

Practice Case DataFlow

IYKRA DF11 Week 2

Group 3:

- Ashila Ghassani Astmara
- Rizka Virgianti
- Mentari Enggar Rizki
- Rizqullah Ramadhan Murdiansyah
- Yogi Fitriadi Rakhim

Step 1. The CSV data stored in the GCS bucket

- Pertama kita harus membuka google cloud storage untuk membuat GCS bucket dengan create.
- Buat bucket name, location type yang sesuai, storage class, control access, dan protect data.
- Buka bucket yang telah create, pilih upload file dan pilih raw data untuk dimasukkan kedalam bucket.

The screenshot displays the Google Cloud Storage interface for a bucket named 'kdrama_task_ashila'. The bucket is located in 'asia-southeast2 (Jakarta)', uses 'Standard' storage class, and has 'Not public' access. The 'OBJECTS' tab is selected, showing a list of files. The breadcrumb navigation indicates the path: Buckets > kdrama_task_ashila > dramataask. Below the navigation bar, there are buttons for 'UPLOAD FILES', 'UPLOAD FOLDER', 'CREATE FOLDER', 'TRANSFER DATA', 'MANAGE HOLDS', 'DOWNLOAD', and 'DELETE'. A table lists the objects with columns for Name, Size, Type, Created, Storage class, Last modified, Public access, Version history, Encryption, and Retention expiration date. The table contains seven entries, including 'big_query_schema.json', 'bigquery_to_bigquery.py', 'index.html', 'kdrama.csv', 'kdrama.js', 'kdrama_sample.csv', and 'kdrama_sample2.csv'.

Name	Size	Type	Created	Storage class	Last modified	Public access	Version history	Encryption	Retention expiration date
big_query_schema.json	1.5 KB	application/json	Oct 13, 2023, 11:05:20 AM	Standard	Oct 13, 2023, 11:05:20 AM	Not public	—	Google-managed	—
bigquery_to_bigquery.py	3.5 KB	text/x-python	Oct 13, 2023, 11:05:20 AM	Standard	Oct 13, 2023, 11:05:20 AM	Not public	—	Google-managed	—
index.html	10.5 KB	text/html	Oct 13, 2023, 11:05:20 AM	Standard	Oct 13, 2023, 11:05:20 AM	Not public	—	Google-managed	—
kdrama.csv	247.7 KB	text/csv	Oct 13, 2023, 11:05:21 AM	Standard	Oct 13, 2023, 11:05:21 AM	Not public	—	Google-managed	—
kdrama.js	10.4 KB	text/javascript	Oct 13, 2023, 11:05:21 AM	Standard	Oct 13, 2023, 11:05:21 AM	Not public	—	Google-managed	—
kdrama_sample.csv	9.1 KB	text/csv	Oct 13, 2023, 12:40:22 PM	Standard	Oct 13, 2023, 12:40:22 PM	Not public	—	Google-managed	—
kdrama_sample2.csv	4.1 KB	text/csv	Oct 13, 2023, 2:31:52 PM	Standard	Oct 13, 2023, 2:31:52 PM	Not public	—	Google-managed	—

Step 2 : A BQ table from the CSV to BQ DataFlow pipeline

- Setelah memastikan raw data/ input file telah berhasil tersimpan di bucket gcp, kita dapat membuat kode file python untuk membuat dan menjalankan pipeline.
- File csv_gcs_to_bigquery.py merupakan file python untuk membuat dan menjalankan pipeline sebuah data csv ke dalam BigQuery. Kita bisa memodifikasi dalam file tersebut sesuai dengan proses transform data yang kita inginkan
- Pada data kdrama.csv memiliki beberapa kasus dimana kita perlu melakukan cleansing process. Contoh : terdapat row data yang split menjadi data baru. Hal ini dapat dilihat pada row 8 & 9 dibawah ini dengan film Weak Hero 1.

AutoSave

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kdrama2

kd

- Selain itu juga pada kolom Aired_Data , ada case dimana terdapat perbedaan format seperti 'Feb 14,2023' dan 'Jan 1,2023 - Jun 3, 2023'
- Oleh karena itu, kami melakukan beberapa modifikasi untuk dapat menyelesaikan transform data beberapa case diatas. Untuk case split kami menambahkan function MergeLine dan untuk case Aired_Date kami menambahkan beberapa conditional seperti dibawah ini:

```
#cleansing process tp merge every kdrama that split to a row
class MergeLines(beam.DoFn):
    def __init__(self, delimiter=','):
        self.buffered_line = ''
        self.delimiter = delimiter
        self.num_columns = 17
        self._log = logging.getLogger('apache_beam.transforms.usercodetransformcontext')

    def process(self, line):
        import csv

        # Continue appending lines to the buffer without adding extra spaces or stripping
        self.buffered_line += line

        # Attempt to parse the buffered data
        reader = csv.reader([self.buffered_line], delimiter=self.delimiter, quotechar='')
        row = list(reader)[0]

        if len(row) == self.num_columns:
            result_line = self.buffered_line
            self.buffered_line = '' # Clear the buffer for the next set of lines
            self._log.info(f"Merged Line: {result_line}")
            yield result_line
```

```
# Handle different 'Aired Date' formats
if "-" in values[1]: # Check if date range
    aired_date = datetime.strptime(values[1].split("-")[0].strip(), '%b %d, %Y').strftime('%Y-%m-%d')
else:
    aired_date = datetime.strptime(values[1], '%b %d, %Y').strftime('%Y-%m-%d')

self._log.info(f"Used Format for {values[1]} -> {aired_date}")
```

- Setelah itu, kita jalankan program `csv_gcs_to_bigquery.py`. Ketika berhasil, maka akan muncul pipeline job di gcp seperti berikut ini:

The screenshot shows the Google Cloud Dataflow console. The main area displays the 'JOB GRAPH' for a pipeline named 'beamapp-asu...'. The graph shows four steps in a vertical sequence, all with green checkmarks indicating success:

- Read CSV file from GCS**: Succeeded, 0 sec, 2 of 2 stages succeeded.
- Log Raw Data**: Succeeded, 0 sec, 1 of 1 stage succeeded.
- Merge Lines**: Succeeded, 0 sec, 1 of 1 stage succeeded.
- Log Merged Lines**: Succeeded, 0 sec, 1 of 1 stage succeeded.

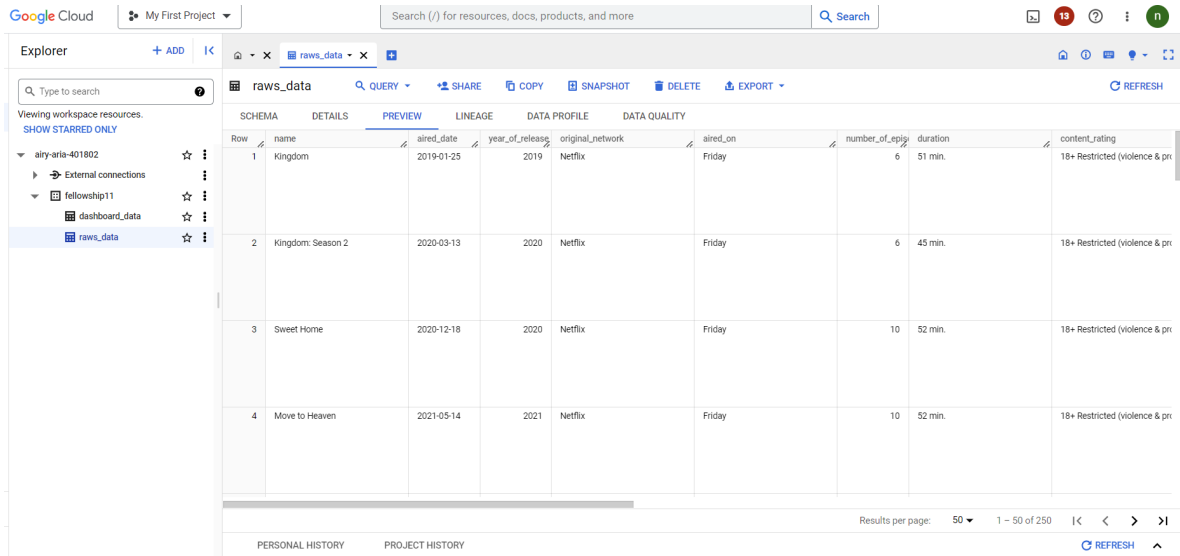
The right-hand panel shows 'Job info' and 'Resource metrics':

Job info	
Job name	beamapp-asuvivobookuser-1013154524-174109-ck67die2
Job ID	2023-10-13_08_45_25-361566885526679569
Job type	Batch
Job status	✓ Succeeded
SDK version	Apache Beam Python 3.11 SDK 2.51.0
Job region	asia-southeast2
Worker location	asia-southeast2
Current workers	0
Latest worker status	Worker pool stopped.
Start time	October 13, 2023 at 10:45:26 PM GMT+7
Elapsed time	3 min 56 sec
Encryption type	Google-managed
Dataflow Prime	Disabled
Runner v2	Enabled
Dataflow Shuffle	Enabled

Resource metrics	
Current vCPUs	1
Total vCPU time	0.043 vCPU hr
Current memory	3.75 GB
Total memory time	0.161 GB hr
Current HDD PD	25 GB
Total HDD PD time	1.073 GB hr
Current SSD PD	0 B

- Jika semua stage pipeline berhasil maka data akan muncul di table bigquery sesuai yang telah kita definisikan sebelumnya.

Berikut contoh output dari pipeline csv to bigquery:



The screenshot shows the Google Cloud BigQuery interface. The left sidebar contains the Explorer panel with a search bar and a list of resources: 'airy-aria-401802', 'External connections', 'fellowship11', 'dashboard_data', and 'raws_data'. The main panel displays the 'raws_data' table with tabs for SCHEMA, DETAILS, PREVIEW, LINEAGE, DATA PROFILE, and DATA QUALITY. The PREVIEW tab is active, showing a table with 4 rows and 9 columns. The columns are: Row, name, aired_date, year_of_release, original_network, aired_on, number_of_episodes, duration, and content_rating. The data rows are: 1. Kingdom (aired 2019-01-25, 6 episodes, 51 min), 2. Kingdom: Season 2 (aired 2020-03-13, 6 episodes, 45 min), 3. Sweet Home (aired 2020-12-18, 10 episodes, 52 min), and 4. Move to Heaven (aired 2021-05-14, 10 episodes, 52 min). The bottom of the interface shows 'Results per page: 50' and '1 - 50 of 250'.

Row	name	aired_date	year_of_release	original_network	aired_on	number_of_episodes	duration	content_rating
1	Kingdom	2019-01-25	2019	Netflix	Friday	6	51 min.	18+ Restricted (violence & pr
2	Kingdom: Season 2	2020-03-13	2020	Netflix	Friday	6	45 min.	18+ Restricted (violence & pr
3	Sweet Home	2020-12-18	2020	Netflix	Friday	10	52 min.	18+ Restricted (violence & pr
4	Move to Heaven	2021-05-14	2021	Netflix	Friday	10	52 min.	18+ Restricted (violence & pr

Step 3. A BQ table from the BQ to BQ DataFlow pipeline

- Setelah memastikan raw data telah berhasil dibuat pada table BigQuery, kita dapat membuat kode file python baru untuk membuat dan menjalankan pipeline.
- File bigquery_to_bigquery.py merupakan file python untuk membuat dan menjalankan pipeline sebuah data dari sebuah table bigquery ke dalam table BigQuery yang lain. Kita bisa memodifikasi dalam file tersebut sesuai dengan output table yang kita inginkan
- Pada tugas ini, kita menginginkan scoring film berdasarkan ratingnya dan tahun rilisnya dan juga dibagi menjadi 3 kelompok yaitu Great, Excellent, dan Exceptional. Terakhir kita juga menghitung rata2 episode pada film yang dirilis setiap tahunnya.
- Oleh karena itu, kita membuat query baru untuk memenuhi case diatas.

```

output = ( pipeline
| "Read data from BigQuery" >> beam.io.ReadFromBigQuery(
    query='''WITH categorized_ratings AS (
        SELECT
            year_of_release,
            CASE
                WHEN rating < 8.5 THEN 'Great'
                WHEN rating >= 8.5 AND rating < 9 THEN 'Excellent'
                WHEN rating >= 9 THEN 'Exceptional'
            END AS rating_category,
            number_of_episodes
        FROM fellowship11.raws_data
    ) ''' \

    '''SELECT
        year_of_release,
        COUNTIF(rating_category = 'Great') AS great,
        COUNTIF(rating_category = 'Excellent') AS excellent,
        COUNTIF(rating_category = 'Exceptional') AS exceptional,
        round(AVG(number_of_episodes),2) AS avg_episodes
    FROM
        categorized_ratings
    GROUP BY
        year_of_release
    ORDER BY
        year_of_release DESC'''

,
    use_standard_sql=True)

```

- Setelah itu, kita jalankan program `bigquery_to_bigquery.py`. Ketika berhasil, maka akan muncul pipeline job di gcp seperti berikut ini:
- Jika semua stage pipeline berhasil maka data akan muncul di table

The screenshot displays the Google Cloud Platform console for a BigQuery pipeline job. The job is named 'beamapp-asu...' and is in a 'Succeeded' state. The job graph shows two stages: 'Read data from BigQuery' (17 sec, 4 of 4 stages succeeded) and 'Write data to BigQuery' (11 sec, 8 of 8 stages succeeded). The job info panel on the right provides details about the job, including its name, ID, status, and resource metrics.

Job info	
Job name	beamapp-asuvsivobookuser-1014120322-407953-obib96f3
Job ID	2023-10-14_05_03_26-9405519093834019717
Job type	Batch
Job status	Succeeded
SDK version	Apache Beam Python 3.11 SDK 2.51.0
Job region	asia-southeast2
Worker location	asia-southeast2
Current workers	0
Latest worker status	Worker pool stopped.
Start time	October 14, 2023 at 7:03:27 PM GMT+7
Elapsed time	4 min 10 sec
Encryption type	Google-managed
Dataflow Prime	Disabled
Runner v2	Enabled
Dataflow Shuffle	Enabled
Resource metrics	
Current vCPUs	1
Total vCPU time	0.047 vCPU hr
Current memory	3.75 GB
Total memory time	0.176 GB hr
Current HDD PD	25 GB
Total HDD PD	1.171 GB hr

- bigquery sesuai yang telah kita definisikan sebelumnya.
- Jika semua stage pipeline berhasil maka data akan muncul di table bigquery sesuai yang telah kita definisikan sebelumnya.
- Berikut contoh output dari pipeline bigquery to bigquery:

Google Cloud

My First Project

Search (/) for resources, docs, products, and more

Explorer

Type to search

Viewing workspace resources.
SHOW STARRED ONLY

airy-aria-401802

External connections

fellowship11

dashboard_data

raws_data

dashboard_data

QUERY

SHARE

COPY

SNAPSHOT

DELETE

EXPORT

SCHEMA

DETAILS

PREVIEW

LINEAGE

DATA PROFILE

DATA QUALITY

Row	year_of_release	great	excellent	exceptional	avg_episodes
1	2010	0	1	0	60.0
2	2009	0	1	0	62.0
3	2003	0	1	0	54.0
4	2007	1	0	0	17.0
5	2006	1	0	0	81.0
6	2014	3	4	0	16.43
7	2011	3	1	0	22.0
8	2013	5	6	0	32.36
9	2012	5	2	0	18.71
10	2016	4	8	1	19.85
11	2015	6	2	1	20.44
12	2017	9	17	1	20.52
13	2019	20	16	1	20.16
14	2018	13	9	2	20.25
15	2022	15	14	3	13.63
16	2021	13	22	4	13.67
17	2020	14	17	4	17.29