Analysis of Global Video Games Sales

Tom Nook Analytics

Hannah Golledge | Hung Nguyen | Monte Rodriguez | Peter Lu

Introduction

Project Objectives:

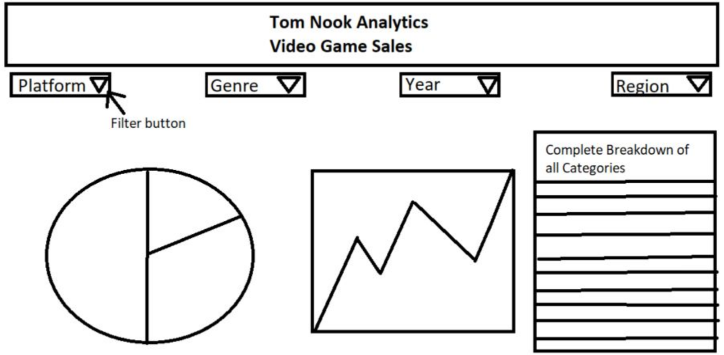
- Conducting Global Video Games Sales Data Analysis using the data source from Kaggle
- Building data visualizations, which are broken down further into genre, platform, publisher, and year

Our Vision

- Using the data, we aimed to create charts that will illustrate the different types of video games that are successful in various regions of the world.

- Focus on providing users an interactive means to explore data themselves.
  a. Dropdowns
  b. Clickable buttons

Initial sketch of website

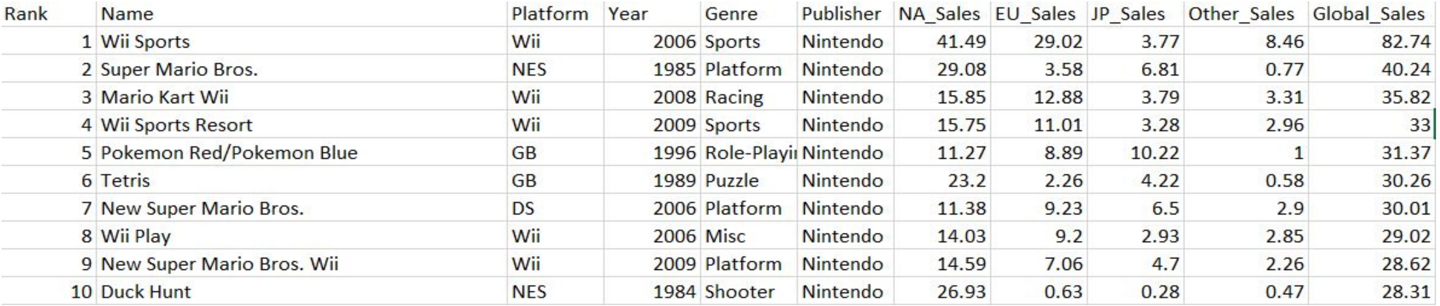


Overview of the steps

- Extract the data from Kaggle.com
- Import and host the data on SQL database
- Wrangle the data into clean and usable formats
- Develop the visualizations
- Set up Flask API
- Push files into GitHub
- Test and launch the web pages

Our Dataset:

Source: Kaggle
Dataset Details: 7 columns x 16599 rows



Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75	11.01	3.28	2.96	33
5	Pokemon Red/Pokemon Blue	GB	1996	Role-Playi	Nintendo	11.27	8.89	10.22	1	31.37
6	Tetris	GB	1989	Puzzle	Nintendo	23.2	2.26	4.22	0.58	30.26
7	New Super Mario Bros.	DS	2006	Platform	Nintendo	11.38	9.23	6.5	2.9	30.01
8	Wii Play	Wii	2006	Misc	Nintendo	14.03	9.2	2.93	2.85	29.02
9	New Super Mario Bros. Wii	Wii	2009	Platform	Nintendo	14.59	7.06	4.7	2.26	28.62
10	Duck Hunt	NES	1984	Shooter	Nintendo	26.93	0.63	0.28	0.47	28.31

Technologies applied

- HTML
- CSS
- D3.js
- Plotly.js
- jQuery.js
- Python Pandas
- Python Flask
- SQLite

Seting up database

- We used DB Browser for SQLite to set up the database.

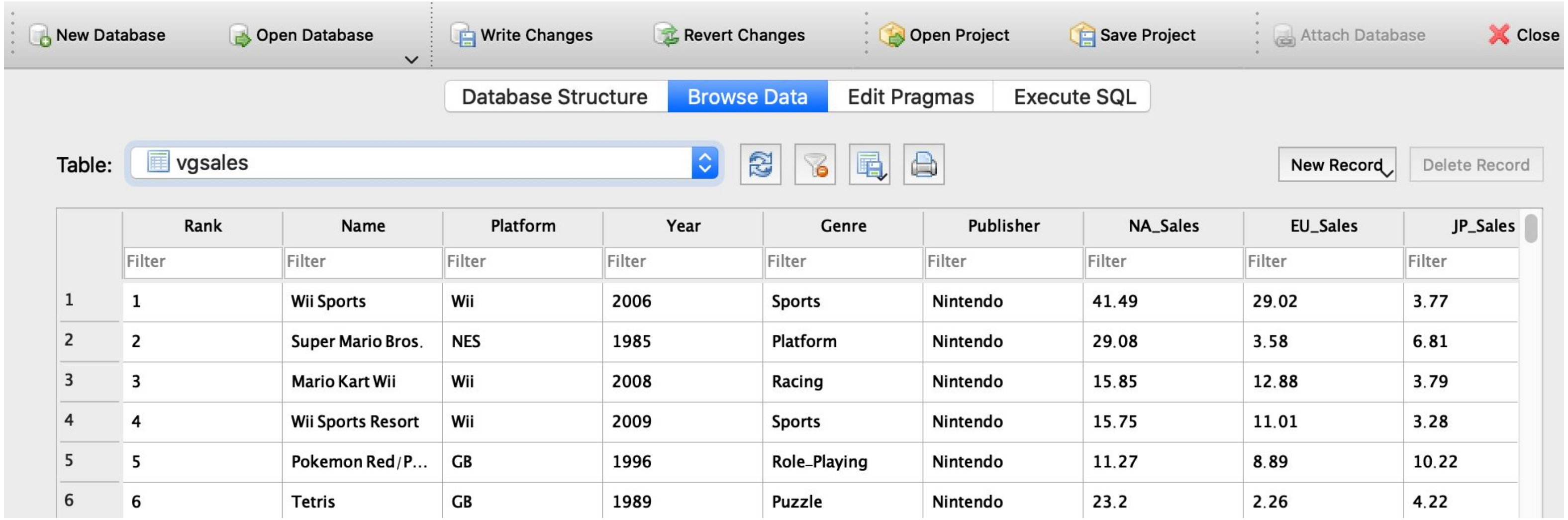
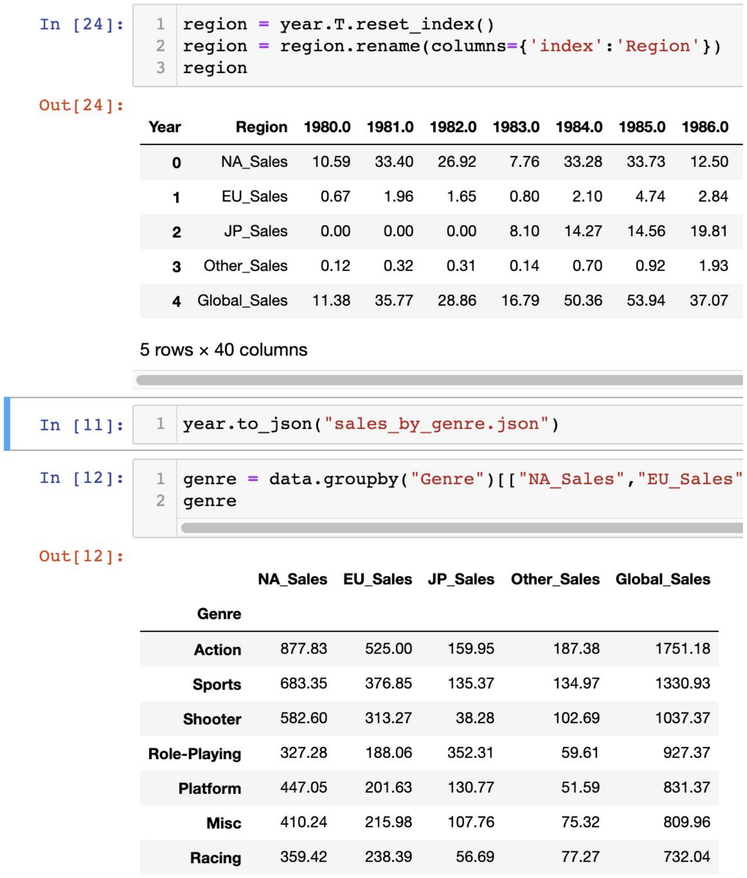


Table: vgsales

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77
2	2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81
3	3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85	12.88	3.79
4	4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75	11.01	3.28
5	5	Pokemon Red/P...	GB	1996	Role_Playing	Nintendo	11.27	8.89	10.22
6	6	Tetris	GB	1989	Puzzle	Nintendo	23.2	2.26	4.22

Wrangling data

… and then Python Pandas to clean up, manipulate the data and export to appropriate formats for use in building charts



In [24]:
```
region = year.T.reset_index()
region = region.rename(columns={'index':'Region'})
region
```

Out[24]:

Year	Region	1980.0	1981.0	1982.0	1983.0	1984.0	1985.0	1986.0
0	NA_Sales	10.59	33.40	26.92	7.76	33.28	33.73	12.50
1	EU_Sales	0.67	1.96	1.65	0.80	2.10	4.74	2.84
2	JP_Sales	0.00	0.00	0.00	8.10	14.27	14.56	19.81
3	Other_Sales	0.12	0.32	0.31	0.14	0.70	0.92	1.93
4	Global_Sales	11.38	35.77	28.86	16.79	50.36	53.94	37.07

5 rows × 40 columns

In [11]:
```
year.to_json("sales_by_genre.json")
```

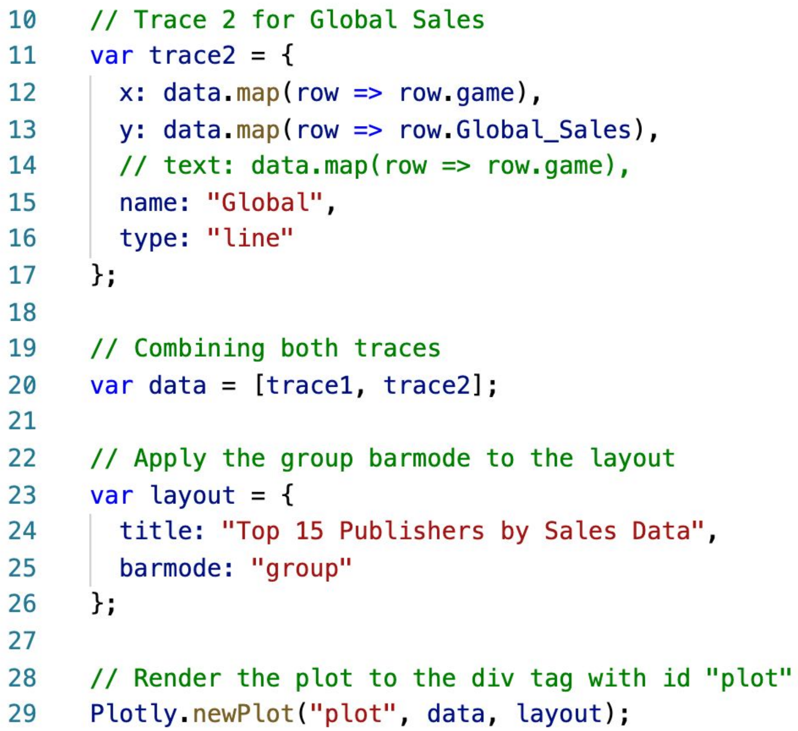
In [12]:
```
genre = data.groupby("Genre")[["NA_Sales","EU_Sales"
genre
```

Out[12]:

	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
Genre					
Action	877.83	525.00	159.95	187.38	1751.18
Sports	683.35	376.85	135.37	134.97	1330.93
Shooter	582.60	313.27	38.28	102.69	1037.37
Role-Playing	327.28	188.06	352.31	59.61	927.37
Platform	447.05	201.63	130.77	51.59	831.37
Misc	410.24	215.98	107.76	75.32	809.96
Racing	359.42	238.39	56.69	77.27	732.04

Building the charts

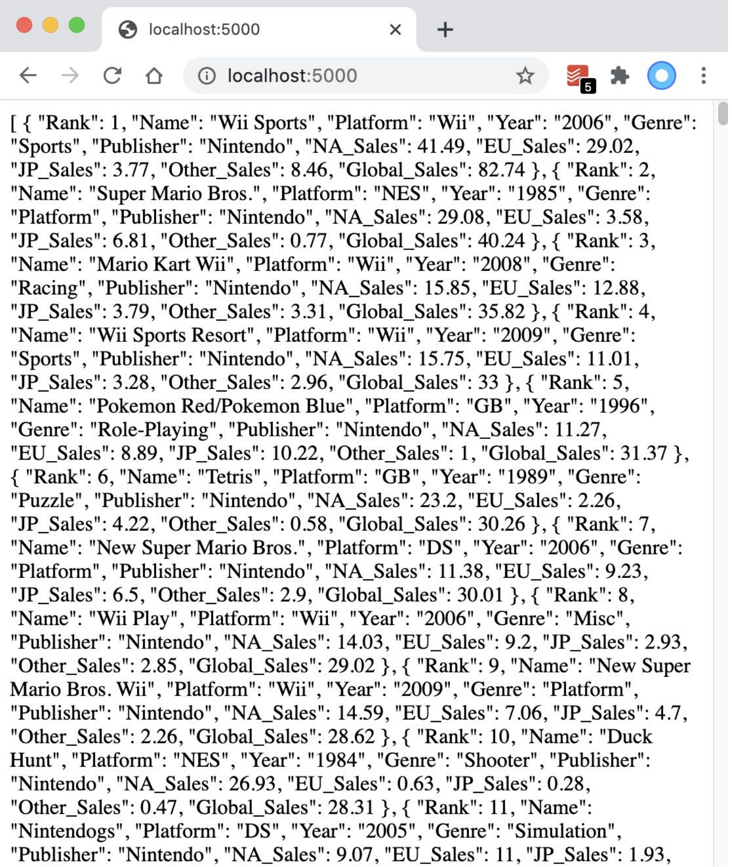
We used HTML, CSS, D3.js and Plotly.js to develop visualizations



```
// Trace 2 for Global Sales
var trace2 = {
  x: data.map(row => row.game),
  y: data.map(row => row.Global_Sales),
  // text: data.map(row => row.game),
  name: "Global",
  type: "line"
};

// Combining both traces
var data = [trace1, trace2];

// Apply the group barmode to the layout
var layout = {
  title: "Top 15 Publishers by Sales Data",
  barmode: "group"
};

// Render the plot to the div tag with id "plot"
Plotly.newPlot("plot", data, layout);
```

Flask-powered API

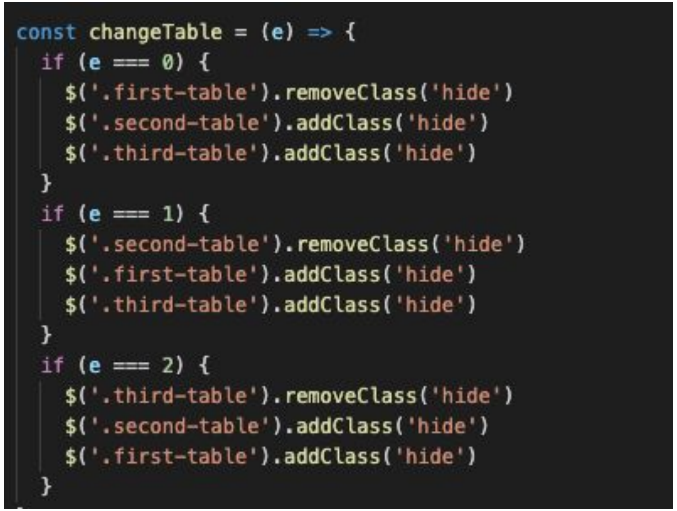
We also included a Flask app that helps users access RESTful API data under json format



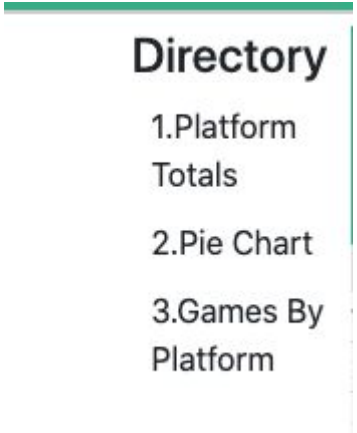
```
[ { "Rank": 1, "Name": "Wii Sports", "Platform": "Wii", "Year": "2006", "Genre":
"Sports", "Publisher": "Nintendo", "NA_Sales": 41.49, "EU_Sales": 29.02,
"JP_Sales": 3.77, "Other_Sales": 8.46, "Global_Sales": 82.74 }, { "Rank": 2,
"Name": "Super Mario Bros.", "Platform": "NES", "Year": "1985", "Genre":
"Platform", "Publisher": "Nintendo", "NA_Sales": 29.08, "EU_Sales": 3.58,
"JP_Sales": 6.81, "Other_Sales": 0.77, "Global_Sales": 40.24 }, { "Rank": 3,
"Name": "Mario Kart Wii", "Platform": "Wii", "Year": "2008", "Genre":
"Racing", "Publisher": "Nintendo", "NA_Sales": 15.85, "EU_Sales": 12.88,
"JP_Sales": 3.79, "Other_Sales": 3.31, "Global_Sales": 35.82 }, { "Rank": 4,
"Name": "Wii Sports Resort", "Platform": "Wii", "Year": "2009", "Genre":
"Sports", "Publisher": "Nintendo", "NA_Sales": 15.75, "EU_Sales": 11.01,
"JP_Sales": 3.28, "Other_Sales": 2.96, "Global_Sales": 33 }, { "Rank": 5,
"Name": "Pokemon Red/Pokemon Blue", "Platform": "GB", "Year": "1996",
"Genre": "Role-Playing", "Publisher": "Nintendo", "NA_Sales": 11.27,
"EU_Sales": 8.89, "JP_Sales": 10.22, "Other_Sales": 1, "Global_Sales": 31.37 },
{ "Rank": 6, "Name": "Tetris", "Platform": "GB", "Year": "1989", "Genre":
"Puzzle", "Publisher": "Nintendo", "NA_Sales": 23.2, "EU_Sales": 2.26,
"JP_Sales": 4.22, "Other_Sales": 0.58, "Global_Sales": 30.26 }, { "Rank": 7,
"Name": "New Super Mario Bros.", "Platform": "DS", "Year": "2006", "Genre":
"Platform", "Publisher": "Nintendo", "NA_Sales": 11.38, "EU_Sales": 9.23,
"JP_Sales": 6.5, "Other_Sales": 2.9, "Global_Sales": 30.01 }, { "Rank": 8,
"Name": "Wii Play", "Platform": "Wii", "Year": "2006", "Genre": "Misc",
"Publisher": "Nintendo", "NA_Sales": 14.03, "EU_Sales": 9.2, "JP_Sales": 2.93,
"Other_Sales": 2.85, "Global_Sales": 29.02 }, { "Rank": 9, "Name": "New Super
Mario Bros. Wii", "Platform": "Wii", "Year": "2009", "Genre": "Platform",
"Publisher": "Nintendo", "NA_Sales": 14.59, "EU_Sales": 7.06, "JP_Sales": 4.7,
"Other_Sales": 2.26, "Global_Sales": 28.62 }, { "Rank": 10, "Name": "Duck
Hunt", "Platform": "NES", "Year": "1984", "Genre": "Shooter", "Publisher":
"Nintendo", "NA_Sales": 26.93, "EU_Sales": 0.63, "JP_Sales": 0.28,
"Other_Sales": 0.47, "Global_Sales": 28.31 }, { "Rank": 11, "Name":
"Nintendogs", "Platform": "DS", "Year": "2005", "Genre": "Simulation",
"Publisher": "Nintendo", "NA_Sales": 9.07, "EU_Sales": 11, "JP_Sales": 1.93,
```

Jquery

Jquery was used to build an Interactive sidebar that allows viewers to switch between different charts.



```
const changeTable = (e) => {
  if (e === 0) {
    $('.first-table').removeClass('hide')
    $('.second-table').addClass('hide')
    $('.third-table').addClass('hide')
  }
  if (e === 1) {
    $('.second-table').removeClass('hide')
    $('.first-table').addClass('hide')
    $('.third-table').addClass('hide')
  }
  if (e === 2) {
    $('.third-table').removeClass('hide')
    $('.second-table').addClass('hide')
    $('.first-table').addClass('hide')
  }
```



Directory

1.Platform Totals

2.Pie Chart

3.Games By Platform

Showcase webpages

- Next, let's enjoy the final products: visualizations built by team Tom Nook Analytics!

Questions?