

# UserCase Project-IE6700

*Database management for Analytics*

*“The most profitable Monetization model on the Internet.”*

----- "Computational Advertising"

**Group: 8**

**Author Name: Rundong Xu, Haotian Chen**

## I. Business problem definition & requirement

Marketers can segment customers based on their habits (purchasing and browsing data etc.), track their behavior on a website (it's even easier if the customer is a registered user), and use customers' activity on social media sites to find the most receptive prospects.

Before the Internet, marketing was relatively simple. Now, there are so many ways to reach customers (social media platforms, online ads, TV commercials, print ads, emails, etc.) that things get very complicated. Today, we focus the media platform owning soaring growth in the market, which own the excellent monetization ability.

Relying on the huge user groups of Tiktok, Monetization platform has accumulated a rich targeting system, dividing users into 9 major targeting dimensions and 1000+ groups of people, which is conducive to more accurate advertising for advertisers.

- So marketers have many ways to track a campaign and determine how successful it was. Among these, increases in revenue and profit are probably the most important metrics.
- When running multiple campaigns, each one should be tracked separately. Ideally, results would be measured at the customer level (Did the customer specific favorite content ? Did they interactive with advertisement such as comment, like, sharing and click? Whether the user made a real purchase? )

Usually, Marketers can segment customers based on their habits (purchasing and browsing data etc.), track their behavior on a website (it's even easier if the customer is a registered user), and use customers' activity on social media sites to find the most receptive prospects.

There are three sides for Monetization in internet marketing based on Platform view.

### 1. Demand Side

The Demand Side usually include "**Advertiser**" owning budget to advertise their brand or product and the "**Agency**" who contracted with Advertiser and are responsible for the whole cycle of advertisement from Planing and output the campaigns, designing to the examination of final performance of advertisement.

## 2. Advertising Platform

In our project , the "**Advertising platform**" is the monetization platform that corresponds to the needs of Demand Side and achieves transactions of advertisement, they are responsible for advertising various advertisement in front-end APP to target user, and present the performance of these advertisement to agency and advertiser, thus closing the entire monetization procedure chain.

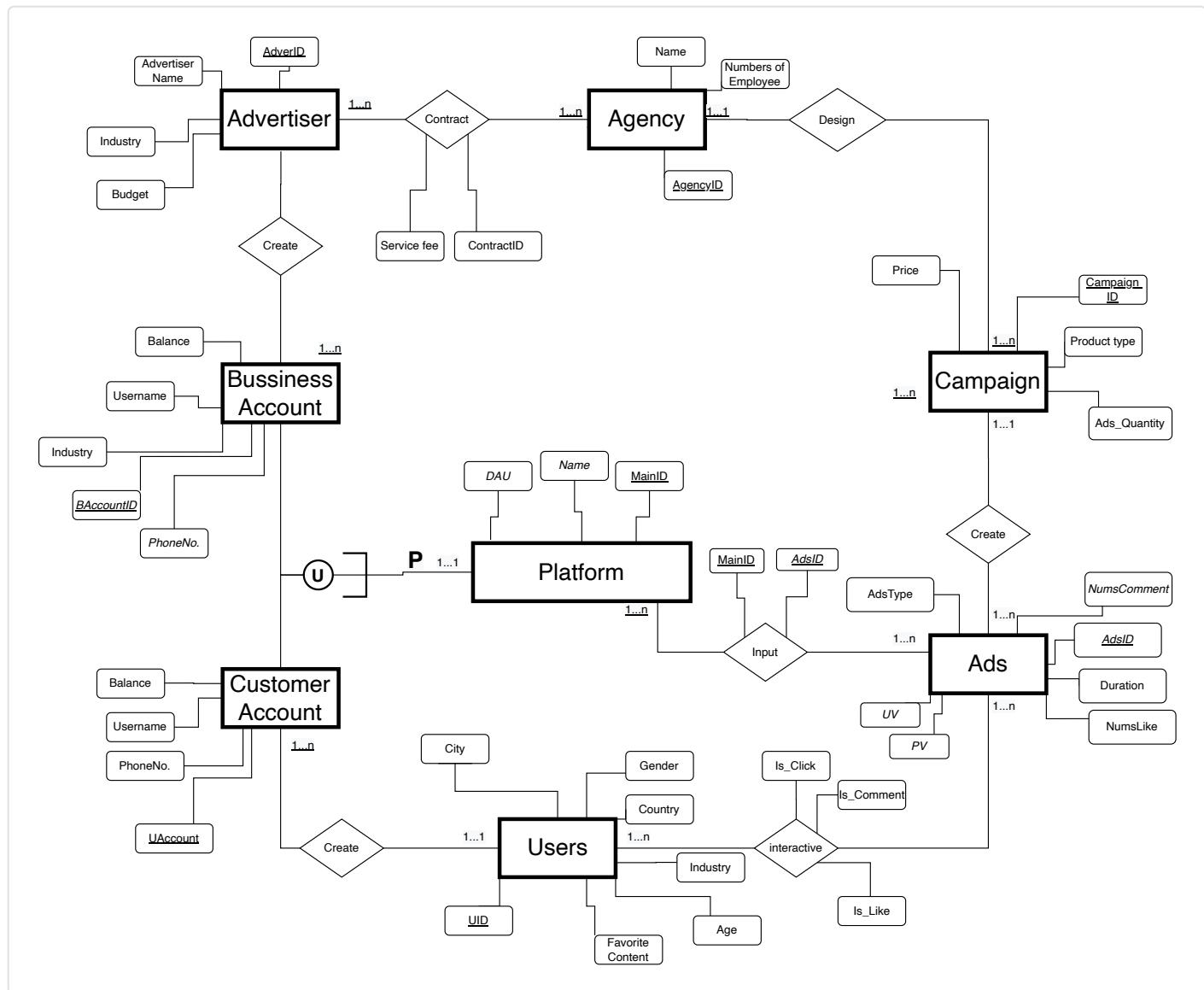
## 3. Target users

The "**Target user**" is based on the needs of the advertising audience provided by the advertiser and agency. Among the active users on tiktok, the specific group delineated by the user portrait is used, so that the advertisement can be accurately targeted to the target group who is most likely to make a purchase operation.

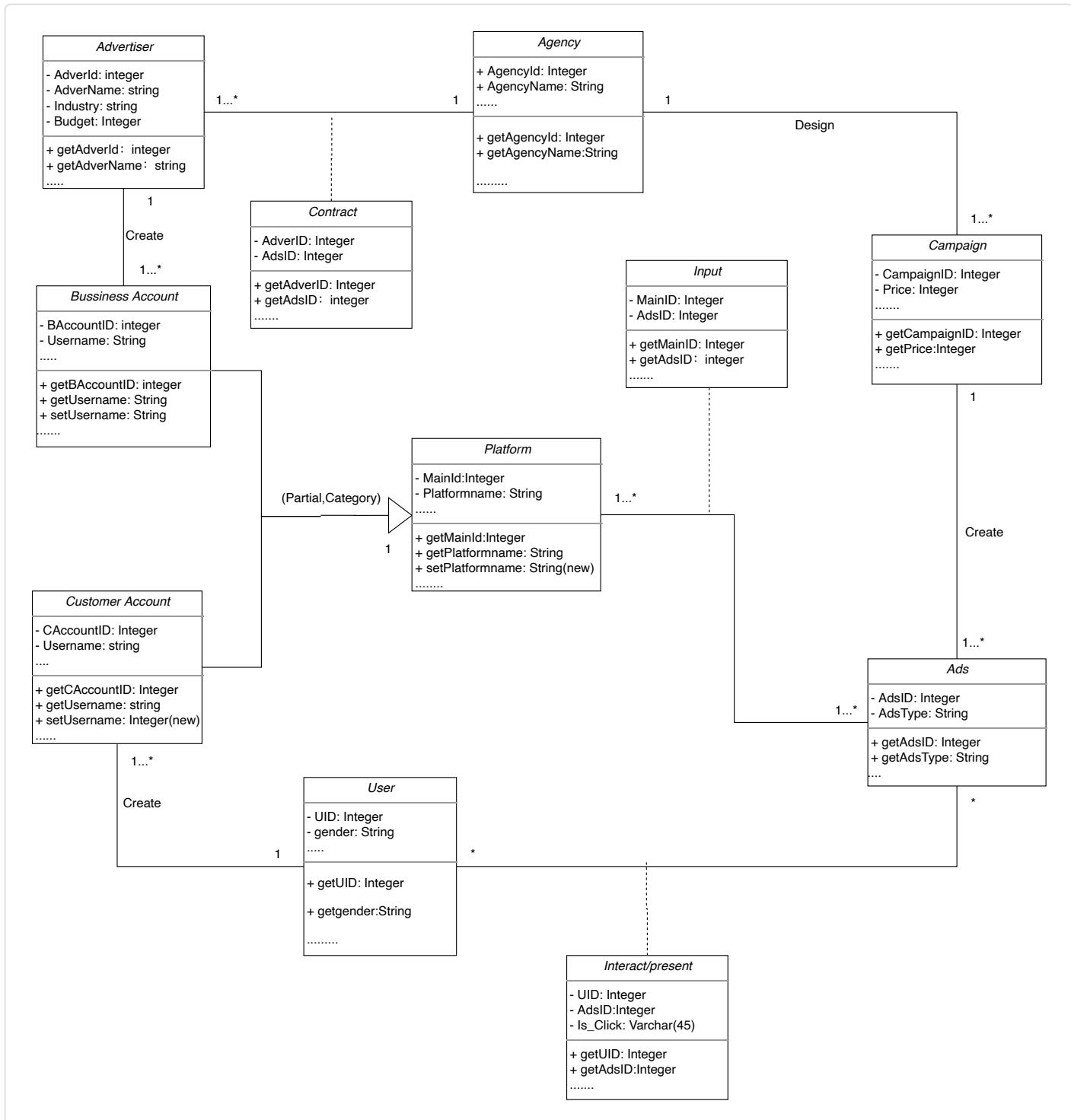
All in all, the basic business logic between aforementioned three core side are obvious. The "**Advertiser**" provide **budget, product type and the anticipation of performance of advertisement to "Agency"**, they analyzing the need and then planing and design the advertisement, besides they register the account and provides and complete the data collection table in "**Platform**", the "**Platform**" have two type of account could be registered, if the Agency provide the intergal digital advertisement so that they only want to advertising in Tiktok, the Ads Platform would allocate campaign the agency created to evaluate and push to front-end, in order to present to their target user based on these campaign. Usually, If the Agency need some help such as: transferring their designing of adverstisment to digital or need the professional resource of advertising videos, the Partener platform would help them to make the contract with agency in order to complete and design and allocate the employee to make campaign, and advertising the advertisement in Tiktok platform, but in this project, we focus the main lifetime cycle the user's behavior also be recorded such as "Like, comment, UV, PV, completion rate and CTR etc.", the Monetization platform provide the data performance feedback to the agency. That is the business closed loop of Tiktok advertisement.

# II. Conceptual data model (EER)

## 1. EER Diagram



## 2. Conceptual data model (UML)



### III. Mapping Conceptual Model to Relational Model

- Primary Key - Underlined      Foreign Key - *Italicized*

**Advertiser**(AdvertiserID, AdvertiserName, Industry, Budget)

**Contract**(AdvertiserID, AgencyID, ServiceFee)

- AdvertiserID foreign key refers to ID in Advertiser, NOT NULL.

- AgencyID foreign key refers to ID in Agency, NOT NULL.

**Agency**(AgencyID, AgencyName, Employee\_Scale)

**BusinessAccount**(BAccountID, Username, Industry, Balance, *MainID*, *AdvertiserID*)

- *MainID* ID foreign key refers to ID in Platform, NOT NULL.
- *AdvertiserID* foreign key refers to ID in Advertiser, NOT NULL.

**Campagin**(CampaginID, Product\_Type, Ads\_Quantity, Price, *AgencyID*)

- AgencyID foreign key refers to ID in Agency, NOT NULL.

**Platform**(MainID, PlatformName, DAU)

**Input**(MainID, AdsID)

- *MainID* foreign key refers to ID in Platform, NOT NULL.
- *AdsID* foreign key refers to ID in Advertisement, NOT NULL.

**Ads**(AdsID, Ads\_Type, Duration, NumsLike, NumsComment, UV, PV, *CampaginID*)

- *AdsID* foreign key refers to ID in Advertisement, NOT NULL.
- *CampaignID* foreign key refers to ID in Campaign, NOT NULL.

**Interaction**(AdsID, Uid, Is\_Like, Is\_Comment, Is\_Click)

- *AdsID* foreign key refers to ID in Advertisement, NOT NULL.
- *UID* foreign key refers to ID in Users, NOT NULL.

**User**(Uid, Gender, Age, Industry, Edu\_Level, FavoriteContent, City, Country, *CAccountID*)

- *CAccountID* foreign key refers to ID in CustomerAccount , NOT NULL.

**CustomerAccount**(CAccountID, Username, PhoneNo, Balance, *Uid*, *MainID*)

- *MainID* foreign key refers to ID in Platform, NOT NULL.
- *UID* foreign key refers to ID in Users, NOT NULL.

## IV. Implementation of Relation Model via MySQL and NoSQL

### 1. MySQL Implementation:

- a. **Query 1** Return all the advertiser names with budget and the corresponding agency names according to the size of the budget to distinguish the customer level, 5w-10w is LA(LargeAccount), 10w-50w KA(KeyAccount), greater than 50w SKA(SuperKeyAccount) and sorted according to the size of budgetd.

## SQL

```

1 select AdvertiserName, AgencyName, budget,
2       case when budget > 500000 THEN "SKA"
3       WHEN budget < 100000 THEN "LA"
4       ELSE "KA"
5       END AS "Clientlevel"
6 from advertiser
7 join contract on advertiser.advertiserID = Contract.advertiserID
8 join agency on contract.agencyID = agency.agencyID
9 order by budget DESC;

```

**Output:**

citi	Advertiser	606308	SKA
at&t	vendorsend	602424	SKA
Amazon	Recanalu	587982	SKA
psa peugeot citroen	Recanalu	561688	SKA
european commission	Durena	557591	SKA
Toyota	Promovea	542058	SKA
education nationale	Ednai	541813	SKA
Walmart	conjex	535463	SKA
abn amro bank n.v.	conjex	524321	SKA
cummins inc.	vendorsend	523087	SKA
bank of america	Rado	497356	KA
bradesco	Escubot	493831	KA
infosys	advertaria	489601	KA
MICHELIN	sixza	486816	KA
nationwide insurance	Recanalu	466617	KA
PEPSICO	Escubot	446898	KA
imss	Brandstorm	400463	KA
stay at home mom	Signsy	389715	KA
ford motor company	Ingenious	363446	KA
national australia bank	ocupa	361824	KA
amgen	Publician	361507	KA
trw	conjex	361077	KA
abbvie	Rado	346677	KA
pwc	Compose	316274	KA
jpmorgan chase & co.	ocupa	294731	KA
mcdonald's corporation	sixza	289416	KA
google	bipmi	286682	KA
kaiser permanente	Indabox	285220	KA
jcpenney	Promovea	265107	KA
hsbc	Benis	262015	KA
capgemini	Ednai	260329	KA
cvs pharmacy	advertaria	259947	KA
BMW	Benis	256245	KA
lenovo	Ingenious	234823	KA
united states air force	advertaria	73108	LA
DELTA	Compose	65582	LA

ncr corporation	Nexcog	61639	LA
vodafone	Indabox	54414	LA

- b. Returns all advertisers and agency names that are KAaccount, contracted service fees and agency fees, CommisionRate, arranged in proportion. (Check the service fee and commision rate of contract between KAaccount and agency.)**

## SQL

```

1 select AdvertiserName, AgencyName, Employee_Scale, ServiesFee, ServiesFee/budget as CommisionRate,
2       case when budget > 500000 THEN "SKA"
3       WHEN budget < 100000 THEN "LA"
4       ELSE "KA"
5       END AS "Clientlevel"
6 from advertiser
7 join contract on advertiser.advertiserID = Contract.advertiserID
8 join agency on contract.agencyID = agency.agencyID
9 where budget < 500000
10 AND budget > 100000
11 order by CommisionRate DESC;

```

**Output:**

	AdvertiserName	AgencyName	Employee_Scale	ServiesFee	CommisionRate	Clientlevel
▶	lenovo	Ingenious	3470	47314	0.2015	KA
◀	abbvie	Rado	6527	71157	0.2053	KA
◀	pwc	Compose	6231	65967	0.2086	KA
◀	stay at home mom	Signsy	1602	82555	0.2118	KA
◀	kaiser permanente	Indabox	2487	62317	0.2185	KA
◀	bradesco	Escubot	3832	114299	0.2315	KA
◀	BMW	Benis	373	59804	0.2334	KA
◀	jpmorgan chase &...	ocupa	5072	68950	0.2339	KA
◀	trw	conjex	8587	85969	0.2381	KA
◀	capgemini	Ednai	5367	63458	0.2438	KA
◀	imss	Brandstorm	5646	98582	0.2462	KA
◀	PEPSICO	Escubot	3832	111177	0.2488	KA
◀	national australia...	ocupa	5072	91257	0.2522	KA
◀	amgen	Publician	6495	92006	0.2545	KA
◀	cvs pharmacy	advertaria	2687	67225	0.2586	KA
◀	google	bipmi	5089	75746	0.2642	KA
◀	infosys	advertaria	2687	132922	0.2715	KA
◀	nationwide insura...	Recanalu	7231	128589	0.2756	KA
◀	jcpenney	Promovea	6186	73192	0.2761	KA
◀	bank of america	Rado	6527	140807	0.2831	KA
◀	hsbc	Benis	373	74687	0.2850	KA
◀	ford motor company	Ingenious	3470	103574	0.2850	KA
◀	mcdonald's corpo...	sixza	7817	85353	0.2949	KA
◀	MICHELIN	sixza	7817	145298	0.2985	KA
◀						
◀						
◀						

**c. Query 3 Return platform name, DAU, number of Ads, evaluate the commercialization performance of all platforms**

## SQL

```

1 select p.platformName,Count(DISTINCT(a.adsID)) as NumsOfAds, ANY_VALUE(DA
U) AS DAU
2 from platform p
3 inner join input i on p.mainID = i.mainID
4 inner join ads a on i.adsID = a.adsID
5 group by p.platformName
6 order by NumsOfAds desc;

```

**Output:**

	platformName	NumsOfAds	DAU
▶	Outbrain	6	790925
▶	Kaggle	5	801221
▶	Reddit	5	75420
▶	Twitter	5	485314
▶	Amazon	4	114192
▶	Yahoo	4	28555
▶	Bing	3	641037
▶	Facebook	3	794025
▶	Fyber	3	849664
▶	Amobee	2	360771
▶	Baidu	2	611420
▶	Microsoft	2	204201
▶	Pinterest	2	378273
▶	Quora	2	618066
▶	Tiktok	2	452202
▶	YouTube	2	546914
▶	Chegg	1	675379
▶	Choozie	1	548272
▶	Google	1	87429
▶	Instagram	1	813222
▶	LinkedIn	1	978700
▶	Smarty	1	283179
▶	Snapchat	1	30373

- d. Query 4 Returns ad\_id, ad\_type, duration, DV(Depth of visit), DAU with more than 500 likes or more than 300 comments on Tiktok platform. (Check what Advertisement have actively users who interact with it and the performance of ads(Depth of visit))**

## SQL

```

1 Select AdsID,Ads_Type,Duration_in_sec, PV/UV as Depth_of_Visit, DAU
2 From Ads
3 Natural join Platform
4 Natural join Interact_or_Present
5 Where platformName='TikTok'
6 and NumsLike>500
7 Union
8 Select AdsID,Ads_Type,Duration_in_sec, PV/UV as Depth_of_Visit, DAU
9 From Ads
10 Natural join Platform
11 Natural join Interact_or_Present
12 Where PlatformName = 'TikTok'
13 and NumsComment>300
14 order by Depth_of_Visit desc;

```

Output:

	AdsID	Ads_Type	Duration_in_s...	Depth_of_Visit	DAU	
▶	40056	Native	31	1.4833	452202	
	40050	Video	53	1.3406	452202	
	40019	Mobile	57	0.8328	452202	
	40040	Mobile	34	0.6719	452202	
	40001	Native	23	0.6023	452202	
	40072	Mobile	59	0.5193	452202	
	40043	Native	42	0.4597	452202	
	40066	Display	48	0.4559	452202	
	40080	Display	35	0.4496	452202	
	40085	Display	45	0.4254	452202	
	40032	Video	40	0.3658	452202	
	40064	Display	49	0.3496	452202	
	40079	Native	33	0.3430	452202	
	40047	Video	20	0.3341	452202	
	40049	Native	54	0.3053	452202	
	40084	Mobile	33	0.3050	452202	
	40025	Display	37	0.2697	452202	
	40071	Native	38	0.2513	452202	
	40031	Native	19	0.2449	452202	
	40029	Native	17	0.2416	452202	
	40055	Display	22	0.2342	452202	
	40008	Video	60	0.2232	452202	

40058	Video	35	0.1930	452202
40018	Mobile	56	0.1925	452202
40004	Display	44	0.1898	452202
40003	Native	17	0.1897	452202
40048	Video	50	0.1838	452202
40081	Display	58	0.1828	452202
40057	Native	50	0.1634	452202
40023	Mobile	17	0.1572	452202
40039	Mobile	45	0.1469	452202
40035	Native	48	0.1284	452202
<hr/>				

- e. **Query 5 Return the Top3 performance ads uv/pv= Depth of visit, the adsid of the three ads with the highest DV, the ad type and their campaign id, and price.**

SQL

```

1 Select a.AdsID,a.Ads_Type, a.Duration_in_sec, a.PV/a.UV as Depth_of_Visit,
      c.campaignID,c.price
2 From Ads a
3 Natural join campaign c
4 Order by Depth_of_Visit DESC
5 Limit 3;

```

**Output:**

AdsID	Ads_Type	Duration_in_s...	Depth_of_Visit	campaignID	price	
40041	Mobile	29	1.7510	5001	196441	
40041	Mobile	29	1.7510	5002	390724	
40041	Mobile	29	1.7510	5003	424366	

- f. Query 6 Return user name, user account balance, adsid and advertisement type and education level, the condition is that the user clicks, comments and likes, sorted by education\_level

## SQL

```

1 Select ca.Username, ca.balance,a.ads_type,a.adsId,edu_level
2 From customeraccount ca
3 Natural join User u
4 Natural join Ads a
5 Natural join interact_or_present i
6 where i.is_click='Yes' and i.is_like='Yes' and i.is_comment='Yes'
7 order by edu_level;

```

**Output:**

	Username	balance	ads_type	adsId	edu_level	
▶	marnely1i	477488	Display	40004	Bachelor	
◀	aslide28	167113	Display	40066	Bachelor	
◀	lbonifant2o	336109	Native	40056	Bachelor	
◀	cfilede1a	79278	Video	40058	Bachelor	
◀	shansedp	427163	Video	40050	Bachelor	
◀	tgrinov25	39354	Native	40065	College	
◀	agrishukov2l	353470	Display	40004	College	
◀	pcatley2c	113728	Display	40077	College	
◀	hcastilljo1z	193299	Mobile	40039	Doctor	
◀	sprujean1h	315461	Mobile	40018	Doctor	
◀	ctulley1q	352696	Mobile	40040	High-sc...	
◀	cverrallsx	96389	Video	40013	High-sc...	

**g. Query 7 Return accountID, username and the quantity of these account as simple account quantity, the username consisting of abc and d or a username ending with 0-9.**

SQL

```
1 Select CAccountID, username, count(CAccountID) As SpAccount_quantity
2 From CustomerAccount
3 Where `username` REGEXP '[abc]d|[0-9]$'
4 group by CAccountID;
```

Output:

	CAccountID	username	SpAccount_quant...
▶	1	iphateplace0	1
◀	2	awadly1	1
3	eoade2	1	
4	dgovett3	1	
5	cionesco4	1	
6	egrishakov5	1	
7	erisby6	1	
8	lkingscott7	1	
9	rrosgen8	1	
10	iwhittle9	1	
37	hscatchard10	1	
38	wsproson11	1	
39	tseaborne12	1	
...	...	...	...

**h. Query 8 Return User's Uid, Industry, Edu\_Level and FavoriteContent in age 18-25 and country is not in USA**

SQL

```
1 Select uid, Industry,Edu_Level,FavoriteContent
2 From User
3 Where age between 18 and 25
4 and country <> 'UnitedStates'
```

Output:

	uid	Industry	Edu_Level	FavoriteContent
▶	1006	Personal	College	tourism
◀	1008	Management	Doctor	Workplace
	1013	Internet	Master	learning

	uid	Others	Master	Workplace
	1016	Others	Master	Workplace
	1019	Personal	College	sports
	1021	Accounting	College	sports
	1023	Internet	Doctor	tourism
	1024	Banking	Master	learning
	1035	Internet	College	tourism
	1037	Others	Master	learning
	1041	Personal	Bachelor	tourism
	1049	Internet	College	tourism
	1050	Management	Bachelor	Workplace
	1051	Retail	Bachelor	tourism
	1052	Banking	Doctor	Workplace
	1053	Banking	College	tourism
	1054	Retail	Doctor	Workplace
	1056	Banking	Doctor	tourism
	1057	Others	College	sports
	1064	Others	High-sch...	sports
	1070	Others	College	Workplace
	1071	Accounting	Master	sports
	1079	Personal	High-sch...	Workplace
	1084	Banking	College	learning

- i. **Query 9 Return user's uid, age, gender, FavoriteContent, and city, the condition is their age less than average age of all users.**

SQL

```

1 Select uid, age, gender, FavoriteContent, city
2 From User
3 where age <
4 (select avg(age)
5 from user);

```

Output:

	uid	age	gender	FavoriteContent	city
▶	1002	16	Male	tourism	Chattanooga
	1006	24	Male	tourism	Scottsdale
	1008	24	Female	Workplace	Tucson
	1010	30	Female	sports	Huntington Beach
	1012	28	Male	learning	Lansing
	1013	22	Male	learning	Paterson
	1016	19	Female	Workplace	Louisville
	1017	15	Female	tourism	Bethlehem

1019	22	Female	sports	New Brunswick
1021	21	Male	sports	Huntington
1022	30	Male	tourism	Paterson
1023	24	Male	tourism	Reno
1024	21	Female	learning	Syracuse
1027	15	Male	learning	Alhambra
1028	17	Female	learning	San Jose
1030	27	Agender	tourism	Delray Beach
1032	26	Male	Workplace	Chicago
1035	25	Female	tourism	Irvine
1037	18	Female	learning	Minneapolis
1040	15	Female	Workplace	Saint Petersburg

j. **Query 10 Return Platformid, Platformname, DAU, the total number of accounts platform owned.**

SQL

```

1  SELECT p.mainID,p.PlatformName, DAU, count(b.BAccountID)+count(c.CAccountI
D) as TotalNumOfAccounts
2  FROM Platform p
3  Join CustomerAccount c on c.mainID = p.mainID
4  Join BusinessAccount b on b.mainID = p.mainID
5  group by p.mainID
6  order by TotalNumOfAccounts desc;

```

Output:

	mainID	PlatformName	DAU	TotalNumOfAccou...
▶	803	Instagram	813222	84
▶	811	Microsoft	204201	50
▶	806	Twitter	485314	40
▶	807	Amazon	114192	40
▶	816	Quora	618066	40
▶	813	Reddit	75420	36
▶	814	Outbrain	790925	30
▶	812	Tiktok	452202	24
▶	822	Amobee	360771	24
▶	824	Choozie	548272	24

802	Google	87429	20
805	LinkedIn	978700	20
809	Bing	641037	12
818	Yahoo	28555	12
819	Baidu	611420	12
808	Pinterest	378273	8
815	Taboola	592525	8
817	Chegg	675379	8
804	YouTube	546914	6
821	Fyber	849664	6
810	Snapchat	30373	2

## 2. NoSQL Implementation

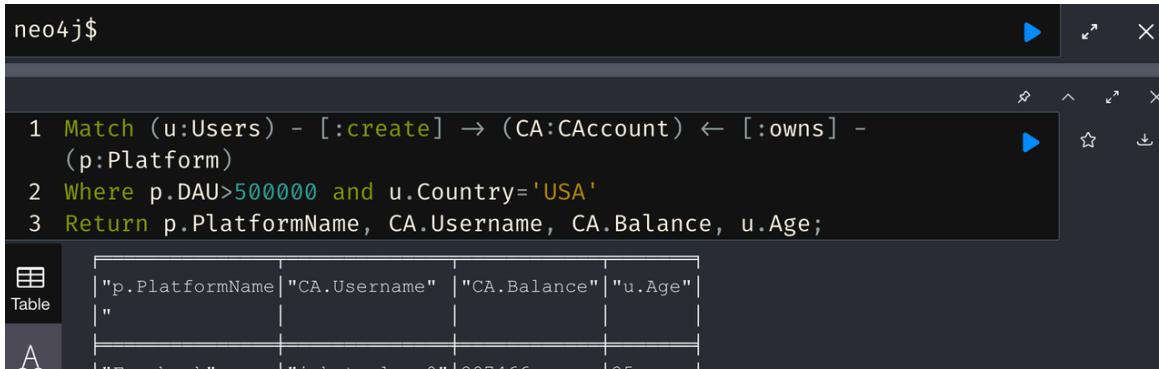
Password:A-Yxgn1Dx14D-gtfcbaCR\_7HqMY-S5ePgQkTeMhDhwY

- a. Query 1 : Return platformname, and their users name, balance, age, the DAU>50w, and user in the USA

Plain Text

```

1 Match (u:Users) - [:create] -> (CA:CAccount) <- [:owns] - (p:Platform)
2 Where p.DAU>500000 and u.Country='USA'
3 Return p.PlatformName, CA.Username, CA.Balance, u.Age;
```



The screenshot shows the Neo4j browser interface. In the top-left, there's a command line input field with "neo4j\$". Below it is a results pane containing a table with three rows of data. The first row has four columns: "p.PlatformName", "CA.Username", "CA.Balance", and "u.Age". The second row is empty. The third row contains the values: "Facebook", "iphateplace0", "287466", and "25". To the left of the results pane, there's a sidebar with a "Table" icon and a letter "A".

p.PlatformName	CA.Username	CA.Balance	u.Age
"			
"Facebook"	"iphateplace0"	287466	25

The screenshot shows a table with columns: Platform, Username, Balance, and Age. The data includes:

Platform	Username	Balance	Age
"Google"	"awadly1"	335507	21
"Instagram"	"eoade2"	271066	26
"TikTok"	"dgovett3"	52599	28
"YouTube"	"cionesco4"	238758	19
"Twitter"	"egrishakov5"	354482	24

MAX COLUMN WIDTH:

- b. Query 2: Return uid, user name, phonenumb. Average age, if the female FavoriteContent is sport and the balance is less than 10w.**

#### Plain Text

```

1 Match (u:Users) - [:create] -> (CA: CAccount)
2 Where u.FavoriteContent in ['sport','workplace'] and CA.Balance < 10000000
3 Return u.Uid, CA.Username, u.Age;

```

The screenshot shows a query result with columns: u.Uid, CA.Username, and u.Age. The data includes:

u.Uid	CA.Username	u.Age
1006	"egrishakov5"	24
1008	"lkingscott7"	33
1009	"rrrosgen8"	28
1010	"iwhittle9"	18

### c. Query 3: Return User's Uid,Username, Industry, degree\_Level and FavoriteContent in age 18-25 and country is not in USA

#### Plain Text

```

1 Match (u:Users) - [:create] -> (CA:CAccount)
2 Where 18 < u.Age < 30 and u.Country <> 'USA'
3 Return u.Uid, CA.Username, u.Country, u.FavoriteContent

```

The screenshot shows the neo4j browser interface. The query entered is:

```

1 Match (u:Users) - [:create] -> (CA:CAccount)
2 Where 18 < u.Age < 30 and u.Country <> 'USA'
3 Return u.Uid, CA.Username, u.Country, u.FavoriteContent

```

The results are displayed in a table:

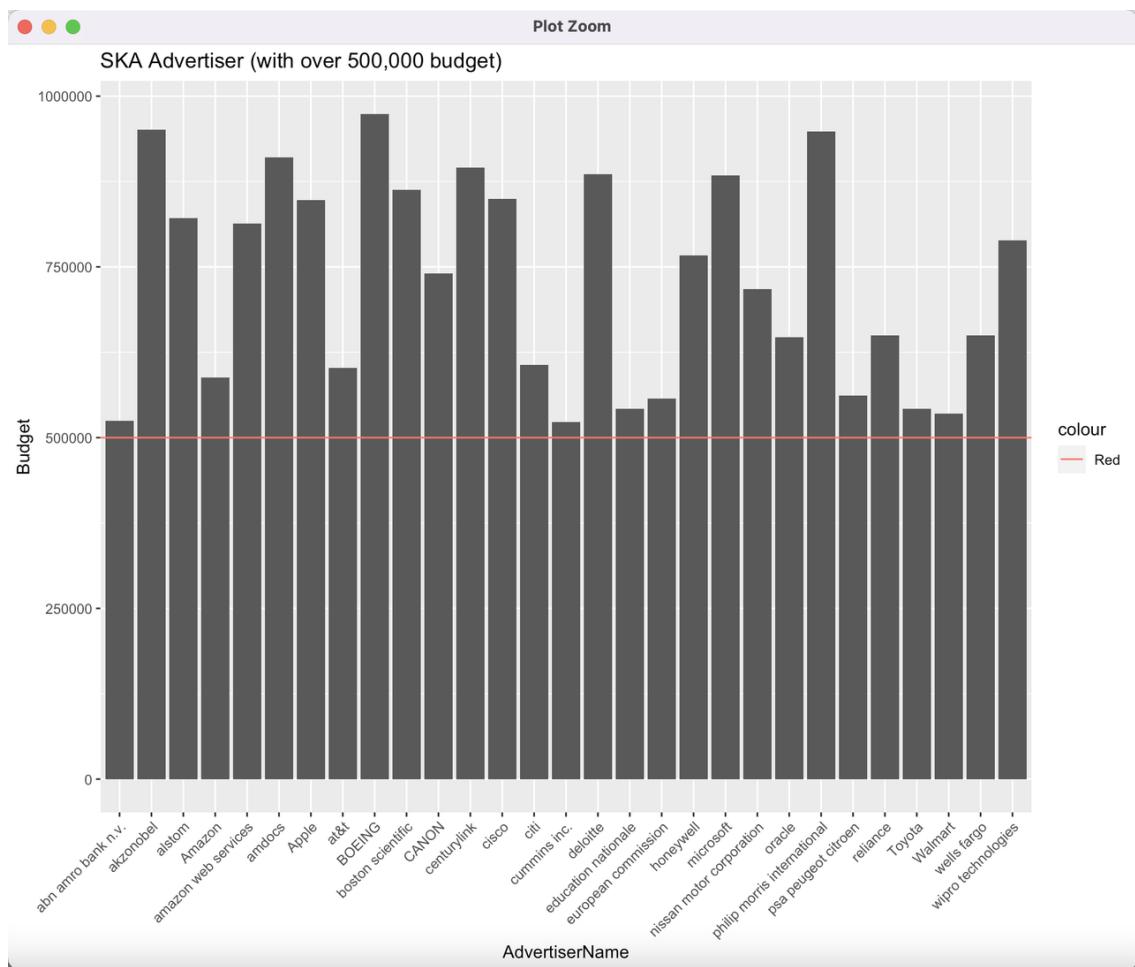
	u.Uid	CA.Username	u.Country	u.FavoriteContent
1	1007	"erisby6"	"France"	"learning"
2	1009	"rosogen8"	"China"	"workplace"

At the bottom, it says "Started streaming 2 records after 2 ms and completed after 4 ms."

## V. Database Access via R

### 1. Demand Side

We mainly look at SKA(SuperKeyAccount) Advertisers and their budget distribution in order to help platform and agency to target these clients.

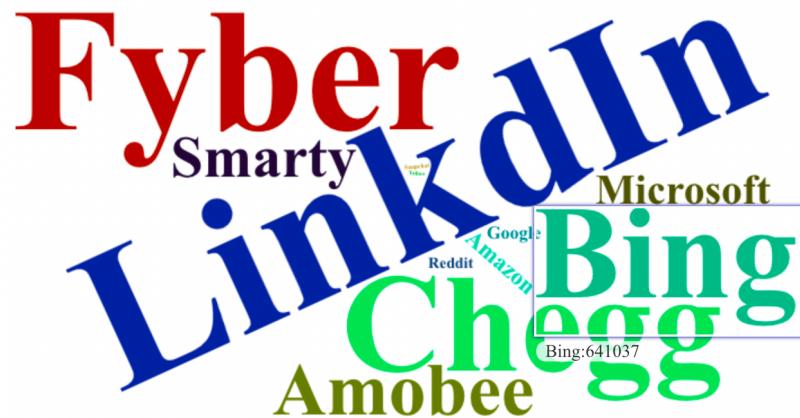


## 2. Monetization Platform Side

How many account platform owned.



What is the DAU quantity of Platform?



### 3. Target User Side

Find the correlation between user's age and their account balance.

