

Project - 01 (DATA ANALYST)

M	T	W	T	F	S	S
Page No.:						
Date:						

Name - Nilansh

→ Problem = Imagine ABC company named ABC runs a real money online gaming app (like wdo)

- Players can :-
1.) Play Games
2.) Deposit money
3.) Withdraw winning amount set in terms

→ To keep the players engaged Company give loyalty points
these points are given based on
1.) How much they deposit
2.) How much money they withdraw
3.) How many games they played
AND whether they deposit more often than
withdraw

OUR TASK :-

1) ABC gives you Dataset :-

- 1.) Game Record (User Gameplay data) \Rightarrow UserId || Played || Date Time
- 2.) Deposit Record \Rightarrow UserId || Date Time || Amount
- 3.) Withdrawal Record \Rightarrow UserId || Date Time || Amount

what we have to do ? :-

PART - A \Rightarrow Find loyalty points for each player

\Rightarrow push user user set

$$\text{Loyalty Points} = 0.01 \times \text{Total Deposit Amount} +$$

$$0.005 \times \text{Total Withdrawl Amount} +$$

$$0.001 \times \max(\# \text{Deposit} - \# \text{Withdrawls}, 0) +$$

$$0.02 \times \text{Total Games Played}$$

Based on reason

(TEYJANA ATAC) 10

M	T	W	T	F	S	S
Position						Page No.:
Date:						10

- Q(2) Answer these 5 things
- 1) Points for players on 4 specific day - slots (eg - 2nd 5th)
 - 2) Total points per player in Oct.
 - 3) Who are the top players (rank them)
 - 4) What is the average amount?
 - 5) What is the average deposit per user & games per user?

Start off with (pseudo) players are only part of (-)

PART-B → ABC wants to give ₹ 50,000 in bonus to the top

50 players

⇒ You need to find (somehow)

1) Suggest a fair way to divide this money
months (not days)

2) Should it be based on loyalty points?

Games played? Something else?

PART-C ⇒ Feed back

→ Give your opinion :-

1) Is the formula fair? (using data)

between (=(slots 20) result) can we improve

formula / function / formula (← break + mod)

function / formula / function (← break + mod)

Approach :-

1) Step 0 → Load and prepare all data (← A-TAB)

we have these data :-

+ transaction (1) user-gameplay → tells how many games

+ transaction (2) Deposit-data

+ transaction (3) withdrawl - data

- user-gameplay → tells when & how many Games each user played

- deposit-data → tells when & how much user deposit

- withdrawl - data → tells when & how much user withdraw

M	T	W	T	F	S	S
Page No.:	81	82	83	84	85	86

Step 01: Load the Data

- load all the data into the python dataframe
- convert the data column to proper format and get them ready for filtering and then filter the slot

2.) Step: 02 → Filter the Date by time slot

We know,

- Slot S1 is from → 12:00 AM to 12:00 PM
- Slot S2 is from → 12:00 PM to 12:00 AM

3.) Step: 03 ⇒ Group the data by each user

- Till this step we have filtered data
- We will group them with their user id
- AND calculate :- sum of play with user id

• S - time 1.) Total Games played

2.) Total Deposit Amount

3.) Total Withdrawal Amount

4.) Number of deposit

5.) Number of withdrawal measured at 6

4.) Step: 04 ⇒ Till this step we have already filtered out the data and grouped them with their user id

Now we will apply loyalty formula for each user.

$$\text{Loyalty} = (0.01 \times \text{Total deposit}) + (0.05 \times \text{Total withdrawal}) +$$

$$\text{Points} = (0.001 \times \max(\#deposit - \#withdrawals, 0)) + (0.2 \times \text{total game played})$$

example :- we have 11 a data collected as

User	Deposit	Withdrawal	#Deposit	#Withdrawal	Games Played
101	₹1000	₹500	1	1	10

Then, we have to calculate $(1000 \times 0.01) + (500 \times 0.05) + (0.001 \times \max(1-1, 0))$

$$\text{loyalty points} = (0.01 \times 1000) + (0.05 \times 500) + (0.001 \times \max(1-1, 0)) \\ + (0.2 \times 10) \Rightarrow 10 + 2.5 + 0.001 + 2 \Rightarrow 14.50 \text{ points}$$

5.) Step: 05 ⇒ we will answer all the questions asked from PART - A, B, C .

• Movie clear & Short Summary of

our Approach :- ~~using~~ our own skills with the help of others.

→ Data → Filter by Date → Grouping by Player

Applying formula (2)

Answer (Part A, B, C)

2018-01-21 09:00:15 → message is true.

let's Begin :-

questions and points with quotes) ← easiest (s)

* Create a file .pynb in VS Code to begin coding.

* Import pandas as pd, used for working with excel/csv data

- load the data | game, deposit, withdraw)

Question → sheet 0  sheet - 01  sheet - 1  sheet - 2  sheet - 3

- Convert Datetime + Columns to Real Date time format

Raw Data → plain text (String). → "2022-10-02 11:45:00"

⇒ To perform action on dates we convert the Date

"2022-10-02 11:45:00" → Python (see this as a text)

So we tell python that this is datetime.

`pd.to_datetime("2022-10-02 11:45:00")`

+ (Convention) \hookrightarrow python) industry understand that this is one

Wurst-Lister & S. et al (2014) date/time = timestamp (ms) * 1000

* Let's create Date and the Slot both all 103rd data in storage

Date → 2022-10-02 (Friday) | 4120x1024 | 128 KB

1 hour → ± 14 (2 PM) → 00017 → 101

slot → s_1 (if hour < 12), s_2 o

will help the python to understand date, time & std lib.

Date/Time: 10-21-11 14:51:00 Hour: 13 Slot: S1

2022-10-02 16:38:00

→ Here a error will face because of date format? So it is cleaned & parsed dates.

Fixed date format → pd.to_datetime ('-.-', dayfirst)

→ then we calculated loyalty points = Israel

→ filtered October month

→ Then we grouped and merged all data

→ calculate loyalty points and then ranked players

Done with PART - A

PART - B

→ Calculate bonus money divided in 50 users

Steps:- 1) Take or sort first 50 players from October leaderboard
 2) Then we normalize the points (loyalty, game played)
 normalizing → reducing or make it reduce to compare data easily.

eg:- A → 1000 points B → 500 points

$$1000/1000 = 1.0 \quad 500/1000 = 0.5$$

Now it comes on scale 0 → 1 easy to compare

3) Bonus Distribution

↳ loyalty (x)

↳ Gameplay (x)

↳ Something else (v)

lets do both loyalty + Gameplay here as it will be balance of both the behaviors and it is good for our user to make them trust us more. but lets do give loyalty customer advantage as there are our loyal user so, loyalty + Gameplay

70% + 30%

- 4.) we calculate final score of each user with respect to
- 5.) Now final score of each user divide it with 50 user

6.) Final Step we will multiply it by ₹50,000 to get the amount for each player.

		Display result	$x/1000$	$y/90$	(FS)	Final score	Final amount
eg:-		Player	loyalty points	played	Norm	Norm	norm
A	1000	50	1.00	0.56	0.868	0.386	₹ 19300
B	500	60	0.50	0.67	0.551	0.245	₹ 12250
C	800	80	0.80	0.89	0.827	0.368	₹ 18400

$$A \rightarrow T = 2.246 \text{ action score}$$

Done with Part - B

Part C -

draw of no bistro's perform scenario discussed

- Current formula is good as it reward both the money and activity (user's the points are given to every user).
- It also helps to encourage user to be active.

but there are some issues:

$$E.A = 0.001/0.05 \quad E.L = 0.001/0.001$$

- Same points to every type of user is not fair.
- low point in playing game is compared to money deposit.
- No bonus for consistency.
- can give negative loyalty.

- Improvement:-
- Add bonus for daily activity.
 - Add points for winning and performing well in games.
 - make recent activity count more as compare to old ones.
 - minimum activity requirement to prevent cheat.
 - limit impact of withdraws or reduce their weight of previous day.
 - new idea is to use less bonus for withdraws if user activity is high, new idea is to use less bonus for withdraws if user activity is low.