Operation Analytics and Investigating Metric Spike

Advance SQL

Project description

The project is about Operation Analytics and Investigating Metric Spike. As a Data Analyst Lead for a company like Microsoft, I have been provided with different data sets and tables, which I need to analyze to derive insights and answer the questions asked by different departments.

In case study 1 there is job_data table while in case study 2 there are tables. Operation Analytics and Investigating Metric Spike.

In case study 1 the insights are found based on following questions:

1. Number of jobs reviewed:

2. Throughput: Calculate 7 day rolling average of throughput? For throughput, do you prefer daily metric or 7-day rolling and why?

3. Percentage share of each language:

4. Duplicate rows: Rows that have the same value present in them.

In case study 2 the insights are found based on following questions:

1. User Engagement: To measure the activeness of a user. Measuring if the user finds quality in a product/service.

2. User Growth: Amount of users growing over time for a product.

3. Weekly Retention: Users getting retained weekly after signing

4. Weekly Engagement: To measure the activeness of a user. Measuring if the user finds quality in a product/service weekly.

5. Email Engagement: Users engaging with the email service.

Approach

To start with the project, I first spent some time understanding the table given and the questions that need to be answered. I then created a database and tables using the structure provided. Using SQL, I performed the analysis and answered the questions asked above.

Tech-Stack Used

I used MySQL workbench to perform analysis for the project. I used Ms Excel to convert data set into CSV file and then export into MySQL workbench.

Insights

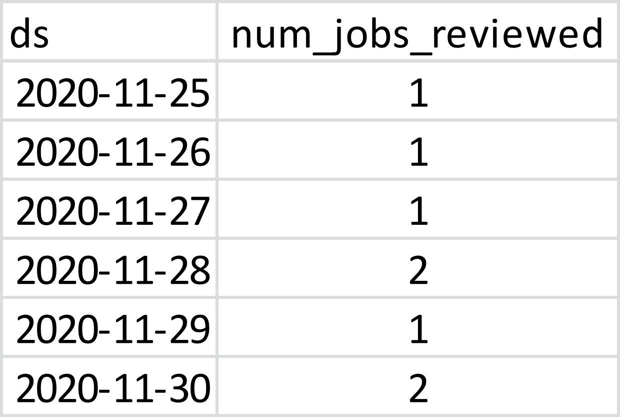
Case study 1 (JOB DATA)

1. Calculate the number of jobs reviewed per hour per day for November 2020:

Query:

```
SELECT ds, COUNT(job_id) AS num_jobs_reviewed
 FROM job_data
 WHERE ds BETWEEN '2020-11-01' AND '2020-11-30‘
 GROUP BY ds
 ORDER BY ds;
```

OUTPUT:



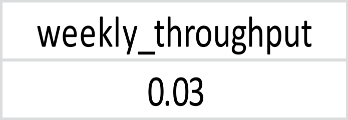
ds	num_jobs_reviewed
2020-11-25	1
2020-11-26	1
2020-11-27	1
2020-11-28	2
2020-11-29	1
2020-11-30	2

2. Let’s say the above metric is called throughput. Calculate 7 day rolling average of throughput? For throughput, do you prefer daily metric or 7-day rolling and why?

Query :

```
SELECT ROUND(COUNT(event)/sum(time_spent),2) as
 weekly_throughput
 FROM job_data;
```

Output:

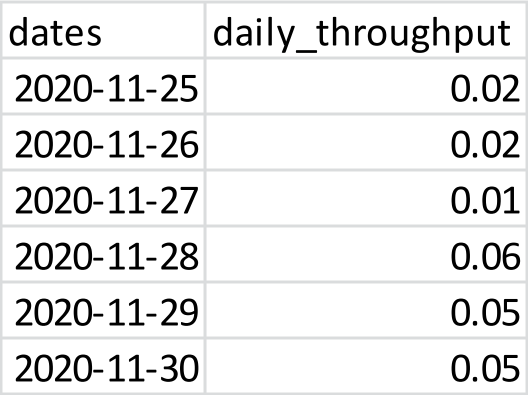


weekly_throughput
0.03

Query 2:

```
Select ds as dates, round(count(event)/sum(time_spent),2) as
daily_throughput
 From job_data
Group by ds
Order by ds;
```

Output:



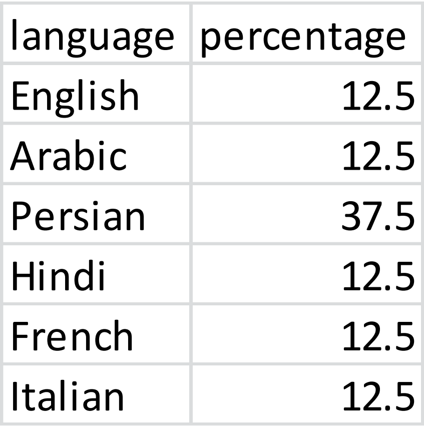
dates	daily_throughput
2020-11-25	0.02
2020-11-26	0.02
2020-11-27	0.01
2020-11-28	0.06
2020-11-29	0.05
2020-11-30	0.05

3. Calculate the percentage share of each language in the last 30 days?

Query:

```
Select language, count(*)*100.0/sum(count(*)) over() as percentage
From job_data
Group by language;
```

Output:



language	percentage
English	12.5
Arabic	12.5
Persian	37.5
Hindi	12.5
French	12.5
Italian	12.5

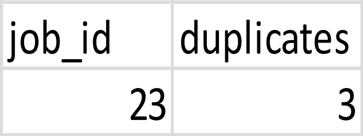
4. Let’s say you see some duplicate rows in the data. How will you display duplicates from the table?

Lets take example of job_id

Query:

```
SELECT  job_id, count(*) as duplicates
FROM job_data
GROUP BY job_id
HAVING count(*)>1;
```

OUTPUT:



job_id	duplicates
23	3

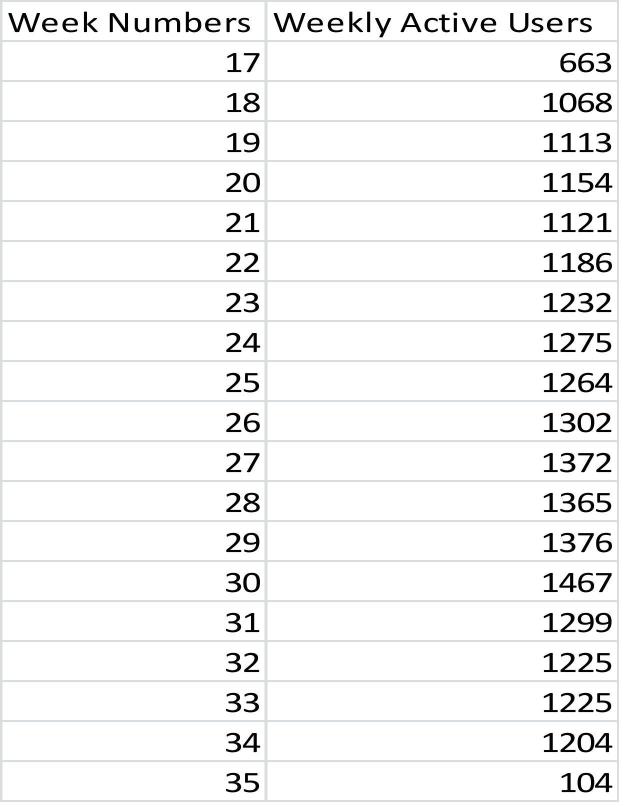
CASE STUDY 2 (Investigating metric spike)

A. Calculate the weekly user engagement?

Query:

```
Select EXTRACT(WEEK FROM occurred_at) AS “Week Numbers”, COUNT(
DISTINCT user_id) AS “weekly Active Users”
FROM events
WHERE events_type=‘engagement’
GROUP BY 1;
```

Output:



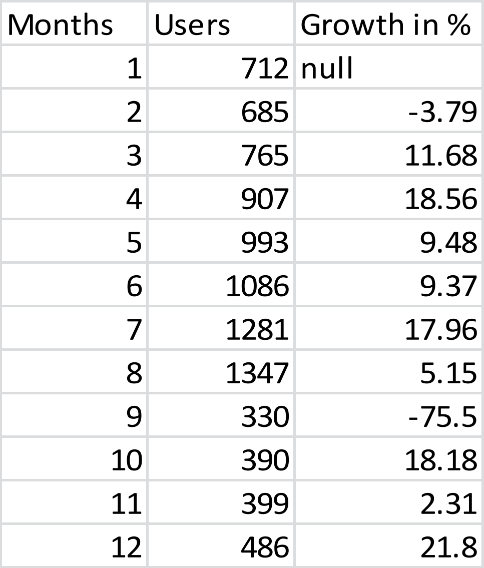
Week Numbers	Weekly Active Users
17	663
18	1068
19	1113
20	1154
21	1121
22	1186
23	1232
24	1275
25	1264
26	1302
27	1372
28	1365
29	1376
30	1467
31	1299
32	1225
33	1225
34	1204
35	104

2. Calculate the user growth for product?

Query:

```
Selcet Months. Users, Round(((Users/LAG(Users, 1) OVER (ORDER BY
Months) -1)*100),2) as “Growth in %”
From (
 Select EXTRACT(MONTH From  created_at) as Months,
 COUNT(activated_at) AS Users
 From users
 WHERE activated_at NOT IN(“”)
 GROUP BY 1
 ORDER BY 1
  ) sub;
```

Output:



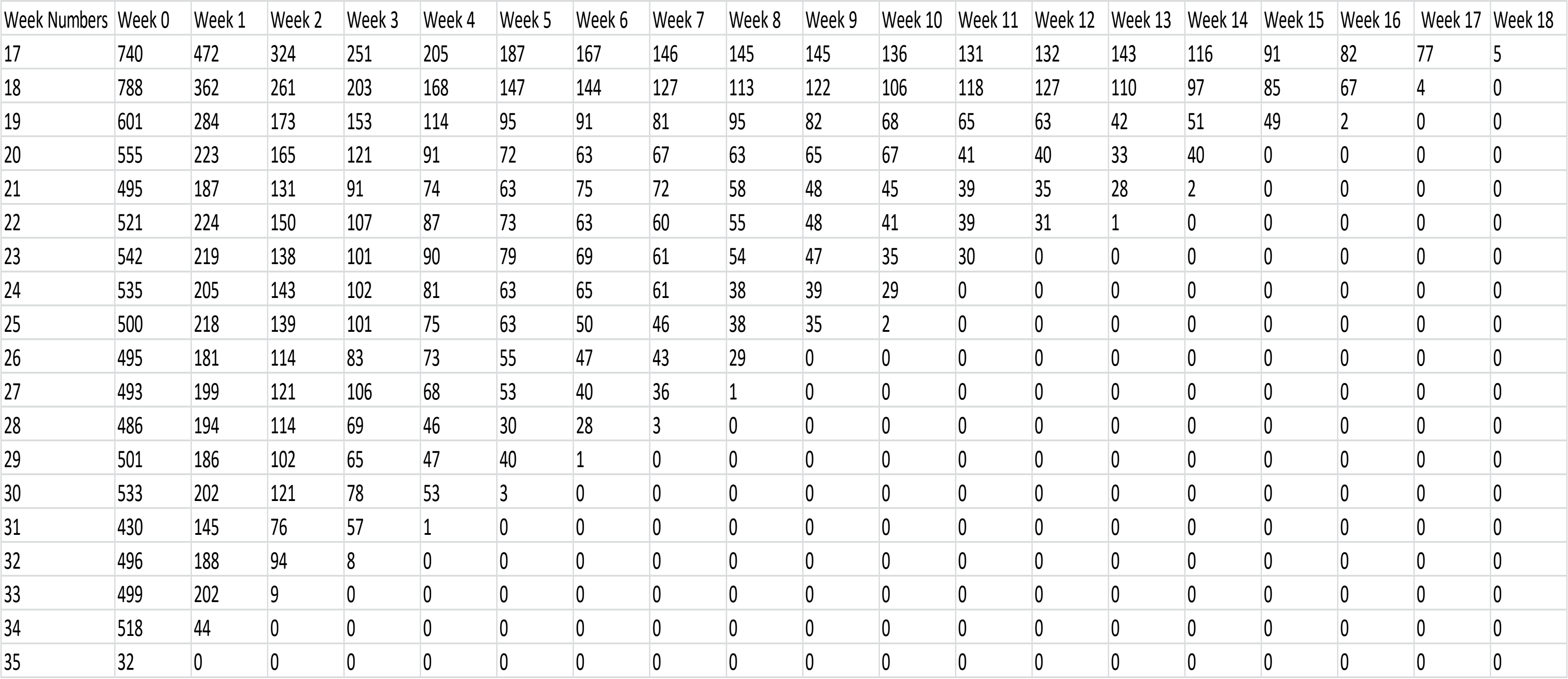
Months	Users	Growth in %
1	712	null
2	685	-3.79
3	765	11.68
4	907	18.56
5	993	9.48
6	1086	9.37
7	1281	17.96
8	1347	5.15
9	330	-75.5
10	390	18.18
11	399	2.31
12	486	21.8

3.Calculate the weekly retention of users-sign up cohort?

Query:

```
 SELECT first AS “Week Numbers”
SUM(CASE WHEN week_number = 0 THEN 1 ELSE 0 END) AS “Week 0”,
SUM(CASE WHEN week_number =  1 THEN 1 ELSE 0 END) AS “Week 1”,
SUM(CASE WHEN week_number = 2 THEN 1 ELSE 0 END) AS “Week 2”,
SUM(CASE WHEN week_number = 3 THEN 1 ELSE 0 END) AS “Week 3”,
SUM(CASE WHEN week_number = 4 THEN 1 ELSE 0 END) AS “Week 4”,
SUM(CASE WHEN week_number = 5 THEN 1 ELSE 0 END) AS “Week 5”,
 SUM(CASE WHEN week_number = 6 THEN 1 ELSE 0 END) AS “Week 6”,
 SUM(CASE WHEN week_number = 7 THEN 1 ELSE 0 END) AS “Week 7”,
SUM(CASE WHEN week_number = 8 THEN 1 ELSE 0 END) AS “Week 8”,
 SUM(CASE WHEN week_number = 9 THEN 1 ELSE 0 END) AS “Week 9”,
 SUM(CASE WHEN week_number = 10 THEN 1 ELSE 0 END) AS “Week 10”,
SUM(CASE WHEN week_number = 11 THEN 1 ELSE 0 END) AS “Week 11”,
SUM(CASE WHEN week_number = 12 THEN 1 ELSE 0 END) AS “Week 12”,
SUM(CASE WHEN week_number = 13 THEN 1 ELSE 0 END) AS “Week 13”,
SUM(CASE WHEN week_number = 14 THEN 1 ELSE 0 END) AS “Week 14”,
SUM(CASE WHEN week_number = 15 THEN 1 ELSE 0 END) AS “Week 15”,
SUM(CASE WHEN week_number = 16 THEN 1 ELSE 0 END) AS “Week 16”,
SUM(CASE WHEN week_number = 17 THEN 1 ELSE 0 END) AS “Week 17”,
SUM(CASE WHEN week_number = 18 THEN 1 ELSE 0 END) AS “Week 18”,
FROM (
SELECT m.user_id, m.login_week, n.first, m.login_week – first AS week_number
FROM
 (SELECT  user_id, EXTRACT(WEEK FROM occurred_at) AS login_week FROM events
GROUP BY 1, 2) m,
 (SELECT  user_id,  MIN(EXTRACT(WEEK FROM occurred_at)) AS first FROM events
GROUP BY 1) n
WHERE m.user_id = n.user_id
) sub
GROUP BY first
ORDER BY first;
```

Output:

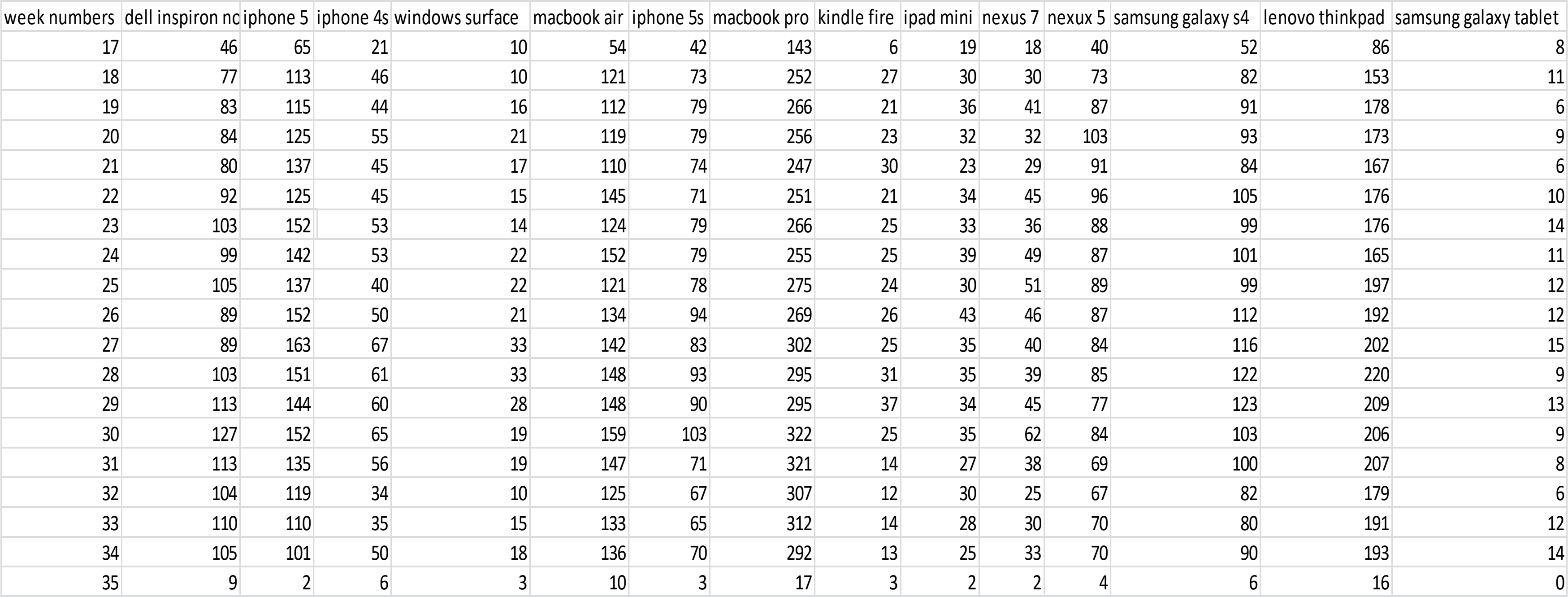


Week Numbers	Week 0	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14	Week 15	Week 16	Week 17	Week 18
17	740	472	324	251	205	187	167	146	145	145	136	131	132	143	116	91	82	77	5
18	788	362	261	203	168	147	144	127	113	122	106	118	127	110	97	85	67	4	0
19	601	284	173	153	114	95	91	81	95	82	68	65	63	42	51	49	2	0	0
20	555	223	165	121	91	72	63	67	63	65	67	41	40	33	40	0	0	0	0
21	495	187	131	91	74	63	75	72	58	48	45	39	35	28	2	0	0	0	0
22	521	224	150	107	87	73	63	60	55	48	41	39	31	1	0	0	0	0	0
23	542	219	138	101	90	79	69	61	54	47	35	30	0	0	0	0	0	0	0
24	535	205	143	102	81	63	65	61	38	39	29	0	0	0	0	0	0	0	0
25	500	218	139	101	75	63	50	46	38	35	2	0	0	0	0	0	0	0	0
26	495	181	114	83	73	55	47	43	29	0	0	0	0	0	0	0	0	0	0
27	493	199	121	106	68	53	40	36	1	0	0	0	0	0	0	0	0	0	0
28	486	194	114	69	46	30	28	3	0	0	0	0	0	0	0	0	0	0	0
29	501	186	102	65	47	40	1	0	0	0	0	0	0	0	0	0	0	0	0
30	533	202	121	78	53	3	0	0	0	0	0	0	0	0	0	0	0	0	0
31	430	145	76	57	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	496	188	94	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	499	202	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	518	44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
35	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

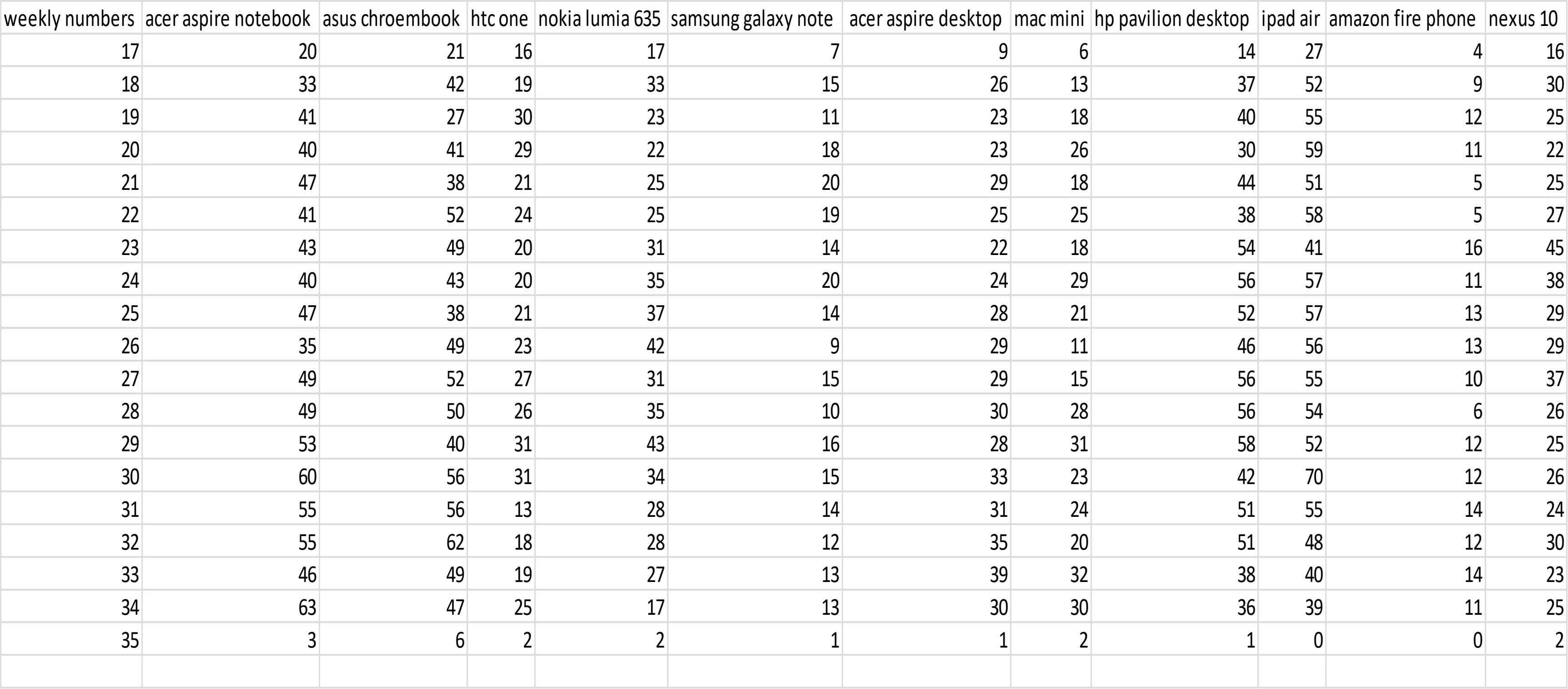
4.Calculate the weekly engagement per device?
Query:

```
Select EXTRACT(WEEK FROM occurred_at) AS “week Numbers”,
COUNT(DISTINCT CASE WHEN device IN(‘dell inspiron notebook’) THEN user_id ELSE
NULL END) AS “Dell Inspiron Notebook”,
COUNT(DISTINCT CASE WHEN device IN(‘iphone 5’) THEN user_id ELSE NULL END) AS
“iphone 5”,
COUNT(DISTINCT CASE WHEN device IN(‘iphone 4s’) THEN user_id ELSE NULL END) AS
“iphone 4s”,
COUNT(DISTINCT CASE WHEN device IN(‘windows surface’) THEN user_id ELSE NULL
END) AS “windows surface”,
COUNT(DISTINCT CASE WHEN device IN(‘macbook air’) THEN user_id ELSE NULL END)
AS “macbook air”,
COUNT(DISTINCT CASE WHEN device IN(‘iphone 5s’) THEN user_id ELSE NULL END) AS
“iphone 5s”,
COUNT(DISTINCT CASE WHEN device IN(‘macbook pro’) THEN user_id ELSE NULL
END) AS “macbook pro”,
COUNT(DISTINCT CASE WHEN device IN(‘kindle fire’) THEN user_id ELSE NULL END)
AS “kindle fire”,
COUNT(DISTINCT CASE WHEN device IN(‘ipad mini’) THEN user_id ELSE NULL END) AS “ipad mini”,
COUNT(DISTINCT CASE WHEN device IN(‘nexus 5’) THEN user_id ELSE NULL END) AS “nexus 5”,
COUNT(DISTINCT CASE WHEN device IN(‘samsung galaxy s4’) THEN user_id ELSE NULL END) AS “samsung galaxy s4”,
COUNT(DISTINCT CASE WHEN device IN(‘lenovo thinkpad’) THEN user_id ELSE NULL END) AS “lenovo thinkpad”,
COUNT(DISTINCT CASE WHEN device IN(‘samsung galaxy tablet’) THEN user_id ELSE NULL END) AS “samsung galaxy tablet”,
COUNT(DISTINCT CASE WHEN device IN(‘acer aspire notebook’) THEN user_id ELSE NULL END) AS “acer aspire notebook”,
COUNT(DISTINCT CASE WHEN device IN(‘asus chromebook’) THEN user_id ELSE NULL END) AS “asus chromebook”,
COUNT(DISTINCT CASE WHEN device IN(‘htc one’) THEN user_id ELSE NULL END) AS “htc one”,
COUNT(DISTINCT CASE WHEN device IN(‘nokia lumia 635’) THEN user_id ELSE NULL END) AS “nokia lumia 635”,
COUNT(DISTINCT CASE WHEN device IN(‘samsung galaxy note’) THEN user_id ELSE NULL END) AS “samsung galaxy note”,
COUNT(DISTINCT CASE WHEN device IN(‘acer aspire desktop’) THEN user_id ELSE NULL
END) AS “acer aspire desktop”,
COUNT(DISTINCT CASE WHEN device IN(‘mac mini’) THEN user_id ELSE NULL END) AS
“mac mini”,
COUNT(DISTINCT CASE WHEN device IN(‘hp pavilion desktop’) THEN user_id ELSE NULL
END) AS “hp pavilion desktop”,
COUNT(DISTINCT CASE WHEN device IN(‘dell inspiron desktop’) THEN user_id ELSE
NULL END) AS “dell inspiron desktop”,
COUNT(DISTINCT CASE WHEN device IN(‘ipad air’) THEN user_id ELSE NULL END) AS
“ipad air”,
COUNT(DISTINCT CASE WHEN device IN(‘amazon fire phone’) THEN user_id ELSE NULL
END) AS “amazon fire phone”,
COUNT(DISTINCT CASE WHEN device IN(‘nexus 10’) THEN user_id ELSE NULL END) AS
“nexus 10”,
From events
WHERE event_type=‘engagement’
GROUP BY 1
ORDER BY 1;
```

Output:



week numbers	dell inspiron nc	iphone 5	iphone 4s	windows surface	macbook air	iphone 5s	macbook pro	kindle fire	ipad mini	nexus 7	nexux 5	samsung galaxy s4	lenovo thinkpad	samsung galaxy tablet
17	46	65	21	10	54	42	143	6	19	18	40	52	86	8
18	77	113	46	10	121	73	252	27	30	30	73	82	153	11
19	83	115	44	16	112	79	266	21	36	41	87	91	178	6
20	84	125	55	21	119	79	256	23	32	32	103	93	173	9
21	80	137	45	17	110	74	247	30	23	29	91	84	167	6
22	92	125	45	15	145	71	251	21	34	45	96	105	176	10
23	103	152	53	14	124	79	266	25	33	36	88	99	176	14
24	99	142	53	22	152	79	255	25	39	49	87	101	165	11
25	105	137	40	22	121	78	275	24	30	51	89	99	197	12
26	89	152	50	21	134	94	269	26	43	46	87	112	192	12
27	89	163	67	33	142	83	302	25	35	40	84	116	202	15
28	103	151	61	33	148	93	295	31	35	39	85	122	220	9
29	113	144	60	28	148	90	295	37	34	45	77	123	209	13
30	127	152	65	19	159	103	322	25	35	62	84	103	206	9
31	113	135	56	19	147	71	321	14	27	38	69	100	207	8
32	104	119	34	10	125	67	307	12	30	25	67	82	179	6
33	110	110	35	15	133	65	312	14	28	30	70	80	191	12
34	105	101	50	18	136	70	292	13	25	33	70	90	193	14
35	9	2	6	3	10	3	17	3	2	2	4	6	16	0



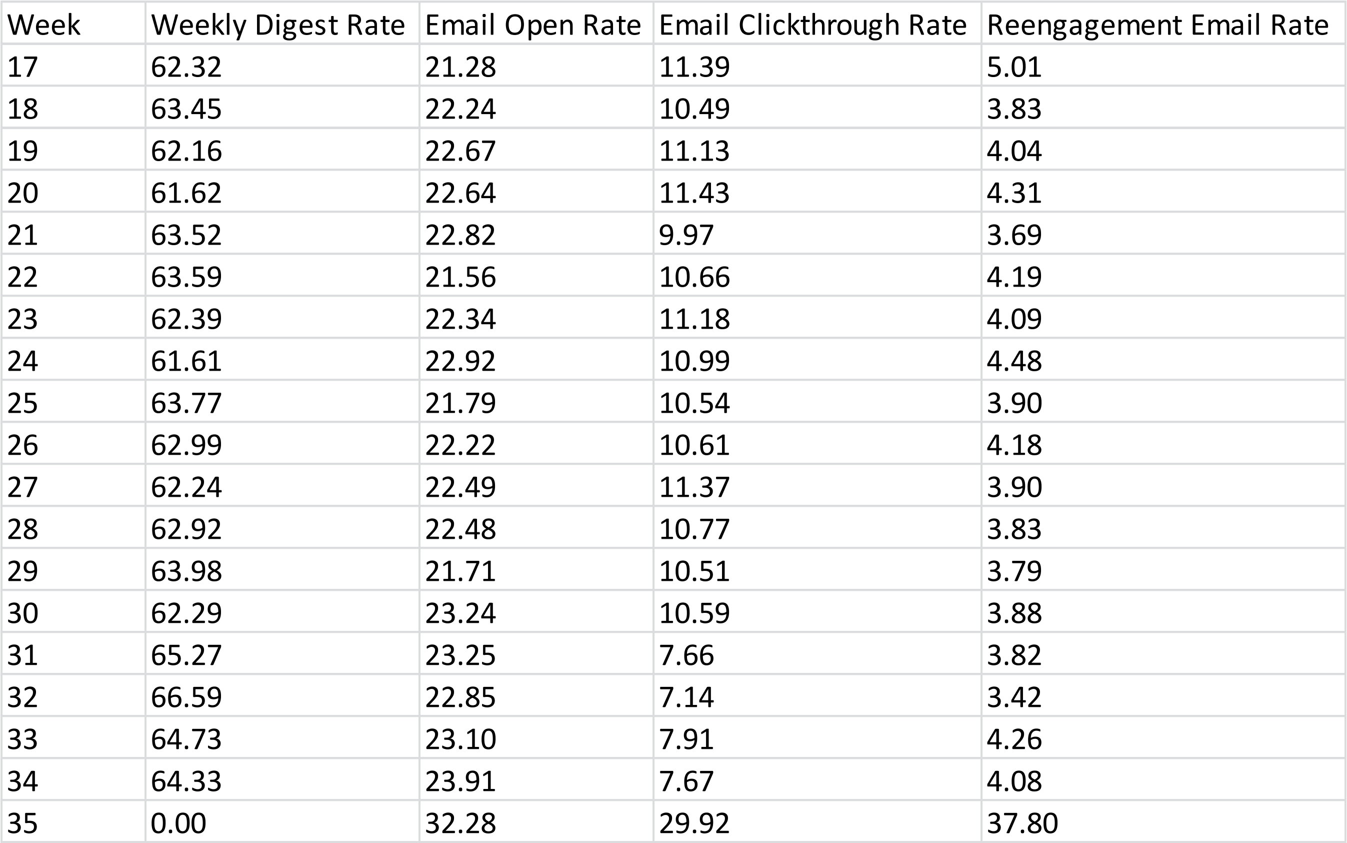
weekly numbers	acer aspire notebook	asus chroembook	htc one	nokia lumia 635	samsung galaxy note	acer aspire desktop	mac mini	hp pavilion desktop	ipad air	amazon fire phone	nexus 10
17	20	21	16	17	7	9	6	14	27	4	16
18	33	42	19	33	15	26	13	37	52	9	30
19	41	27	30	23	11	23	18	40	55	12	25
20	40	41	29	22	18	23	26	30	59	11	22
21	47	38	21	25	20	29	18	44	51	5	25
22	41	52	24	25	19	25	25	38	58	5	27
23	43	49	20	31	14	22	18	54	41	16	45
24	40	43	20	35	20	24	29	56	57	11	38
25	47	38	21	37	14	28	21	52	57	13	29
26	35	49	23	42	9	29	11	46	56	13	29
27	49	52	27	31	15	29	15	56	55	10	37
28	49	50	26	35	10	30	28	56	54	6	26
29	53	40	31	43	16	28	31	58	52	12	25
30	60	56	31	34	15	33	23	42	70	12	26
31	55	56	13	28	14	31	24	51	55	14	24
32	55	62	18	28	12	35	20	51	48	12	30
33	46	49	19	27	13	39	32	38	40	14	23
34	63	47	25	17	13	30	30	36	39	11	25
35	3	6	2	2	1	1	2	1	0	0	2

5.Calculate the email engagement metrics?

Query:

```
Select Week,
ROUND(weekly_digest/total*100),2) as “Weekly Digest Rate”,
ROUND((email_opens/total*100),2) as “Email Open Rate”,
ROUND((email_clickthroughs/total*100),2) as Email Clickthrough Rate”,
ROUND((reengagement_emails/total*100),2) as Reengagement Email Rate”
From
(
SELECT EXTRACT(WEEK FROM occurred_at) as Week,
COUNT(CASE WHEN action= ‘sent_weekly_digest’ THEN user_id ELSE NULL END) AS
weekly_digest,
COUNT(CASE WHEN action= ‘email_open’ THEN user_id ELSE NULL END) AS
email_opens,
COUNT(CASE WHEN action= ‘email_clickthrough’ THEN user_id ELSE NULL END) AS
email_clickthrough,
COUNT(CASE WHEN action= ‘sent_reengagement_email’ THEN user_id ELSE NULL
END) AS reengagement_emails,
COUNT(user_id) as total
FROM email_events
GROUP BY 1
)sub
GROUP BY 1
ORDER BY 1;
```

Output:



Week	Weekly Digest Rate	Email Open Rate	Email Clickthrough Rate	Reengagement Email Rate
17	62.32	21.28	11.39	5.01
18	63.45	22.24	10.49	3.83
19	62.16	22.67	11.13	4.04
20	61.62	22.64	11.43	4.31
21	63.52	22.82	9.97	3.69
22	63.59	21.56	10.66	4.19
23	62.39	22.34	11.18	4.09
24	61.61	22.92	10.99	4.48
25	63.77	21.79	10.54	3.90
26	62.99	22.22	10.61	4.18
27	62.24	22.49	11.37	3.90
28	62.92	22.48	10.77	3.83
29	63.98	21.71	10.51	3.79
30	62.29	23.24	10.59	3.88
31	65.27	23.25	7.66	3.82
32	66.59	22.85	7.14	3.42
33	64.73	23.10	7.91	4.26
34	64.33	23.91	7.67	4.08
35	0.00	32.28	29.92	37.80

Insights

Through this project, insights will be gained into the company's operations by analyzing the data sets and tables provided. In case study 1, we will gain insight into the number of jobs reviewed per hour per day, throughput, percentage share of each language, and how to display duplicates from the table. In case study 2, we will gain insight into weekly user engagement, user growth, weekly retention of users-sign up cohort, weekly engagement per device, and email engagement metrics.

Result

The result of this project will be a report containing all the insights gained from analyzing the data sets and tables provided. The report will provide answers to the questions asked in the case studies and will be useful in improving the overall performance of the company.