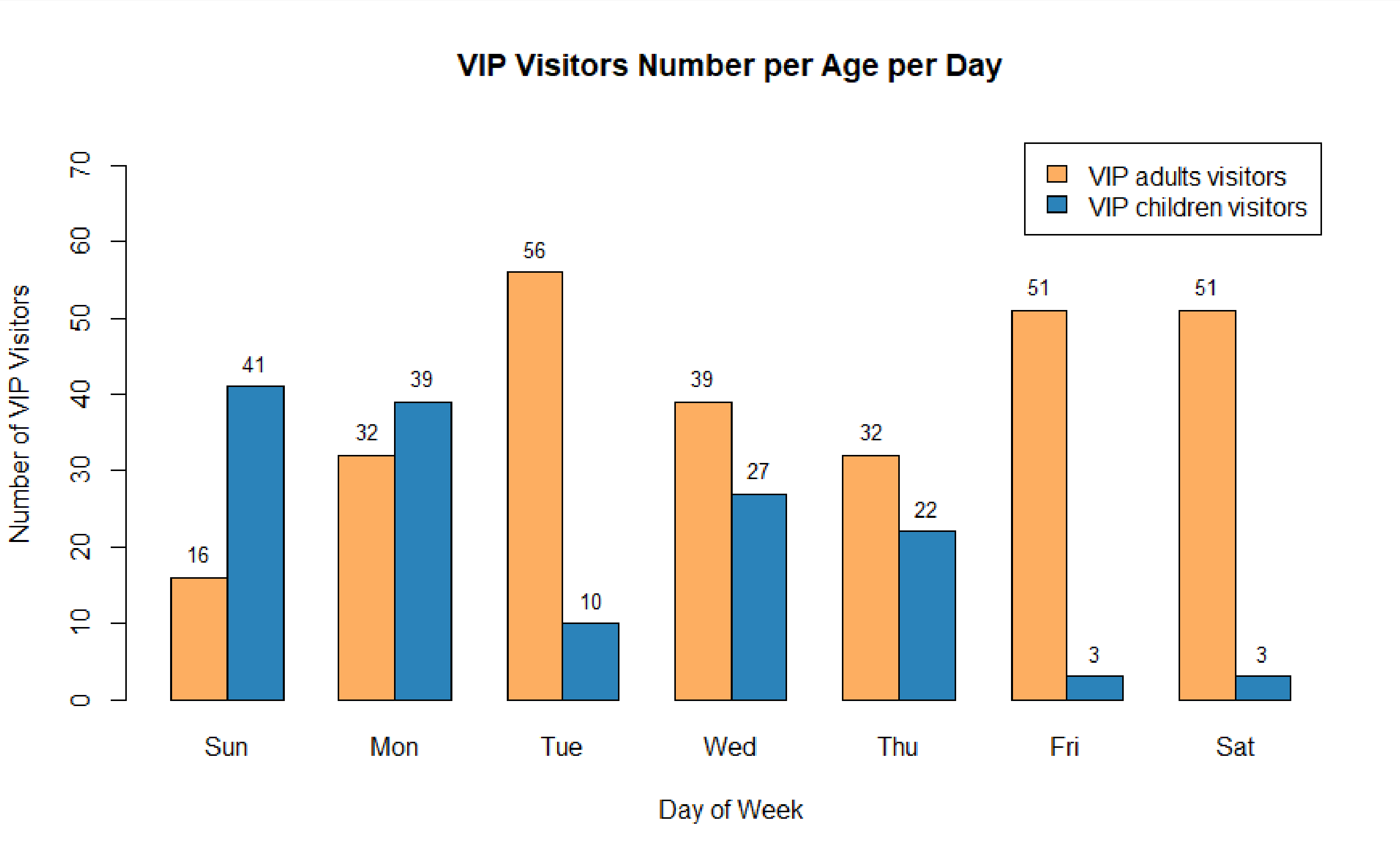| | total_revenue_per_day | adults_visitors | children_visitors | adults_revenue | children_revenue |
|---|---|---|---|---|---|
| 1 | 2536 | 94 | 7 | 1316 | 56 |
| 2 | 2624 | 104 | 7 | 1456 | 56 |
| 3 | 1302 | 135 | 8 | 1890 | 64 |
| 4 | 1880 | 150 | 9 | 2100 | 72 |
| 5 | 2358 | 85 | 8 | 1190 | 64 |
| 6 | 3460 | 236 | 8 | 3304 | 64 |
| 7 | 3438 | 235 | 8 | 3290 | 64 |

# VIP Screens Simulation Function

| | vip_total_revenue_per_day | vip_adults_visitors | vip_children_visitors | vip_adults_revenue | vip_children_revenue |
|---|---|---|---|---|---|
| 1 | 1036 | 35 | 24 | 700 | 336 |
| 2 | 1076 | 44 | 14 | 880 | 196 |
| 3 | 1070 | 43 | 15 | 860 | 210 |
| 4 | 954 | 26 | 31 | 520 | 434 |
| 5 | 1172 | 46 | 18 | 920 | 252 |
| 6 | 1182 | 57 | 3 | 1140 | 42 |
| 7 | 1182 | 57 | 3 | 1140 | 42 |

Tabs: _day_2.R, vip_data, Movie_Theatre.R, movie_simu.R, movie_simu.R*, regular_data, R_day_4.R

Filter

# Graph 4

# Graph 5

# Graph 6



**Revenue per Visitors Age per Day (VIP)**

Legend:
- VIP adults revenue
- VIP children revenue

| Day of Week | VIP adults revenue | VIP children revenue |
|---|---|---|
| Sun | $700 | $336 |
| Mon | $880 | $196 |
| Tue | $860 | $210 |
| Wed | $520 | $434 |
| Thu | $920 | $252 |
| Fri | $1140 | $42 |
| Sat | $1140 | $42 |

# Graph 7



**Revenue Percentage per Visitors Type per Day**

Legend: VIP revenue, Regular revenue

| Day | VIP revenue | Regular revenue |
|-----|-------------|-----------------|
| Sun | 29% | 71% |
| Mon | 29.1% | 70.9% |
| Tue | 45.1% | 54.9% |
| Wed | 33.7% | 66.3% |
| Thu | 33.2% | 66.8% |
| Fri | 25.5% | 74.5% |
| Sat | 25.6% | 74.4% |

Day of Week

# Graph 8



**Weekly Revenue Percentage per Visitors Type**

VIP Revenue
30.4%

Regular Revenue
69.6%
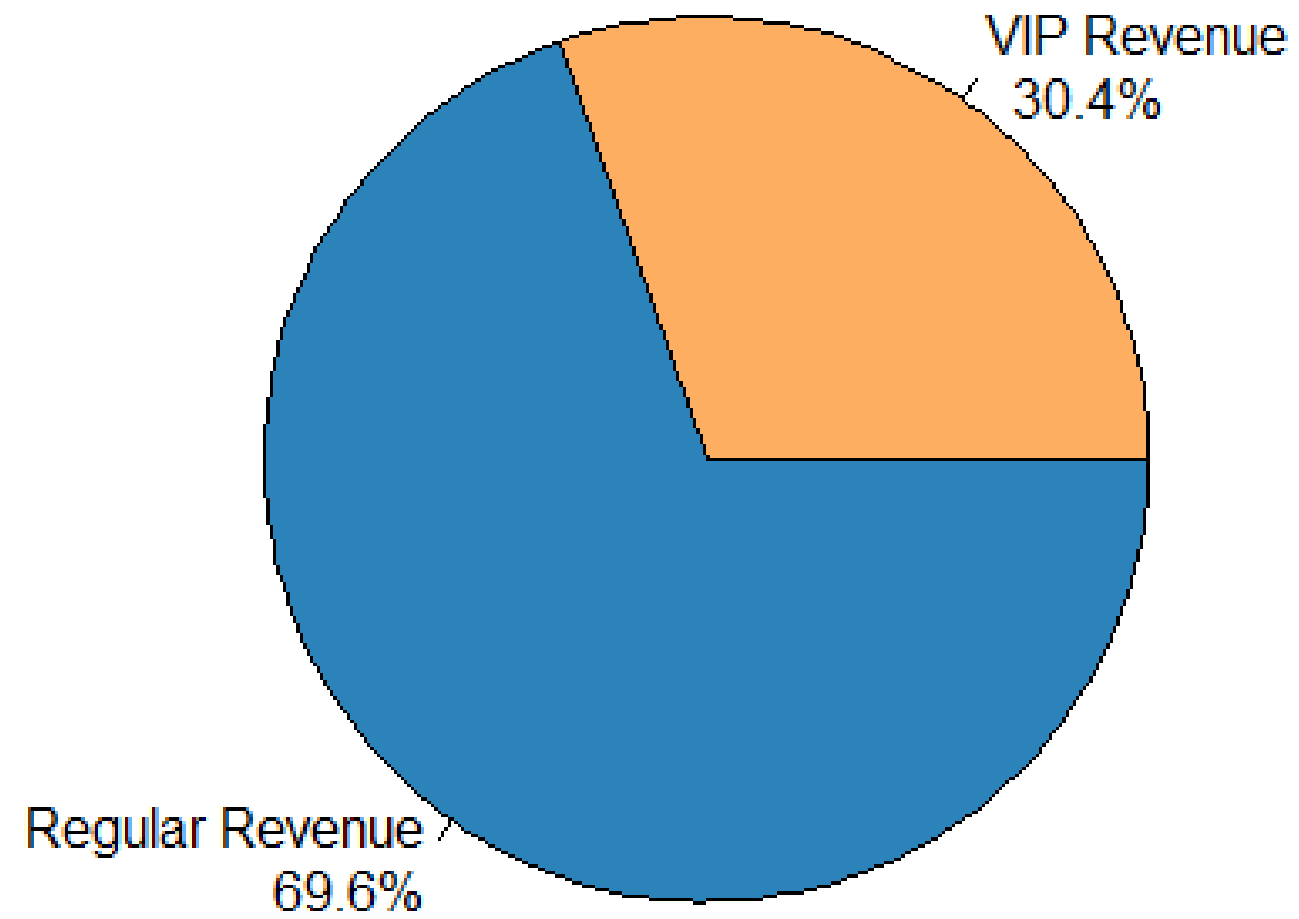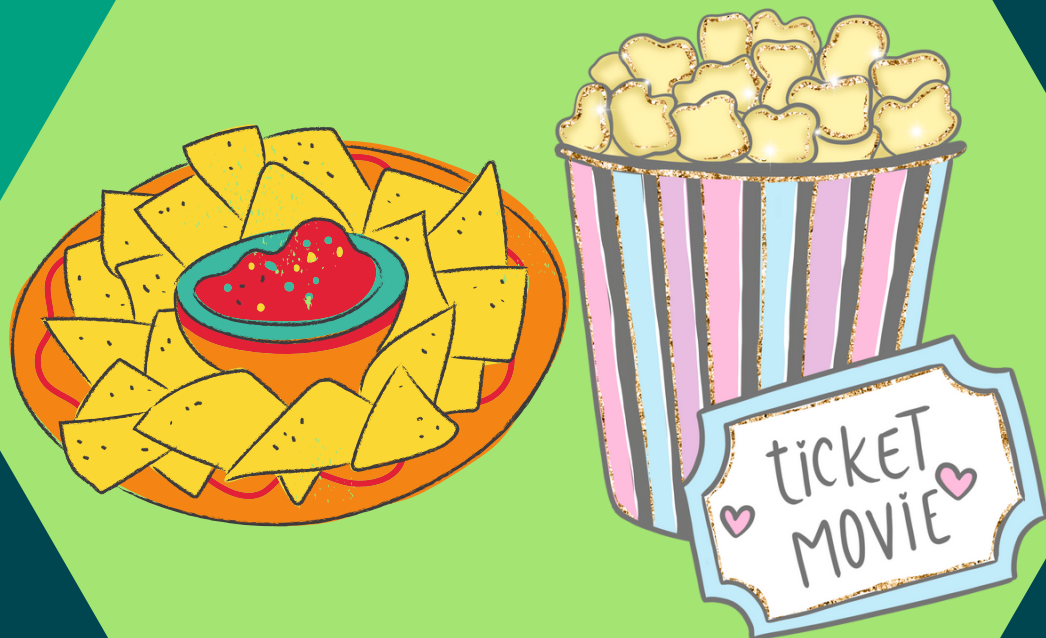
# Snacks Function

# Function 4: Snacks
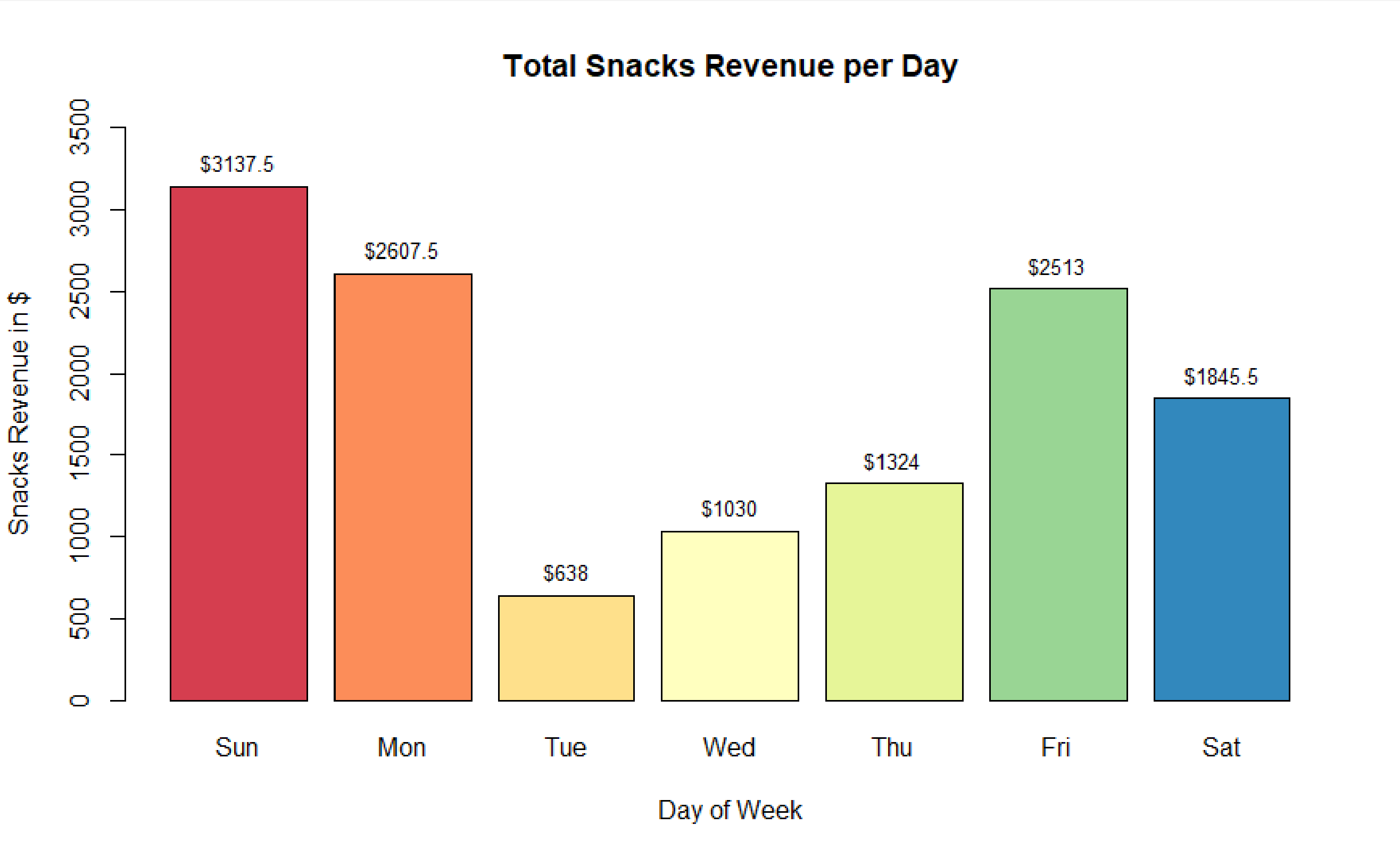
```r
snacks <- c("Popcorn"= 5.5, "Nachos"= 7, "Hotdogs"= 6, "Candy"=4)
```
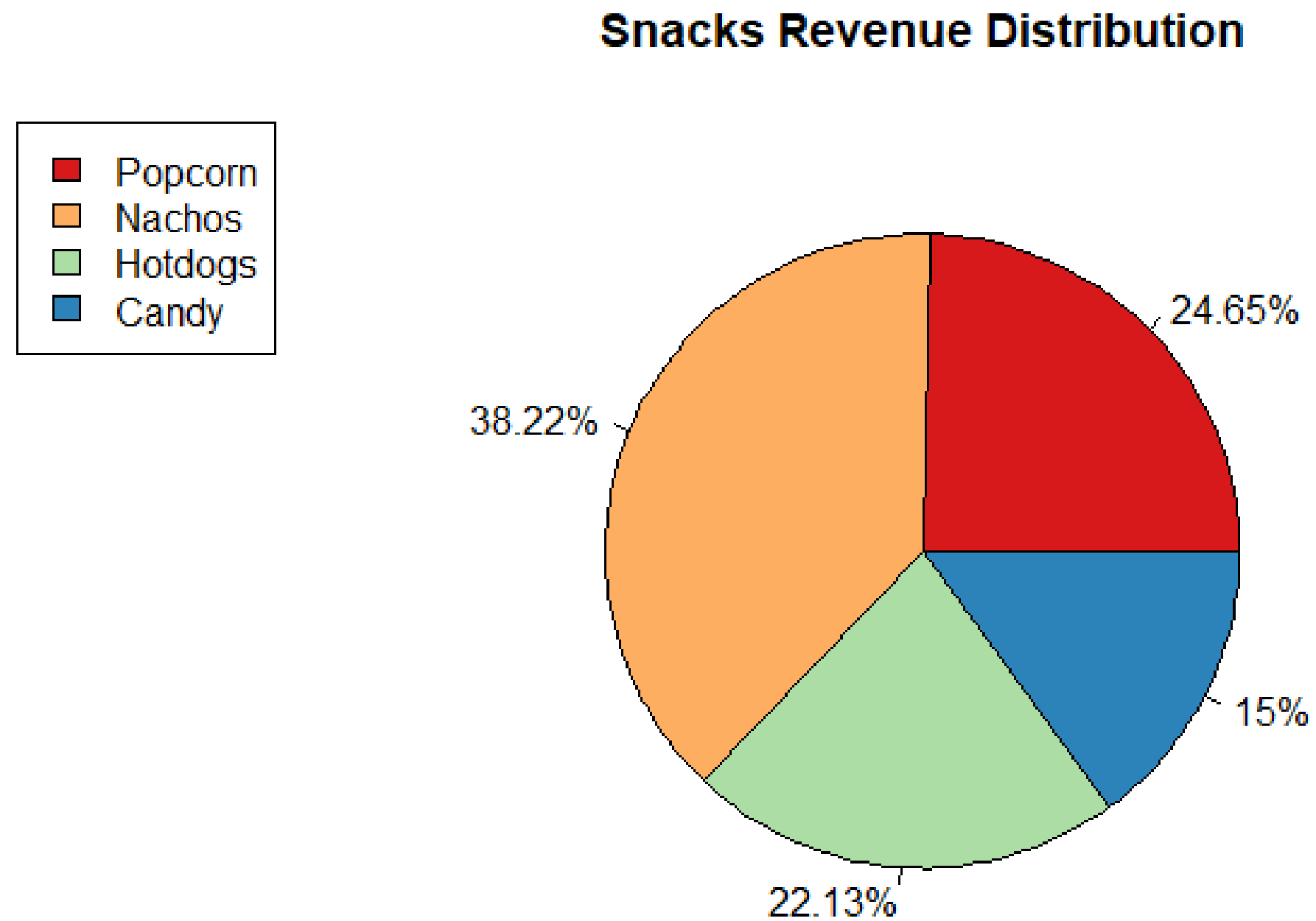
```r
# this function simulates the snacks revenue of the movie theater in a week
snacks_func <- function() {
  snack_reveune <<- matrix(nrow=7, ncol=4)# to keep track of all revenue (by snack and day)
  snacks_week <<- rep(0,7) #Store totals for each day
  s_mat <<- rep(0,4) # to keep track of total snacks revenue
  # Calculate snacks revenue
  for (d in 1:length(week_days)) {# a loop that goes through all 7 days
    #snacks_week <- 0
    for (s in 1:length(snacks)) {  # a loop that goes through all 4 snacks
      # this variable is for storing the snack price multiplied by a random number of people
      selected_snacks <- snacks[[s]] * (sample(1:(total_visitors[d]*4), 1))

      # store the snack revenue
      snack_reveune[d,s] <<- selected_snacks

    }
    #to get the sum of each day.
    snacks_week[d] <<- sum(snack_reveune[d, ])

  }
  # to get the sum of each snack.
  s_mat <<- colSums(snack_reveune)

}
```
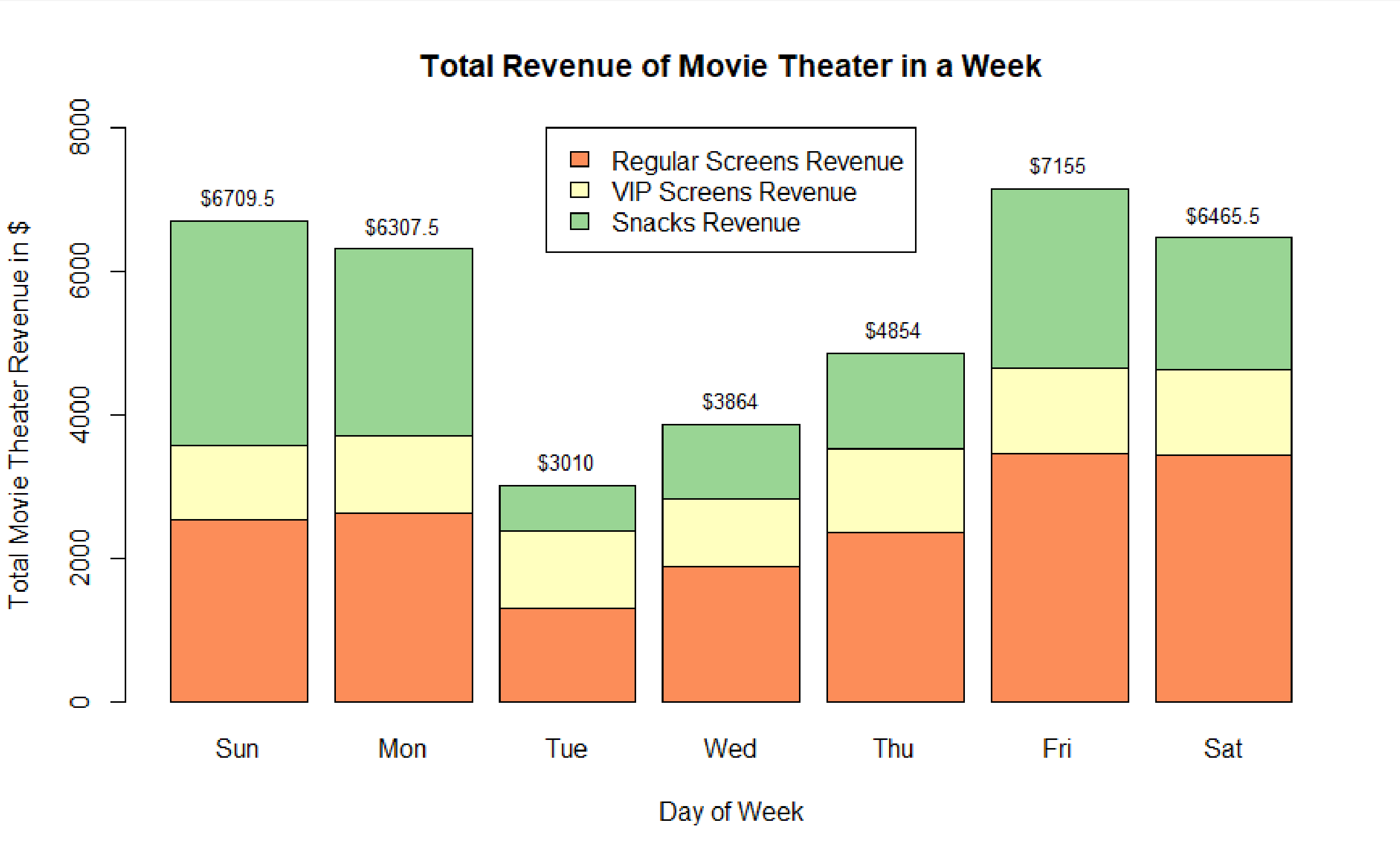
23

# Graph 9

# Graph 10

# Graph 11

# Graph 12



Movies Revenue Distribution in a Week

Legend: Minions 2, Top Gun, Encanto, Paws of Fury, Memory

20.25%, 19.87%, 20.28%, 21.9%, 17.71%