

Programming Challenge

Provakil has a subscription business model where we charge our users a certain amount each month to keep using our service.

We provide a discount to our users if they pay for the whole year upfront, instead of paying monthly.

We also keep our users on certain billing cycles i.e a user who has paid for our subscription always expires on either 1st or 15th of a month. The conditions to determine on what day the user should expire are listed later in this document.

Your task is to create a function **calculate_subscription()** which will take the following arguments:

1. *expiry_date*: String of format dd/mm/yyyy
2. *months_to_buy*: Integer which will be in between 1 and 12 (both inclusive)
3. *monthly_cost*: Integer. Cost of subscription per month, when *months_to_buy* \neq 12
4. *annual_cost*: Integer. Total cost of subscription, when *months_to_buy* = 12

The function should return:

1. *new_expiry*: New expiry date of the user
2. *cost*: Cost of the subscription

The function **calculate_subscription()** should satisfy the following conditions:

- If a user is paying for annual subscription (i.e *months_to_buy* = 12), apart from the 1 year add minimum possible days to user's subscription to bring her to one of our billing cycles (1st or 15th of a month).
Ex: User expiring on 19th June 2018 will expire on 1st July 2019, if she takes an annual subscription. She pays for 12 more days than an year
- If a user is paying for monthly subscription, the first month will end at the farthest possible billing cycle (i.e as close as full month as possible).
Ex 1: User expiring on 19th June 2018 will expire on 15th July 2018, if she takes a subscription for 1 month. She will only pay for 26 days instead of a complete month.
Ex 2: User expiring on 12th June 2018 will expire on 1st September 2018, if she takes a subscription for 3 months. She will pay for 19 days + 2 whole months.
- Use *monthly_cost* to calculate cost of subscription for whole months and *annual_cost* for calculating cost of subscription for a whole year. When calculating cost of subscription for a day, use either (*monthly_cost* / 30) or (*annual_cost* / 365), depending on *months_to_buy*

Test Cases

1. `calculate_subscription("19/06/2018", 3, 1000, 8000)`
Returns: `("15/09/2018", 2866.58)`
2. `calculate_subscription("19/06/2018", 12, 400, 3650)`
Returns: `("01/07/2019", 3770)`

Note

- We will prefer if you solve the problem in Python or Javascript. If you choose to solve it in some other language, please send the instructions for running and testing your solution.
- Use Moment.js (if solving in Javascript), the datetime module (if solving in Python) or whatever their counterparts are in the language your are solving the problem in.