Comparison between JavaScript and Typescript and their essence in the world of computers

Junyi Li

Harrisburg University of Science and Technology

December 2021

Abstract

This paper discusses how JavaScript and TypeScript are used in projects and their effect on the application runtime and efficiency. TypeScript will add type robust type system to JavaScript. Does it worth it and are they good for large scale project? These are some pros and cons about Typescript and JavaScript. I collect user data from Stack overflow and use SQL to analyze them to see TypeScript and JavaScript influence on current programming world.

Keywords: TypeScript, JavaScript, SQL, React

1. Introduction

JavaScript is a scripting or programming language which helps developers to implement features on web pages. You can use it to create dynamically updating content, control multimedia, animate images, and other visualization. [1] JavaScript has no types by default. Since it is easy to use, it is widely used to create kinds of web applications. However, some people think it is difficult to use JavaScript to create complex codebases due to its inherent type-unsafe feature, and the type error can only be caught in runtime when the code is executed. Below are examples of type errors to prevent in the early stage in TypeScript vs JavaScript:

For JavaScript code, we define an add function and pass in Boolean and number as parameters, and it will return 3 and show no error:

[1] See the defin https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript ition of JavaScript at ADN Web Docs

```
const add=(a,b)=>{
  return a+b
}
const c=add(1,2)
console.log(c) // return 3
```

For TypeScript code, we also define an add function and pass in Boolean and number as parameters and it will show error immediately and decrease debugging time, because I have set two parameters as number as requirement:

```
function addT (a: number, b: number){
     return a+b
  }
   addT(true,2) // show error at Boolean word
```

Therefore, some people choose to use TypeScript, which was proposed by Swamy et al.(2014). As he said in the article, TypeScript is a strict superset of JavaScript. It adds robust type system to JavaScript to support a tighter integration with code. It can catch errors at the early stage, and it defines the data type requirements and will throw error when there is inconsistency.

According to Ben Awad[i]"TypeScript is long term investment'. It takes time to learn and take times to add code and type, but you can get some nice reward from it." Another good thing about TypeScript is that autocompletion and auto import. When you write a new file, you can finish code first and then enter ctrl + period, and then the file will automatically import all the module on the top of the code. When we work on React framework, it can also autocomplete the import of reusable component for us.

Also, you can change from TypeScript to JavaScript and import Babel to do that in React. As for the market, many companies use TypeScript. According to article[2],   There is about 60% of JS programmers use TypeScript and 22% more people may change from JavaScript to TypeScript.

[2] See the data of TypeScript popularity from Article : Why You Should Choose TypeScript Over JavaScript, written by
Gints Dreimanis, Olga Bolgurtseva （2020）

However, there are some advantages of using JavaScript over TypeScript.

The first is that TypeScript takes more time. You need to take some time to write these types. Companies may push employees to finish projects quickly. The people with JavaScript may already finish the project while the people with TypeScript still work on typing problems. A lot of developers think it is not necessary to use Type in development. The second is that the error message can sometimes be cryptic and hard to get where is the wrong. The next thing is library support. When you use some library, some packages don't have type. You must add type by yourselves. Also, some browsers don't know how to run Typescript Code. We need to use a compiler to transform TypeScript code to JavaScript code. When I use React, there is a tsconfig. Json file, which is customized TS complier, and it can convert the TS to JS code. This process takes some time.

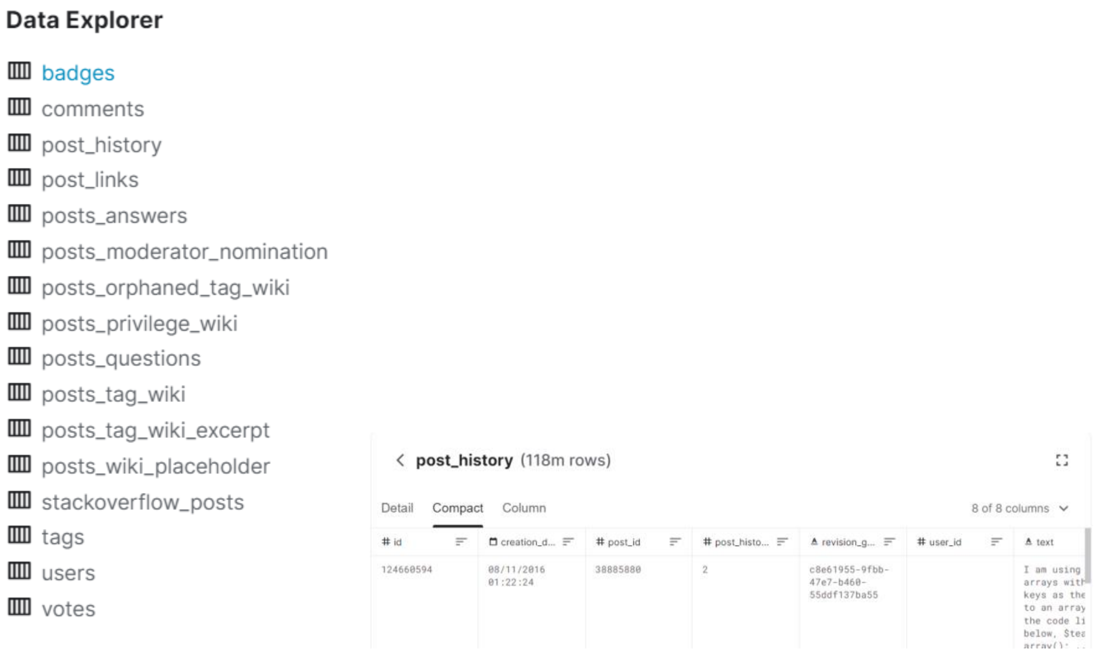
Nowadays, many people start with JavaScript and work on it, because it is much easier and saves time to write products. TypeScript and JavaScript can run same development environments, such as, react and Node.js. When people write type, some people will set "Any" to the type, which makes no difference to JavaScript. According to report, there is only 15% bugs prevented by TypeScript. According to Abnb, there is only 30% bugs found using TypeScript. There is trading off between JavaScript and TypeScript. I will try these technologies deeply and give feedback.

2. Data Analysis

I got Stack Overflow Data from Kaggle website[3]. The dataset is closely related to TypeScript using situation. This dataset also includes all other programming languages, such as JavaScript, Java, and Python. It talks about tools and problem current developers have for the recent years. What is Stack Overflow? as developers, Stack Overflow is the largest online community for us to learn, share our knowledge, and post questions. Therefore, Stack Overflow data is valuable for me to know the essence of JavaScript and TypeScript in the current world. As for analysis, I use the Kaggle platform and use Python as language. Kaggle platform is an online community where people use Python and SQL to do data analysis. It is the best place for me to do research.

[3] Kaggle data https://www.kaggle.com/stackoverflow/stackoverflow

This is BigQuery dataset and is updated on a quarterly basis. The dataset includes an archive of Stack Overflow content, including posts, votes, tags and badges. “This dataset is updated to mirror the Stack Overflow content on the Internet Archive, and is also available through the Stack Exchange Data Explorer.”[4] Below picture is an image of a dataset. It has different tables.



Data Explorer

- badges
- comments
- post_history
- post_links
- posts_answers
- posts_moderator_nomination
- posts_orphaned_tag_wiki
- posts_privilege_wiki
- posts_questions
- posts_tag_wiki
- posts_tag_wiki_excerpt
- posts_wiki_placeholder
- stackoverflow_posts
- tags
- users
- votes

post_history (118m rows)

# id	creation_d...	# post_id	# post_histo...	revision_g...	# user_id	text
124660594	08/11/2016 01:22:24	38885880	2	c8e61955-9fbb-47e7-b460-55ddf137ba55		I am using arrays wit keys as the to an array the code li below, $tea array(): ..

Each table has its schema. “A schema is a collection of database objects like tables, triggers, stored procedures, etc. A schema relates to a user which is known as the schema owner.” [5] For example, in

Table Tags, the schema is

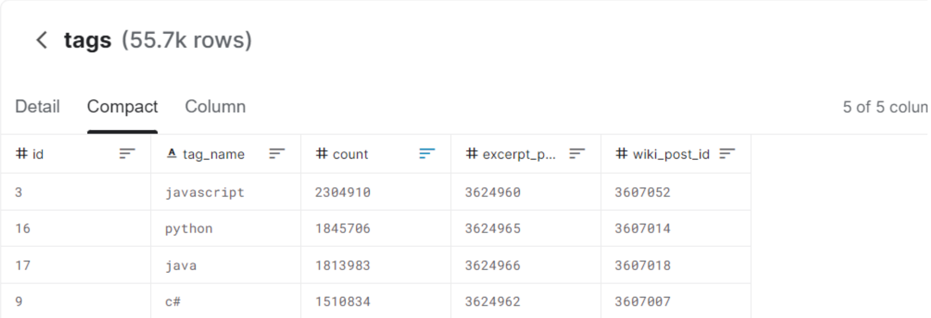
```
[SchemaField('id', 'INTEGER', 'NULLABLE', None, (), None),
 SchemaField('tag_name', 'STRING', 'NULLABLE', None, (), None),
 SchemaField('count', 'INTEGER', 'NULLABLE', None, (), None),
 SchemaField('excerpt_post_id', 'INTEGER', 'NULLABLE', None, (), None),
 SchemaField('wiki_post_id', 'INTEGER', 'NULLABLE', None, (), None)]
```

For each tag, it has id, tag_name, count, excerpt_post_id and wiki_post_id. If we need a certain tag id with its tag name, we can make a SQL query.

Below is the tag table content and I have descended the count column value.

[4] https://www.kaggle.com/stackoverflow/stackoverflow

[5] CREATE SCHEMA in SQL Server , https://www.geeksforgeeks.org/create-schema-in-sql-server/



# id	tag_name	# count	# excerpt_p...	# wiki_post_id
3	javascript	2304910	3624960	3607052
16	python	1845706	3624965	3607014
17	java	1813983	3624966	3607018
9	c#	1510834	3624962	3607007

As we can see from this picture, JavaScript ranks first. Python ranks second place. TypeScript ranks 45 places. It demonstrates that people are more likely to use JavaScript. I used TypeScript before. It is complicated and especially difficult when you use it with other modules. You must set type for many variables and modules.

TypeScript is difficult, but many big companies use TypeScript. Especially many good programmers recommend people to use TypeScript. The companies that use TypeScript include 10X Genomics, Cox Automotive, CloudFlare, Revel IT, BlackLine, and Smartsheet. According to the website HG insight[6], there are 5,606 companies in CA and 2,601 companies in NY using TypeScript.

For JavaScript, there are 113,714 in CA using it, and 54,390 in NY using it. So, a student more easily finds a job if he or she uses JavaScript. Having an experience of using TypeScript will give HR a good impression

Later, I will use SQL to do deep analysis for the use of JavaScript and TypeScript.

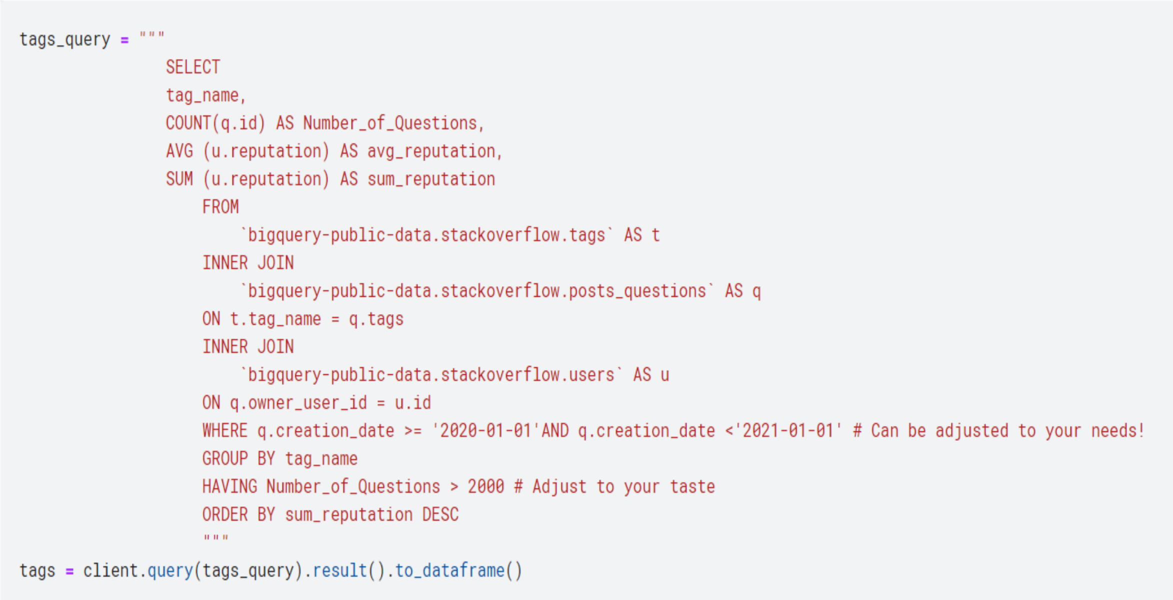
(1) Ranking the programming languages according to user reputation.

First, I will check the period from the year 2020-01-01 to year 2020-12-31 and analysis the tags. Tags means technology names. The measurement is based on the concept that what kind of people would use TypeScript. Junior people? Experience people? Expertise? I will check the reputation of people who use TypeScript frequently or JavaScript frequently.

I use the Reputation as my measurement to know the people' s level. As for the definition of reputation, if we answer more questions on Stack Overflow, we will get a higher reputation, which indicates the programming level. The higher the reputation, the higher the person programming level is." Reputation is a rough measurement of how much the community trusts you; it is earned by convincing your peers that you know what you are talking about. The more reputation you earn, the more privileges you gain and the more tools you'll have access to on the site - at the highest privilege levels, you'll have access to many of the same tools available to the site moderators."[7] This is the definition from Stack Overflow.

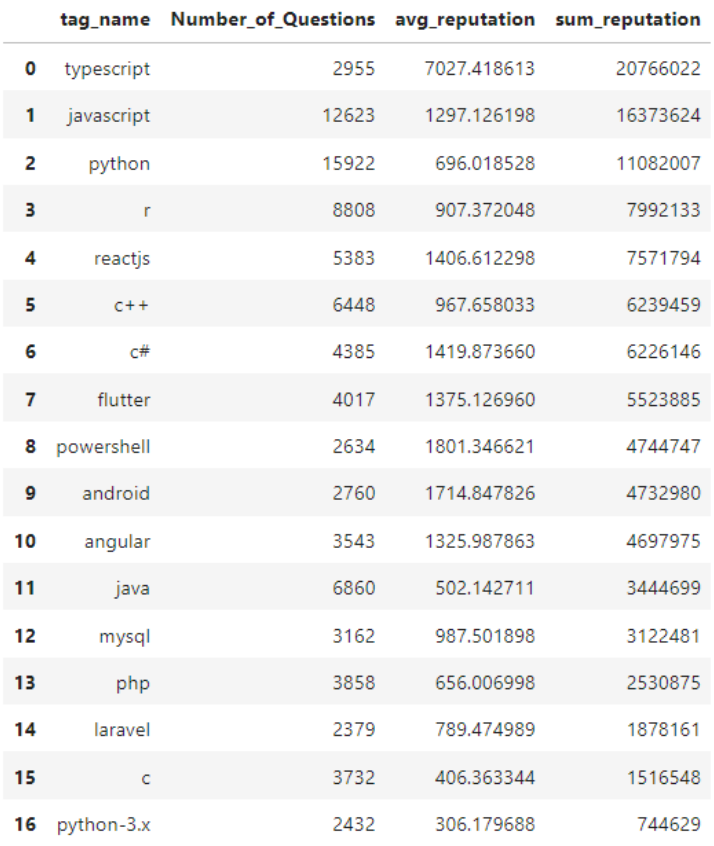
[6] HG Insights, Firmographics of Companies using Microsoft TypeScript, https://discovery.hgdata.com/product/microsoft-typescript
[7] What is reputation ? https://stackoverflow.com/help/whats-reputation

Below is a piece of code that I query the average reputation of question with different technologies.



```
tags_query = """
                SELECT
                tag_name,
                COUNT(q.id) AS Number_of_Questions,
                AVG (u.reputation) AS avg_reputation,
                SUM (u.reputation) AS sum_reputation
                    FROM
                        `bigquery-public-data.stackoverflow.tags` AS t
                    INNER JOIN
                        `bigquery-public-data.stackoverflow.posts_questions` AS q
                    ON t.tag_name = q.tags
                    INNER JOIN
                        `bigquery-public-data.stackoverflow.users` AS u
                    ON q.owner_user_id = u.id
                    WHERE q.creation_date >= '2020-01-01'AND q.creation_date <'2021-01-01' # Can be adjusted to your needs!
                    GROUP BY tag_name
                    HAVING Number_of_Questions > 2000 # Adjust to your taste
                    ORDER BY sum_reputation DESC
                    """
tags = client.query(tags_query).result().to_dataframe()
```

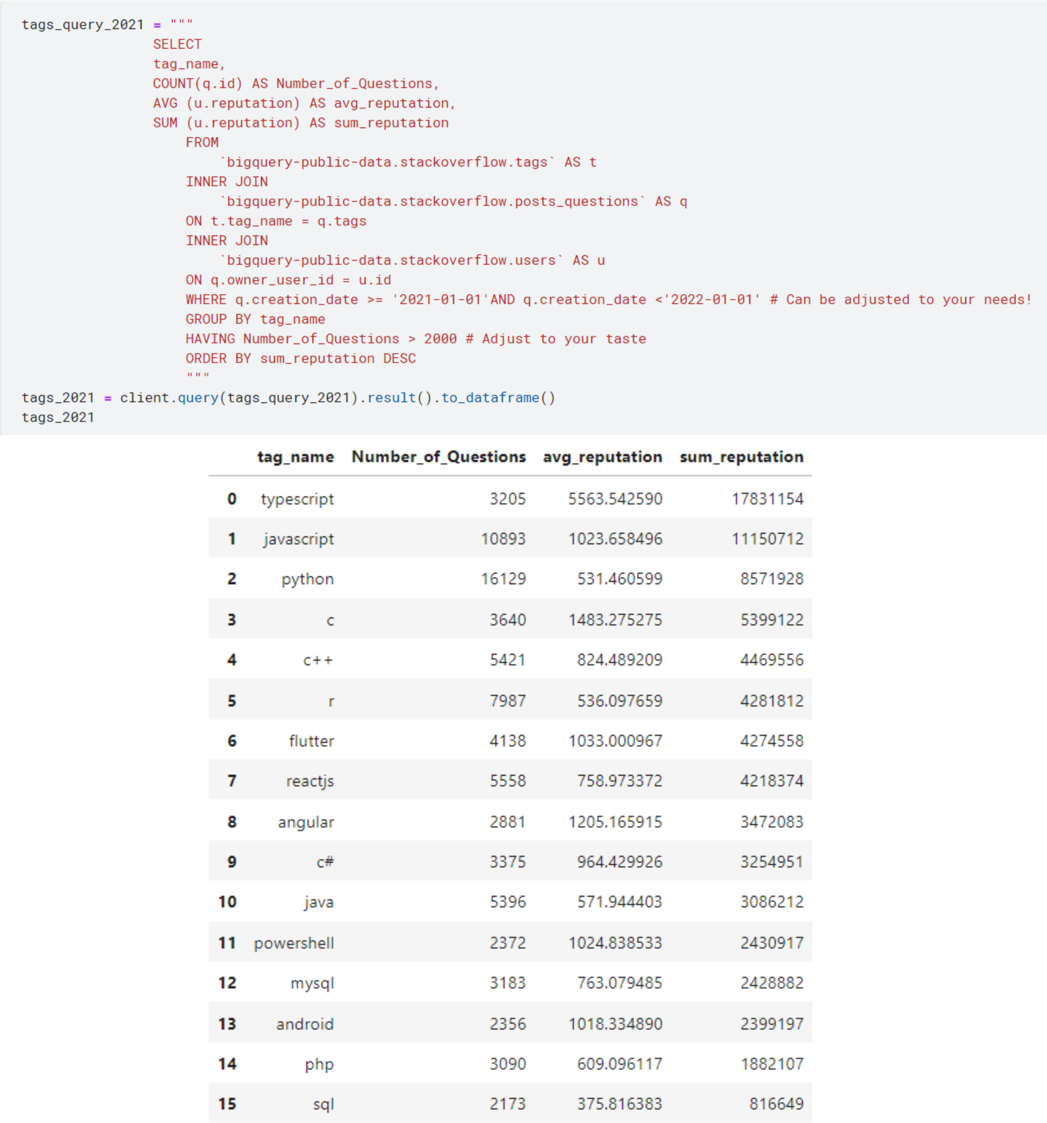
The result is



	tag_name	Number_of_Questions	avg_reputation	sum_reputation
0	typescript	2955	7027.418613	20766022
1	javascript	12623	1297.126198	16373624
2	python	15922	696.018528	11082007
3	r	8808	907.372048	7992133
4	reactjs	5383	1406.612298	7571794
5	c++	6448	967.658033	6239459
6	c#	4385	1419.873660	6226146
7	flutter	4017	1375.126960	5523885
8	powershell	2634	1801.346621	4744747
9	android	2760	1714.847826	4732980
10	angular	3543	1325.987863	4697975
11	java	6860	502.142711	3444699
12	mysql	3162	987.501898	3122481
13	php	3858	656.006998	2530875
14	laravel	2379	789.474989	1878161
15	c	3732	406.363344	1516548
16	python-3.x	2432	306.179688	744629

As we can see from the above page, Typescript has the highest average reputation in 2020, which means that Typescript is used by mostly high-level programmers. Although it has a much lower number of questions than JavaScript, people with high skill would be more likely to use TypeScript.

I will do the same thing in 2021



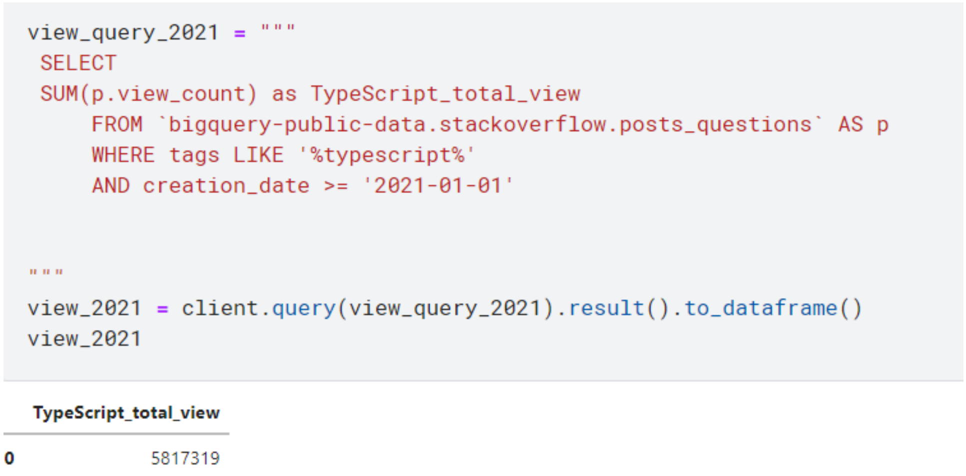
```
tags_query_2021 = """
                SELECT
                tag_name,
                COUNT(q.id) AS Number_of_Questions,
                AVG (u.reputation) AS avg_reputation,
                SUM (u.reputation) AS sum_reputation
                    FROM
                        `bigquery-public-data.stackoverflow.tags` AS t
                    INNER JOIN
                        `bigquery-public-data.stackoverflow.posts_questions` AS q
                    ON t.tag_name = q.tags
                    INNER JOIN
                        `bigquery-public-data.stackoverflow.users` AS u
                    ON q.owner_user_id = u.id
                    WHERE q.creation_date >= '2021-01-01'AND q.creation_date <'2022-01-01' # Can be adjusted to your needs!
                    GROUP BY tag_name
                    HAVING Number_of_Questions > 2000 # Adjust to your taste
                    ORDER BY sum_reputation DESC
                    """
tags_2021 = client.query(tags_query_2021).result().to_dataframe()
tags_2021
```

	tag_name	Number_of_Questions	avg_reputation	sum_reputation
0	typescript	3205	5563.542590	17831154
1	javascript	10893	1023.658496	11150712
2	python	16129	531.460599	8571928
3	c	3640	1483.275275	5399122
4	c++	5421	824.489209	4469556
5	r	7987	536.097659	4281812
6	flutter	4138	1033.000967	4274558
7	reactjs	5558	758.973372	4218374
8	angular	2881	1205.165915	3472083
9	c#	3375	964.429926	3254951
10	java	5396	571.944403	3086212
11	powershell	2372	1024.838533	2430917
12	mysql	3183	763.079485	2428882
13	android	2356	1018.334890	2399197
14	php	3090	609.096117	1882107
15	sql	2173	375.816383	816649

The result is like 2020. TypeScript has the highest reputation but has the lower number of questions, which means the number of people using it is much less than people using JavaScript, but the people who use it are highly skilled developers.

(2) Ranking the view count.

This is TypeScript view count SQL. It counts total views from 2021-01-01 to current for the question related to TypeScript.

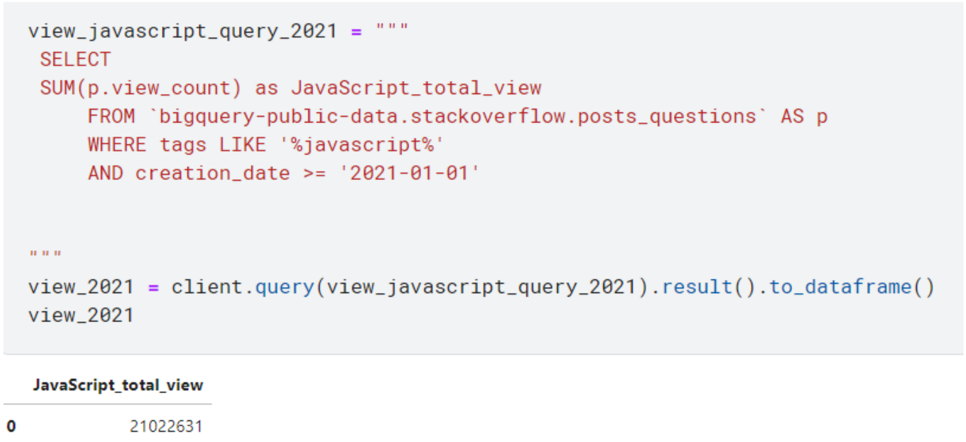


```
view_query_2021 = """
 SELECT
 SUM(p.view_count) as TypeScript_total_view
     FROM `bigquery-public-data.stackoverflow.posts_questions` AS p
     WHERE tags LIKE '%typescript%'
     AND creation_date >= '2021-01-01'


"""
view_2021 = client.query(view_query_2021).result().to_dataframe()
view_2021
```

```
  TypeScript_total_view
0               5817319
```

This is JavaScript view count SQL. It counts total views from 2021-01-01 to current for the question related to JavaScript.



```
view_javascript_query_2021 = """
 SELECT
 SUM(p.view_count) as JavaScript_total_view
     FROM `bigquery-public-data.stackoverflow.posts_questions` AS p
     WHERE tags LIKE '%javascript%'
     AND creation_date >= '2021-01-01'


"""
view_2021 = client.query(view_javascript_query_2021).result().to_dataframe()
view_2021
```

```
  JavaScript_total_view
0              21022631
```

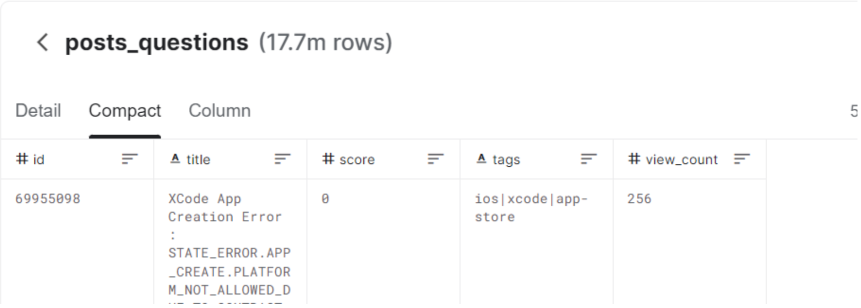
As we can see, TypeScript total view from 2021 to current is 5,817,319, while JavaScript total view is 21,022,631.

JavaScript is more popular on the market.

(3) Analyze Technolgies that are related to TypeScript and JavaScript and check the most popular ones.

In the “post_questions” table, there are title, tags, and view count columns. I check the tags that include TypeScript and the tags that include JavaScript. Since those tags include the other technologies, I can know what other technologies are related to them. First, I will start with JavaScript and then TypeScript.

Below picture is “post_questions” table content:



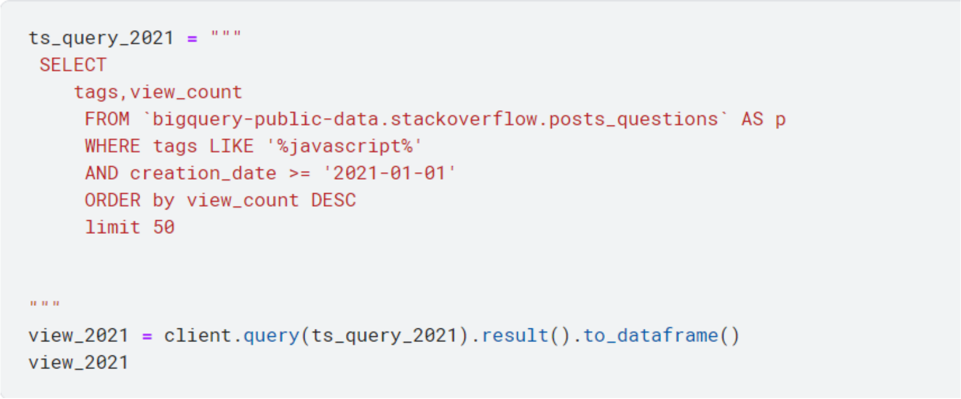
posts_questions (17.7m rows)

# id	title	# score	tags	# view_count
69955098	XCode App Creation Error : STATE_ERROR.APP_CREATE.PLATFORM_NOT_ALLOWED_D	0	ios|xcode|app-store	256

1. JavaScript

I have selected the top 50 views count questions, which includes JavaScript tag. I want to see what other technologies are related to JavaScript.

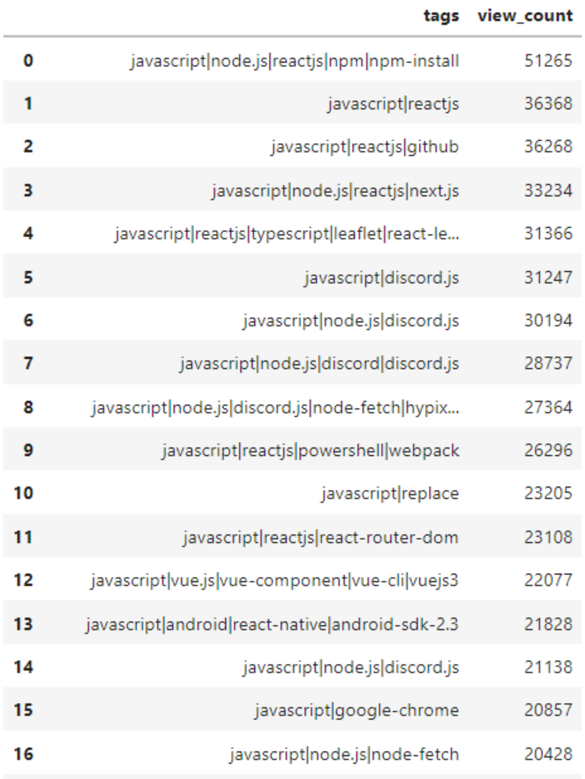
Below is SQL code:



```
ts_query_2021 = """
 SELECT
    tags,view_count
     FROM `bigquery-public-data.stackoverflow.posts_questions` AS p
     WHERE tags LIKE '%javascript%'
     AND creation_date >= '2021-01-01'
     ORDER by view_count DESC
     limit 50


"""
view_2021 = client.query(ts_query_2021).result().to_dataframe()
view_2021
```

The outcome is

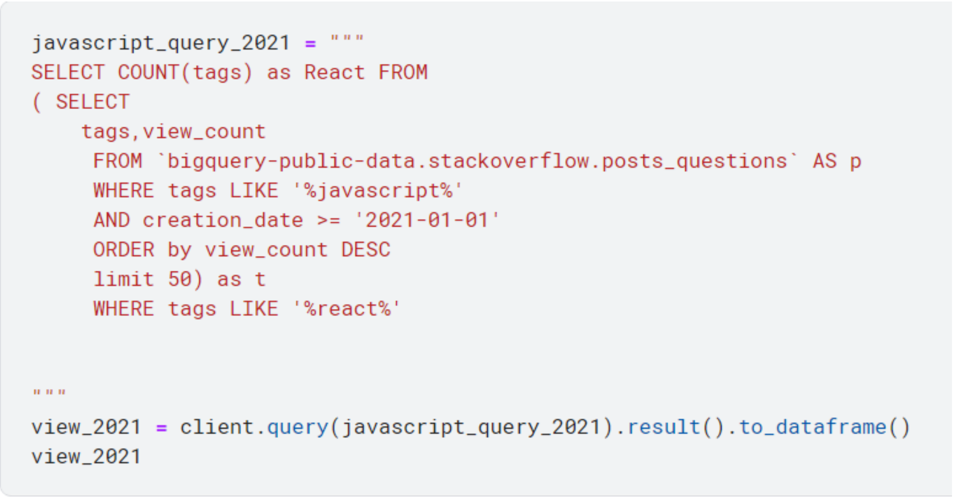


	tags	view_count
0	javascript|node.js|reactjs|npm|npm-install	51265
1	javascript|reactjs	36368
2	javascript|reactjs|github	36268
3	javascript|node.js|reactjs|next.js	33234
4	javascript|reactjs|typescript|leaflet|react-le...	31366
5	javascript|discord.js	31247
6	javascript|node.js|discord.js	30194
7	javascript|node.js|discord|discord.js	28737
8	javascript|node.js|discord.js|node-fetch|hypix...	27364
9	javascript|reactjs|powershell|webpack	26296
10	javascript|replace	23205
11	javascript|reactjs|react-router-dom	23108
12	javascript|vue.js|vue-component|vue-cli|vuejs3	22077
13	javascript|android|react-native|android-sdk-2.3	21828
14	javascript|node.js|discord.js	21138
15	javascript|google-chrome	20857
16	javascript|node.js|node-fetch	20428

T

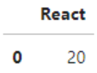
From the above picture, I find that the numbers of occurrences of Node.js, React and Nextjs are higher than others. Therefore, I want to list their frequency, so I can get more details.

Below is the SQL that checks times of each tag showing up in top 50 views questions. There are many related tags. I have checked each one, but I choose React as an example. Below is the code



```
javascript_query_2021 = """
SELECT COUNT(tags) as React FROM
( SELECT
    tags,view_count
     FROM `bigquery-public-data.stackoverflow.posts_questions` AS p
     WHERE tags LIKE '%javascript%'
     AND creation_date >= '2021-01-01'
     ORDER by view_count DESC
     limit 50) as t
     WHERE tags LIKE '%react%'


"""
view_2021 = client.query(javascript_query_2021).result().to_dataframe()
view_2021
```



	React
0	20

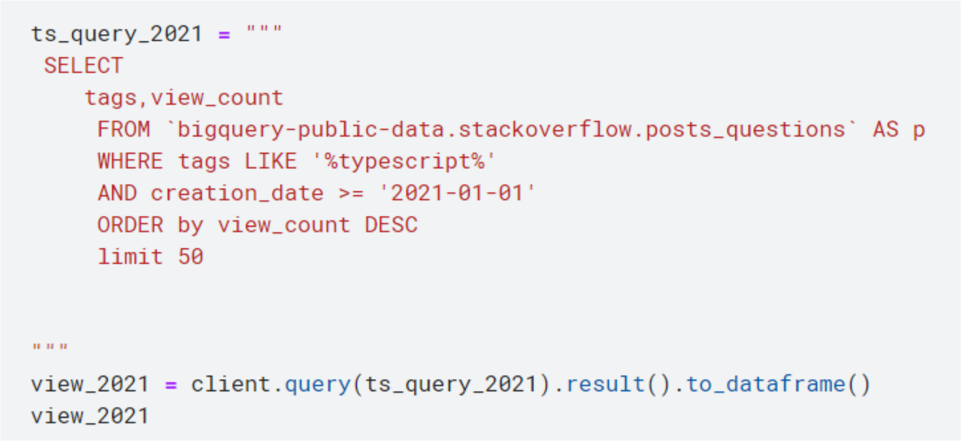
As you can see, the React shows 20 times in top 50 views question. I summary all technologies below for your reference.

	react	Node	Next	npm	discord	Vue	android	Chrome	Express	Webpack
times	20	19	4	5	7	6	2	2	2	6

As we can see, react and node are the most used technologies related to JavaScript. Node is now an extremely popular backend language, and it can be realized by just using JavaScript.

2.TypeScript

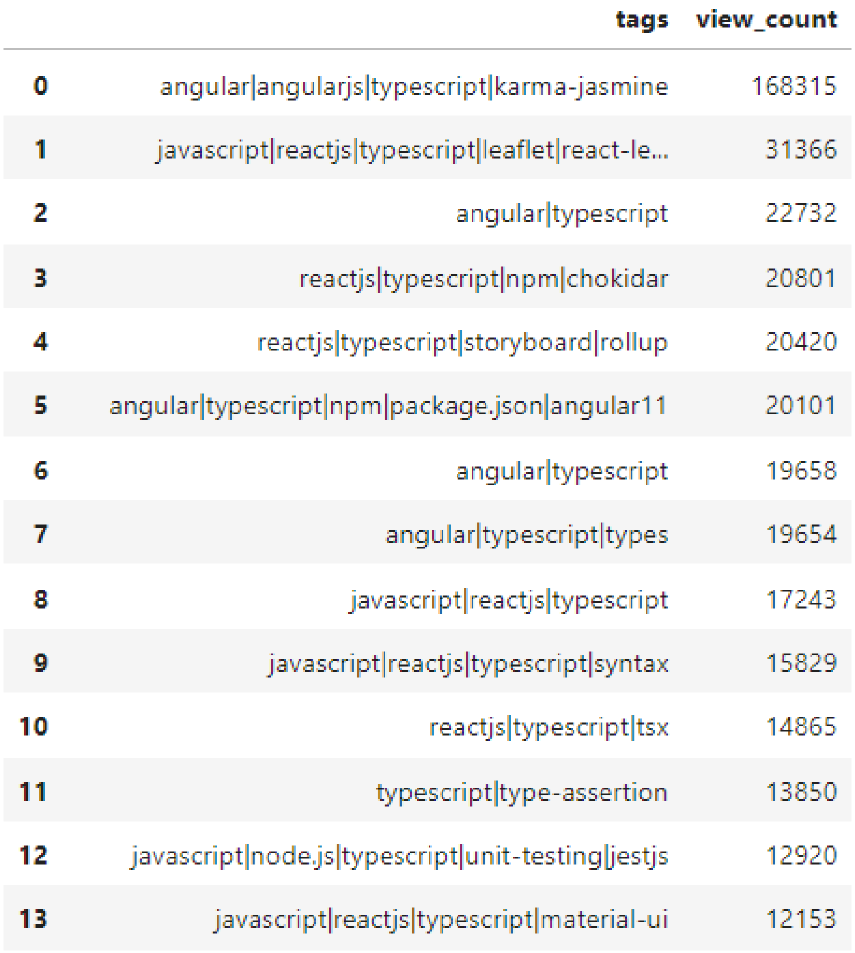
Let me do the same thing with TypeScript.



```
ts_query_2021 = """
 SELECT
    tags,view_count
     FROM `bigquery-public-data.stackoverflow.posts_questions` AS p
     WHERE tags LIKE '%typescript%'
     AND creation_date >= '2021-01-01'
     ORDER by view_count DESC
     limit 50


"""
view_2021 = client.query(ts_query_2021).result().to_dataframe()
view_2021
```

The top questions related to TypeScript are



	tags	view_count
0	angular|angularjs|typescript|karma-jasmine	168315
1	javascript|reactjs|typescript|leaflet|react-le...	31366
2	angular|typescript	22732
3	reactjs|typescript|npm|chokidar	20801
4	reactjs|typescript|storyboard|rollup	20420
5	angular|typescript|npm|package.json|angular11	20101
6	angular|typescript	19658
7	angular|typescript|types	19654
8	javascript|reactjs|typescript	17243
9	javascript|reactjs|typescript|syntax	15829
10	reactjs|typescript|tsx	14865
11	typescript|type-assertion	13850
12	javascript|node.js|typescript|unit-testing|jestjs	12920
13	javascript|reactjs|typescript|material-ui	12153

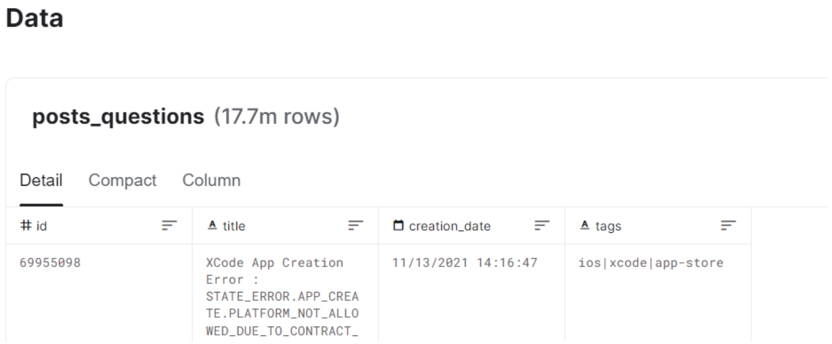
I choose the top 50 questions and count those tag shows up times.

	react	angular	Next	npm	rollup
times	17	22	1	2	2

As we can see, TypeScript is more focused on angular, which is another front-end framework. But people also can run TypeScript with React, and it has become a popular trend now. React has become more popular than Angular, but it is hard to compare React and Angular, since React now adopts TypeScript.

Next, I will compare the percentage change of TypeScript and JavaScript use.

(4) The percentage changes of TypeScript question and JavaScript question in all question from 2009 to 2021



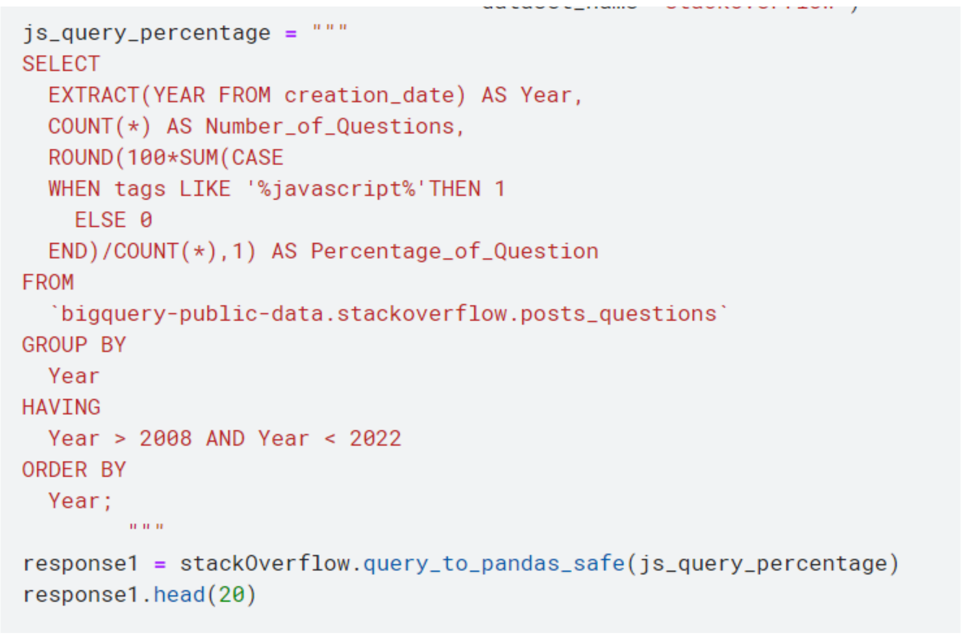
Data

posts_questions (17.7m rows)

# id	title	creation_date	tags
69955098	XCode App Creation Error : STATE_ERROR.APP_CREATE.PLATFORM_NOT_ALLOWED_DUE_TO_CONTRACT_	11/13/2021 14:16:47	ios|xcode|app-store

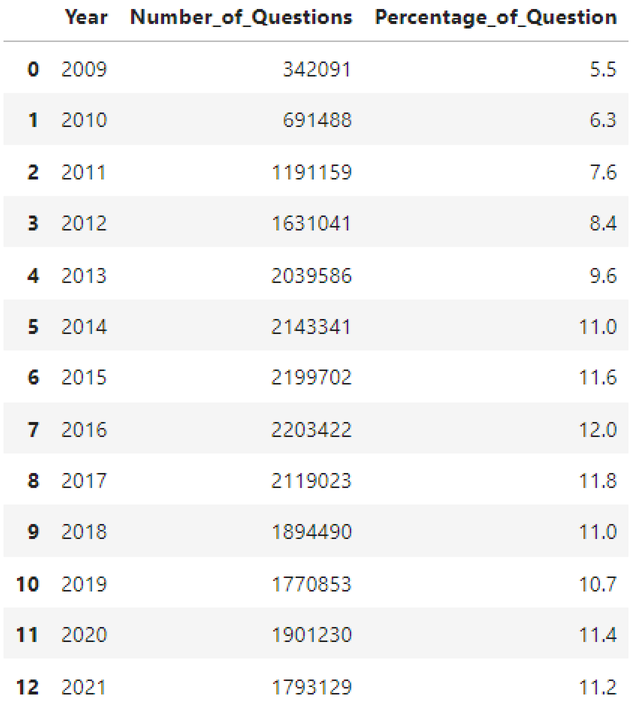
I query from ‘posts_questions’ table. I calculate the percentage of TypeScript question or JavaScript question in all questions, and summaries them from 2009 to 2021 so we can notice the trends in these years.

(1) JavaScript

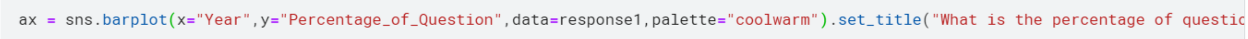


```
js_query_percentage = """
SELECT
  EXTRACT(YEAR FROM creation_date) AS Year,
  COUNT(*) AS Number_of_Questions,
  ROUND(100*SUM(CASE
  WHEN tags LIKE '%javascript%'THEN 1
    ELSE 0
  END)/COUNT(*),1) AS Percentage_of_Question
FROM
  `bigquery-public-data.stackoverflow.posts_questions`
GROUP BY
  Year
HAVING
  Year > 2008 AND Year < 2022
ORDER BY
  Year;
        """
response1 = stackOverflow.query_to_pandas_safe(js_query_percentage)
response1.head(20)
```

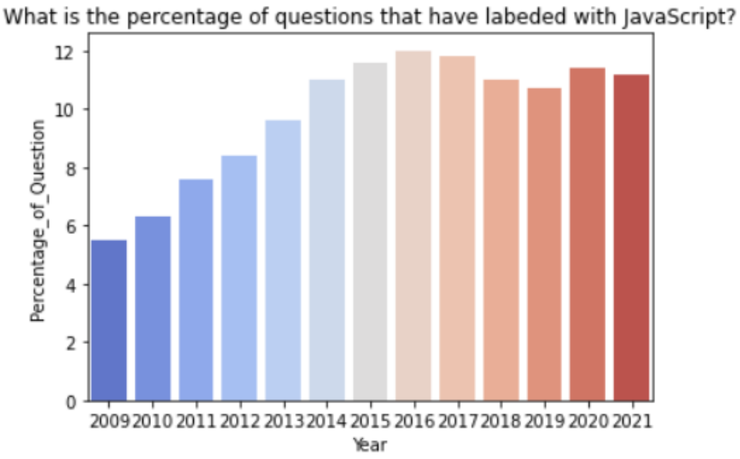
The outcome is



	Year	Number_of_Questions	Percentage_of_Question
0	2009	342091	5.5
1	2010	691488	6.3
2	2011	1191159	7.6
3	2012	1631041	8.4
4	2013	2039586	9.6
5	2014	2143341	11.0
6	2015	2199702	11.6
7	2016	2203422	12.0
8	2017	2119023	11.8
9	2018	1894490	11.0
10	2019	1770853	10.7
11	2020	1901230	11.4
12	2021	1793129	11.2



```
ax = sns.barplot(x="Year",y="Percentage_of_Question",data=response1,palette="coolwarm").set_title("What is the percentage of questic
```

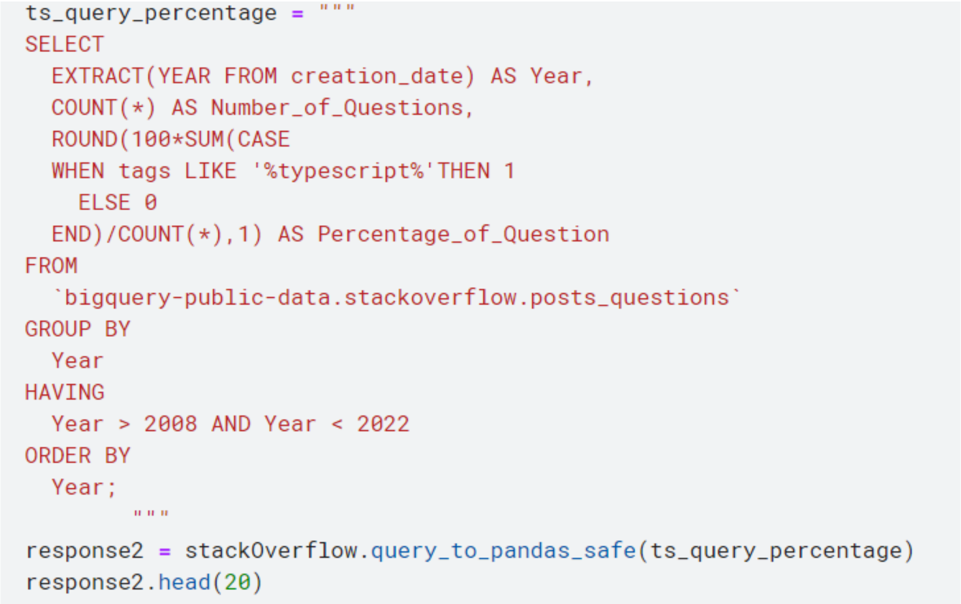


From this picture, I can see that JavaScript question percentage started increasing from 2019. Percentage value jumps from 9.6% in 2013 to 11% in 2014 and then percentage value keeps stable afterward.

(2) TypeScript

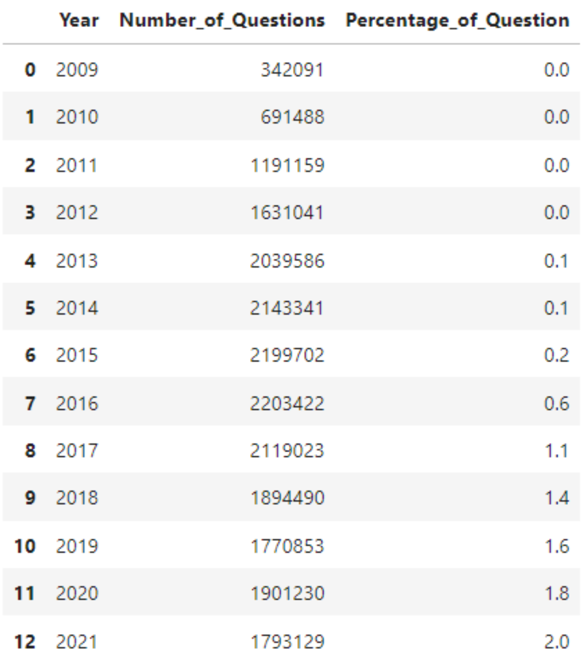
Then, I started to analyze TypeScript. I calculated the question percentage trend below

The SQL code is below

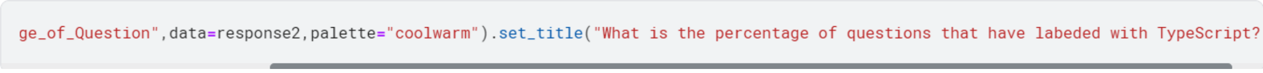


```
ts_query_percentage = """
SELECT
  EXTRACT(YEAR FROM creation_date) AS Year,
  COUNT(*) AS Number_of_Questions,
  ROUND(100*SUM(CASE
  WHEN tags LIKE '%typescript%'THEN 1
    ELSE 0
  END)/COUNT(*),1) AS Percentage_of_Question
FROM
  `bigquery-public-data.stackoverflow.posts_questions`
GROUP BY
  Year
HAVING
  Year > 2008 AND Year < 2022
ORDER BY
  Year;
        """
response2 = stackOverflow.query_to_pandas_safe(ts_query_percentage)
response2.head(20)
```

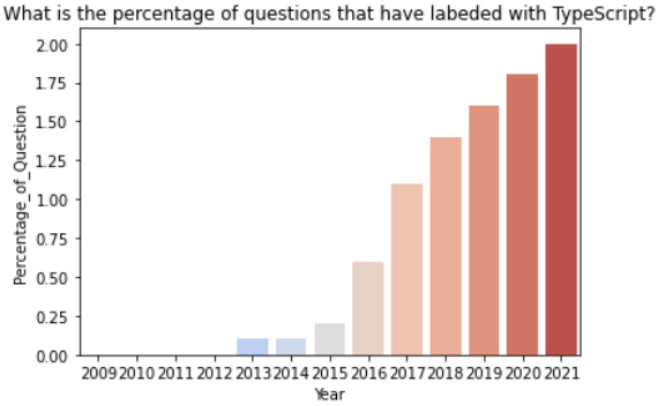
The outcome is



	Year	Number_of_Questions	Percentage_of_Question
0	2009	342091	0.0
1	2010	691488	0.0
2	2011	1191159	0.0
3	2012	1631041	0.0
4	2013	2039586	0.1
5	2014	2143341	0.1
6	2015	2199702	0.2
7	2016	2203422	0.6
8	2017	2119023	1.1
9	2018	1894490	1.4
10	2019	1770853	1.6
11	2020	1901230	1.8
12	2021	1793129	2.0



```
ge_of_Question",data=response2,palette="coolwarm").set_title("What is the percentage of questions that have labeded with TypeScript?
```



As we can see from this picture, TypeScript starts becoming popular in 2017 and the percentage continues increasing. Although it may be lower than JavaScript, but it is steadily favorite by people.

(3) Summary

From the data, we can see that JavaScript is more popular than TypeScript, but people with high programming skills are more likely to use TypeScript. Also, JavaScript is more focused on react framework and node, which is now a popular backend server. While TypeScript is more focused on Angular Framework, which is different from React, but it also starts to be used on react. According to the percentage change, we can see that more people start to use TypeScript, but the increase is small.

3. **Other research**

TypeScript is for safe purposes. After programmers write TypeScript, the compiler achieves soundness by enforcing stricter static checks and embedding residual runtime checks in compiled code. It creates plain JavaScript finally, and the node will run on JavaScript. It is just the compilation that needs time compared to using JavaScript, but programmers can save a lot of time on bugging.

Many people worry about whether Type check would cost a lot of time. According to Microsoft Research, they rely on two innovative ideas to minimize the performance overhead of runtime checks.

The first idea is differential subtyping, “a new form of coercive subtyping that computes the minimum amount of runtime type information that must be added to each object;”

The second idea is erasure modality, “which we use to safely and selectively erase type information. This allows us to scale our design to full-fledged TypeScript, including arrays, maps, classes, inheritance, overloading, and generic types.”8

They did type-checking and compiled around 120,000 lines of TypeScript source code, and found that runtime checks took them some time, about 15% runtime overhead for type safety, but those times

[8] Aseem Rastogi ∗ Nikhil Swamy Cedric Fournet , Safe & Efficient Gradual Typing for TypeScript, 2014

were small for code bases and help them uncover many programming errors. They said that “using TypeScript add significant value to source type annotations at a modest cost. “9

4.Conclusion

TypeScript is a particularly good language, and many good programmers use it, but it is difficult to write, since you must be careful when you enter type, especially with so many libraries. I would recommend that for large applications, people can use TypeScript, since it has type error checking, which may save time in the long term. For small projects, people can use JavaScript and it is easier to write.

4. Reference :

[1]Aseem Rastogi *, Nikhil Swamy , Cedric Fournet, Gavin Bierman , Panagiotis Vekris ∗.
Safe & Efficient Gradual Typing for TypeScript 2015

[2]Panagiotis Vekris, Benjamin Cosman, Ranjit Jhala.

Refinement Types for TypeScript 2016

[3] G. M. Bierman, M. Abadi, and M. Torgersen.

Understanding TypeScript. In Proceedings of ECOOP, 2014

[4] R. Chugh, D. Herman, and R. Jhala. Dependent Types for JavaScript.

In Proceedings of OOPSLA, 2012.

[5] Asger Feldthaus, Anders Møller

Checking Correctness of TypeScript Interfaces for JavaScript Libraries 2014

[6] Stack overflow https://insights.stackoverflow.com/survey/2021

[7] H. Xi and F. Pfenning.

Dependent Types in Practical Programming. In Proceedings of POPL, 1999.

[8] P. Gardner, S. Maffeis, and G. D. Smith. Towards a program logic for JavaScript. In Proceedings of POPL, 2012.

[9] O. Tardieu, N. Nystrom, I. Peshansky, and V. Saraswat.

Constrained Kinds. In Proceedings of OOPSLA, 2012.

[10] Microsoft Corporation. TypeScript v1.4. http://www. Typescriptlang.org/.

[11] M. Furr, J.-h. D. An, J. S. Foster, and M. Hicks. Static Type Inference for Ruby. In Proceedings of the Symposium on Applied Computing, 2009

[12] C. S. Gordon, M. J. Parkinson, J. Parsons, A. Bromfield, and J. Duffy. Uniqueness and Reference Immutability for Safe Parallelism. In Proceedings of OOPSLA, 2012.

[13] Thomas Ball, Peli de Halleux, Michał Moskal

Static TypeScript an Implementation of a Static Compiler for the TypeScript Language 2019

[14] Manuel Serrano. JavaScript AOT compilation. In Proceedings of the 14th ACM SIGPLAN International Symposium on Dynamic Languages. ACM, 50–63. 2018.

[9] Aseem Rastogi ∗ Nikhil Swamy Cedric Fournet, Safe & Efficient Gradual Typing for TypeScript

[15] Manuel Serrano. JavaScript AOT compilation. In Proceedings of the 14th ACM SIGPLAN International Symposium on Dynamic Languages. ACM, 50–63 2018

[16] Gordon Williams. Making Things Smart: Easy Embedded JavaScript Programming for Making Everyday Objects into Intelligent Machines. Maker Media 2017

[17] Patrick Soquet. XS7. https://www.moddable.com/XS7-TC-39.2017

[18] Gregor Richards1 , Francesco Zappa Nardelli2, and Jan Vitek3 Concrete Types for TypeScript 2015

[19] Boris Cherny. Programming TypeScript: Making Your JavaScript Applications Scale. 2019

[20] A. Takikawa, T. S. Stickland, C. Dimoulas, S. Tobin-Hochstadt, and M. Felleisen. Gradual typing for first-class classes. In Proceedings of OOPSLA, 2012.

[21] G. Siek and W. Taha. Gradual typing for functional languages. In Proceedings of Scheme and functional programming workshop, 2006.

[i] Software engineer.