



SEOUL NATIONAL UNIVERSITY  
**Graduate School of Data Science**

**M3239.003100: Data Analysis and Visualization**

Lecture 3

# Data Cleaning / Storing / Merging

Hyunwoo Park

Graduate School of Data Science

Seoul National University


# Agenda

- Demo
  - Chrome Developer Tools
  - Naver Webtoon
  - Starbucks Locations in Korea
  - Code/Text Editor
- Data Cleaning
  - Types of data dirtiness
  - Importance of data cleaning
  - Data Wrangler / Open Refine
  - Regular Expression
  - OCR
- Data Storing
  - CSV (or other text file-based storage)
  - Sqlite
  - RDBS / NoSQL
- Data Merging
  - SQL
  - Python

# Demo

# Chrome Developer Tools

- <https://developer.chrome.com/docs/devtools/>

 Chrome Developers


Search docs, blogs and more

Home

Docs

Blog

Documentation



Chrome DevTools

Chrome DevTools is a set of web developer tools built directly into the Google Chrome browser.

Overview

Get started with Google Chrome's built-in web developer tools.

Open Chrome DevTools

All of the ways that you can open Chrome DevTools.


What's new in DevTools

Stay up to date with the latest DevTools changes.

Engineering blog

Learn how the team builds new features in DevTools.

# CSS



[View and change CSS](#)

Learn how to use Chrome DevTools to view and change a page's CSS.

# Example: yestrade

- <https://yestrade.go.kr/>

알림/정보마당

**yestrade**

전략물자관리시스템

전략물자관리제도 안내

판정/허가신청

기타민원 신청

자율준수무역거래자

**알림/정보마당**

이용안내

 산업통상자원부  
Ministry of Trade, Industry and Energy

**온라인 자가판정**

전략물자 해당 여부를 무역거래자가 직접 판정하는 서비스

시작하기

**전문판정 신청**

전문판정기관에 전략물자 판정을 의뢰하여 전자물자 해당여부를 확인받는 서비스

신청하기

**수출허가 신청**

전략물자의 경우 허가기관으로부터 수출허가서를 취득하여 수출

신청하기

☒ 공동인증서 로그인

아이디

비밀번호

로그인

인증센터  회원가입 | 아이디/비밀번호찾기

검색어를 입력해주세요. 

**공지사항** 

• 2022년 미국 수출통제제도 설명회 개최

'22.08.23

• 2022 무역의 날 전략물자 수출관리 유공자 포상 안내...

'22.07.14

• 자율준수무역거래자 실적보고(22년도 상반기) 작성 안...

'22.07.04

• 2022년도 무역안보의날 기념행사 개최

'22.06.24

• 전략물자관리시스템 서버 점검 안내

'22.06.15

**자주 찾는 서비스를 이용하세요.**

 DENIAL LIST 검색

 수출거래 보고서 작성

 수입목적확인 발급신청

 방산업체 지정 신청

 정기보고 작성

 자율준수 무역거래자



## Example 1: Naver Webtoon

- <https://comic.naver.com/webtoon/detail?titleId=748105&no=1&weekday=thu>
- Related shop items are fetched as JSON asynchronously.
- Comments are in an IFRAME.

← → ↺ comic.naver.com/webtoon/detail?titleId=748105&no=1&weekday=sun ☆ ⚙️ 👤

태그 독립일기 책갈피하기

희열별점 ★★★★★ 9.95 (참여 56452) 별점주기 ★★★★★ 확인

☺️ 관심템튼 첫회보기 목록보기

예고편 1화 나도 혼자 ... 2화 부등산 대모험 3화 이사 첫날

작가의 말 (자카)  
혼자 사는 독자들을 손 들어 보세요! '!( < > )' 반갑습니다!

작가의 다른 작품 작가 블로그 팬카페

추석선물은 쇼핑라이브가 학계의 정설 자세히

의견쓰기 13,422

주제와 무관한 댓글이나 스포일러, 악플은 경고조치 없이 삭제되며 정제 대상이 될 수 있습니다.

0/500 등록

✓ BEST댓글 전체댓글  
클린봇이 악성댓글을 감지합니다. 설정

이축복(prai\*\*\*\*)  
BEST 자카님이 돌아오셨다 풍악을 울려라!!! 🎉🎉🎉  
2020-06-13 23:02  
답글 17 47487 98

파자마 & 플로우션  
편딩예약중!

전체 캐릭터

자카와 바보개의 일상 머그컵 12,000원

볼 때마다 힐링되는 자카 스티커 3,000원

자꾸만 붙이고 싶어지는 포스트잇 2,500원

반전 매력의 무선노트 3,500원

매일 계획을 세워보까! 주간 플래너 7,000원

1/3 < >

## Example 2: Starbucks Locations in Korea

- [https://www.starbucks.co.kr/store/store\\_map.do](https://www.starbucks.co.kr/store/store_map.do)

starbucks.co.kr/store/store\_map.do

Sign In | My Starbucks | Customer Service & Ideas | Find a Store

COFFEE | MENU | STORE | RESPONSIBILITY | STARBUCKS REWARDS | WHAT'S NEW

매장찾기 | 길찾기

STEP 1 : 시/도를 선택해 주세요.

서울	경기
광주	대구
대전	부산
울산	인천
강원	경남
경북	전남
전북	충남
충북	제주
세종	

Map showing Starbucks locations in Seoul, Korea. The map displays a grid of streets and various landmarks. A green line indicates a route starting from a Starbucks location in the center of the map, passing through several other Starbucks locations, and ending near the Han River. The map also shows various other businesses and landmarks in the area.

# 🔑 Code/Text Editor (Sublime Text)

- Multiple Cursors

- 1) Select next instance of the selected text
  - [https://www.sublimetext.com/docs/multiple\\_selection\\_with\\_the\\_keyboard.html](https://www.sublimetext.com/docs/multiple_selection_with_the_keyboard.html) (See "Quick Add Next")
- 2) Select all instances of the selected text
  - <https://stackoverflow.com/questions/12162047/how-to-select-all-instances-of-selected-region-in-sublime-text>
- 3) Column editing

On Mac OS you can use: **CMD** + **CTRL** + **G**

On Windows/Linux press **Alt** + **F3**.

- Move lines

- Duplicate lines



# Data Cleaning

## How dirty is real data?



## 📌 An example of dirty data

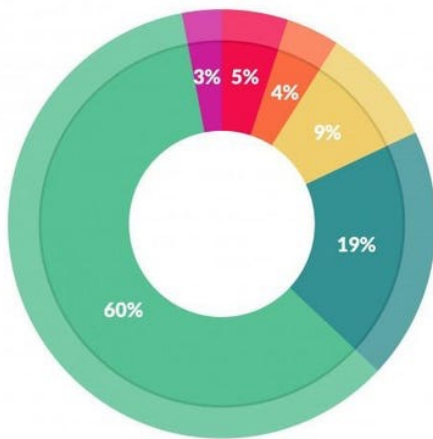
- Sep 9, 2021
- September 9, 21
- 9/9/21
- 2021-09-09
- 21/9/9
- 9th September 2021

# 📌 Types of data dirtiness

- Typos
- Missing data/fields
- Units (different)
- Data types
- Abbreviations
- Duplicates or variations of the same entity
- Encoding
- Dashes (en-dash, em-dash, hyphen), parentheses, delimiters
- White spaces

# 🔑 Importance of data cleaning

- “80%” time spent on data preparation
- Cleaning Big Data: Most Time-Consuming, Least Enjoyable Data Science Task, Survey Says [Forbes]
  - <https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/>
- [https://en.wikipedia.org/wiki/Data\\_janitor](https://en.wikipedia.org/wiki/Data_janitor)

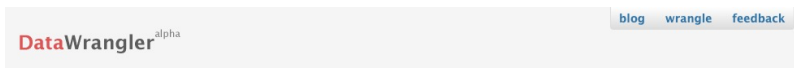


What data scientists spend the most time doing

- Building training sets: 3%
- Cleaning and organizing data: 60%
- Collecting data sets: 19%
- Mining data for patterns: 9%
- Refining algorithms: 4%
- Other: 5%

# Data cleaner tools

- Data Wrangler
  - Research at Stanford
  - <http://vis.stanford.edu/wrangler/>



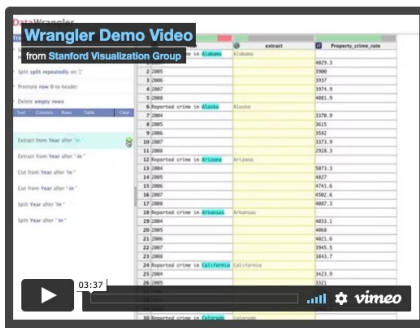
Wrangler is an interactive tool for data cleaning and transformation. Spend less time formatting and more time analyzing your data.

**UPDATE:** The Stanford/Berkeley Wrangler research project is complete, and the software is no longer actively supported. Instead, we have started a commercial venture, [Trifacta](#). For the most recent version of the tool, see the free [Trifacta Wrangler](#).

Why wrangle?

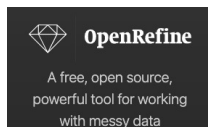
- Too much time is spent manipulating data just to get analysis and visualization tools to read it. Wrangler is designed to accelerate this process: spend less time fighting with your data and more time learning from it.
- Wrangler allows interactive transformation of messy, real-world data into the data tables analysis tools expect. Export data for use in Excel, R, Tableau, Protovis, ...
- Want to learn more about Wrangler's design? Take a look at our [research paper](#).
- Wrangler is still a work-in-progress. Please share your [feedback](#) and [feature requests](#)!

TRY IT NOW



© 2011–2013 Stanford Visualization Group. All rights reserved.

- OpenRefine
  - Previously Google Refine
  - <https://openrefine.org/>



- Home
- Community
- Documentation
- Download
- Data Privacy
- Contact Us
- Blog

## Welcome!

OpenRefine (previously Google Refine) is a powerful tool for working with messy data: cleaning it; transforming it from one format into another; and extending it with web services and external data.

OpenRefine always keeps your data private on your own computer until YOU want to share or collaborate. Your private data never leaves your computer unless you want it to. (It works by running a small server on your computer and you use your web browser to interact with it)

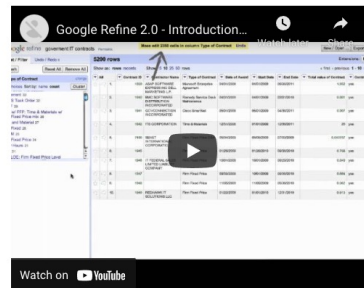
OpenRefine is available in more than 15 languages.

OpenRefine is part of [Code for Science & Society](#).

## Introduction to OpenRefine

### 1. Explore Data

OpenRefine can help you explore large data sets with ease. You can find out more about this functionality by watching the video below.



## 📌 What can Data Wrangler and OpenRefine do?

- [O] Clustering by similarity
- [O, W] Removing empty space
- [O, W] Reformatting
- [W, O] Comprehension/exporting of transformation (e./g., to excel, javascript)
- [W] Keyword extraction
- [O] Different unit (scaling, distribution); outliers [W] suggestions
- [W] Changing data types
- [O, W] undo/redo
- [O, W] Sorting
- [O] supporting scripting



# Regular expressions

- Python built-in package: re

Python » English » 3.8.12 » Documentation » The Python Standard Library » Text Processing Services »

## Table of Contents

- re — Regular expression operations
  - Regular Expression Syntax
  - Module Contents
  - Regular Expression Objects
  - Match Objects
  - Regular Expression Examples
    - Checking for a Pair
    - Simulating scanf()
    - search() vs. match()
    - Making a Phonebook
    - Text Munging
    - Finding all Adverbs
    - Finding all Adverbs and their Positions
    - Raw String Notation
    - Writing a Tokenizer

## Previous topic

string — Common string operations

## Next topic

difflib — Helpers for computing deltas

## This Page

Report a Bug  
Show Source

## re — Regular expression operations

Source code: [Lib/re.py](#)

This module provides regular expression matching operations similar to those found in Perl.

Both patterns and strings to be searched can be Unicode strings ([str](#)) as well as 8-bit strings ([bytes](#)). However, Unicode strings and 8-bit strings cannot be mixed: that is, you cannot match a Unicode string with a byte pattern or vice-versa; similarly, when asking for a substitution, the replacement string must be of the same type as both the pattern and the search string.

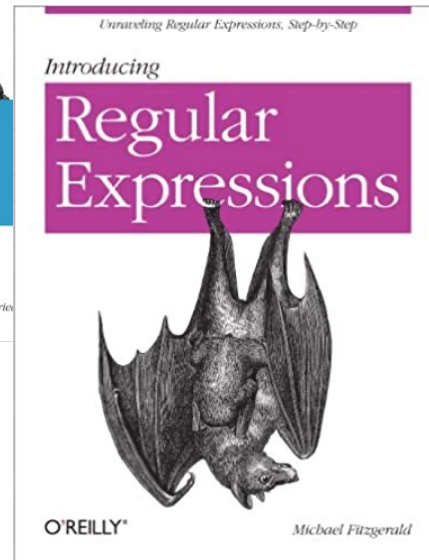
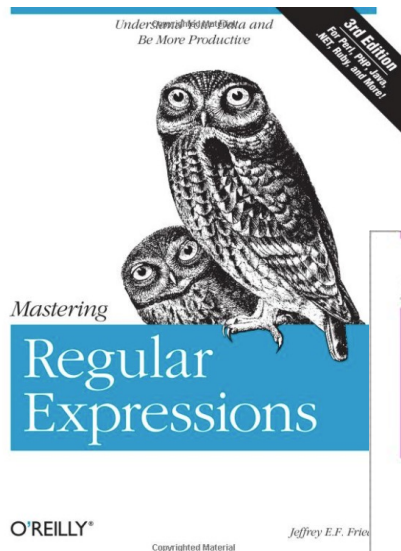
Regular expressions use the backslash character ('\\') to indicate special forms or to allow special characters to be used without invoking their special meaning. This collides with Python's usage of the same character for the same purpose in string literals; for example, to match a literal backslash, one might have to write '\\\\' as the pattern string, because the regular expression must be '\\', and each backslash must be expressed as '\\\\' inside a regular Python string literal. Also, please note that any invalid escape sequences in Python's usage of the backslash in string literals now generate a [DeprecationWarning](#) and in the future this will become a [SyntaxError](#). This behaviour will happen even if it is a valid escape sequence for a regular expression.

The solution is to use Python's raw string notation for regular expression patterns; backslashes are not handled in any special way in a string literal prefixed with 'r'. So `r"\n"` is a two-character string containing '\n' and '\n', while `"\n"` is a one-character string containing a newline. Usually patterns will be expressed in Python code using this raw string notation.

It is important to note that most regular expression operations are available as module-level functions and methods on [compiled regular expressions](#). The functions are shortcuts that don't require you to compile a regex object first, but miss some fine-tuning parameters.

**See also:** The third-party [regex](#) module, which has an API compatible with the standard library [re](#) module, but offers additional functionality and a more thorough Unicode support.

## Regular Expression Syntax





## OCR

- One powerful data collection method, yet producing dirty data
- <https://cloud.google.com/vision>
- Free alternative: tesseract

# Data Storing

# CSV or text-based files

- [https://en.wikipedia.org/wiki/Comma-separated\\_values](https://en.wikipedia.org/wiki/Comma-separated_values)
- Easiest way to store data
- Some other text-based files
  - Fixed width file
  - Tab separated values (TSV)

- Adjacent fields must be separated by a single comma. However, comma is used as a **decimal separator**, semicolon, TAB, or other

```
1997,Ford,E350
```

- Any field *may* be *quoted* (that is, enclosed within double-quote c

```
"1997","Ford","E350"
```

- Fields with embedded commas or double-quote characters must

```
1997,Ford,E350,"Super, luxurious truck"
```

- Each of the embedded double-quote characters must be represe

```
1997,Ford,E350,"Super, ""luxurious"" truck"
```

- Fields with embedded line breaks must be quoted (however, mar

```
1997,Ford,E350,"Go get one now  
they are going fast"
```



[Home](#) [About](#) [Documentation](#) [Download](#) [License](#) [Support](#) [Purchase](#)

## What Is SQLite?

SQLite is a C-language library that implements a [small, fast, self-contained, high-reliability, full-featured](#), SQL database engine. SQLite is the [most used](#) database engine in the world. SQLite is built into all mobile phones and most computers and comes bundled inside countless other applications that people use every day. [More Information...](#)

The SQLite [file format](#) is stable, cross-platform, and backwards compatible and the developers pledge to keep it that way [through the year 2050](#). SQLite database files are commonly used as containers to transfer rich content between systems [\[1\]](#) [\[2\]](#) [\[3\]](#) and as a long-term archival format for data [\[4\]](#). There are over 1 trillion (1e12) SQLite databases in active use [\[5\]](#).

SQLite [source code](#) is in the [public-domain](#) and is free to everyone to use for any purpose.

## Latest Release

[Version 3.36.0](#) (2021-06-18). [Download](#) [Prior Releases](#)

### Comr

- Features
- When to
- Getting S
- Prior Rel
- SQL Syn
  - Pra
  - SQL
  - Dat
  - Agg
  - Win
  - Mat
  - JSO
- C/C++ I
  - Intr
  - List
- The TCL
- Quirks a
- Frequent
- Commit
- Bugs
- News

Python » English

Data Persistence »

3.8.12 » Documentation » The Python Standard Library » previous | next | modules | index

Quick search

Go

## Table of Contents

sqlite3 — DB-API 2.0 interface for SQLite databases

- Module functions and constants
- Connection Objects
- Cursor Objects
- Row Objects
- Exceptions
- SQLite and Python types
  - Introduction
  - Using adapters to store additional Python types in SQLite databases
    - Letting your object adapt itself
    - Registering an adapter callable
  - Converting SQLite values to custom Python types
  - Default adapters and converters
- Controlling Transactions
- Using sqlite3 efficiently
  - Using shortcut methods
  - Accessing columns by name instead of by index
  - Using the connection as a context manager

## Previous topic

dbm — Interfaces to Unix "databases"

# sqlite3 — DB-API 2.0 interface for SQLite databases

Source code: [Lib/sqlite3/](#)

SQLite is a C library that provides a lightweight disk-based database that doesn't require a separate server process and allows accessing the database using a nonstandard variant of the SQL query language. Some applications can use SQLite for internal data storage. It's also possible to prototype an application using SQLite and then port the code to a larger database such as PostgreSQL or Oracle.

The sqlite3 module was written by Gerhard Häring. It provides a SQL interface compliant with the DB-API 2.0 specification described by [PEP 249](#).

To use the module, you must first create a [Connection](#) object that represents the database. Here the data will be stored in the [example.db](#) file:

```
import sqlite3
con = sqlite3.connect('example.db')
```

You can also supply the special name `:memory:` to create a database in RAM.

Once you have a [Connection](#), you can create a [Cursor](#) object and call its [execute\(\)](#) method to perform SQL commands:

```
cur = con.cursor()

# Create table
cur.execute('CREATE TABLE stocks
            (date text, trans text, symbol text, qty real, price real)')

# Insert a row of data
cur.execute("INSERT INTO stocks VALUES ('2006-01-05', 'BUY', 'RHAT', 100, 35.45)")

# Save (commit) the changes
con.commit()

# We can also close the connection if we are done with it.
# Just be sure any changes have been committed or they will be lost.
con.close()
```

# 🔑 RDBMS / NoSQL

- RDBMS
  - [https://en.wikipedia.org/wiki/Relational\\_database#RDBMS](https://en.wikipedia.org/wiki/Relational_database#RDBMS)
  - MySQL, PostgreSQL, SQLite
- NoSQL
  - <https://en.wikipedia.org/wiki/NoSQL>
  - MongoDB, HBase, Redis

# 📌 Data Wrangling with Pandas

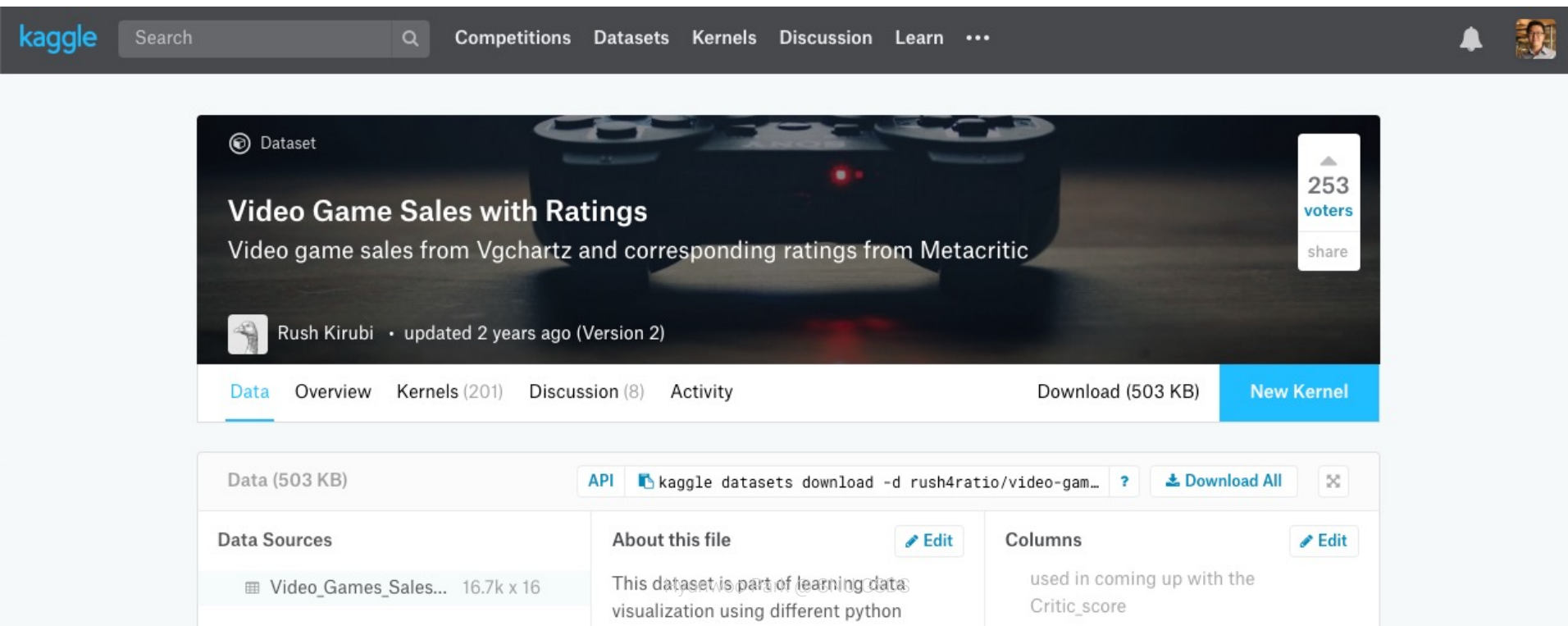
- <https://www.tomasbeuzen.com/python-programming-for-data-science/chapters/chapter7-pandas.html>
- <https://www.tomasbeuzen.com/python-programming-for-data-science/chapters/chapter8-wrangling-basics.html>
- <https://www.tomasbeuzen.com/python-programming-for-data-science/chapters/chapter9-wrangling-advanced.html>
- List of lists?

# Next Week

# Let's grab a dataset.

- Video Game Sales with Ratings

<https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings>




The screenshot shows the Kaggle website interface. At the top is a dark navigation bar with the Kaggle logo, a search bar, and links for Competitions, Datasets, Kernels, Discussion, and Learn. Below this is the dataset page for 'Video Game Sales with Ratings' by Rush Kirubi. The page features a large header image of a video game controller with the dataset title and description. On the right side of the header, it shows 253 voters and a share button. Below the header is a tabbed interface with 'Data' selected, and buttons for 'Download (503 KB)' and 'New Kernel'. The main content area is divided into three sections: 'Data Sources' showing the dataset file 'Video\_Games\_Sales...' with dimensions 16.7k x 16; 'About this file' with an 'Edit' button and a description stating the dataset is part of a learning data visualization project using Python; and 'Columns' with an 'Edit' button and a list of columns including 'Year', 'Genre', 'Platform', 'Sales', and 'Critic\_score'.

**Kaggle** Search Competitions Datasets Kernels Discussion Learn ...

**Dataset**

## Video Game Sales with Ratings

Video game sales from Vgchartz and corresponding ratings from Metacritic

 Rush Kirubi · updated 2 years ago (Version 2)

**253 voters** share

**Data** Overview Kernels (201) Discussion (8) Activity Download (503 KB) **New Kernel**

**Data (503 KB)** **API** `kaggle datasets download -d rush4ratio/video-gam...` **Download All**

### Data Sources

Video_Games_Sales...	16.7k x 16
----------------------	------------

### About this file

**Edit**

This dataset is part of learning data visualization using different python

### Columns

**Edit**

used in coming up with the Critic\_score



Stata/MP 17.0 - https://www.stata-press.com/data/17/nhanes2d.dta

File Edit Data Graphics Statistics User Window Help

Summaries, tables, and tests

Linear models and related

Binary outcomes

Ordinal outcomes

Categorical outcomes

Count outcomes

Fractional outcomes

Generalized linear models

Choice models

Time series

Multivariate time series

Spatial autoregressive models

Longitudinal/panel data

Multilevel mixed-effects models

Survival analysis

Epidemiology and related

Endogenous covariates

Sample-selection models

Treatment effects

SEM (structural equation modeling)

LCA (latent class analysis)

FMM (finite mixture models)

IRT (item response theory)

Multivariate analysis

Survey data analysis

Lasso

Meta-analysis

Multiple imputation

Nonparametric analysis

Exact statistics

Resampling

Power, precision, and sample size

Bayesian analysis

Postestimation

Other

supported; see [help unicode\\_advice](#).  
 2 billion observations are allowed; see [help obs\\_advice](#).  
 Number of variables is set to 5,000; see [help set\\_maxvar](#).

weight  
 (estimation sample)

omial regression

Number of obs = 10,351  
 Population size = 117,157,513  
 Design df = 31  
 F(1, 31) = 719.98  
 Prob > F = 0.0000

Linearized

Coefficient	std. err.	t	P> t	[95% conf. interval]
.022376	.0008339	26.83	0.000	-.0206752 .0240768
.67223	.0882041	-30.30	0.000	-2.852124 +2.492337
.52637	.	.	.	.
1.1e-18	.	.	.	.

Setup and utilities

Tables

Means, proportions, ratios, totals

Linear models and related

Binary outcomes

Ordinal outcomes

Categorical outcomes

Count outcomes

Fractional outcomes

Survival models

Multilevel mixed-effects models

Endogenous covariates

Sample-selection models

Generalized linear model (GLM)

SEM (structural equation modeling)

LCA (latent class analysis)

FMM (finite mixture models)

IRT (item response theory)

DEFF, MEFF, and other statistics

Resampling

Poisson regression

Negative binomial regression

Generalized negative binomial regression

Zero-inflated Poisson regression

Zero-inflated negative binomial regression

Truncated Poisson regression

Truncated negative binomial regression

Censored Poisson regression

Poisson model with sample selection

Variables

Filter variables here:

Name	Label
sampl	Unique case identifier
strata	Stratum identifier
psu	Primary sampling unit
region	Region
smsa	SMSA type
location	Location (stand office ID)
housize	
sex	
race	
age	
height	

Model by/ff/m Weights SE/Robust Reporting Maximization

Dependent variable: highbp

Independent variables: height weight age age2 female black

☐ Suppress constant term

Parameterization of dispersion

☒ A function of the expected mean ☐ A constant

Options

☒ Exposure variable ☐ Offset variable

Constraints: Manage...

Properties

Variables

Name

Label

Type

Format

Value label

Notes

Data

Frame

Filename

Label

Notes

Variables

Observations 10,351

Size 1,09M

Memory 64M

Sorted by

CAP NUM INS

Class Survey  
Due 9/13  
before class 12:30pm KST

# Example Response

<https://forms.gle/SBsqZpFcAcilYRiP8>

## Class Survey for Data Analysis and Visualization Fall 2021

Your personal information will be used for project group formation. Your responses on Kaggle datasets will be used as a dataset for teaching material.

All questions must be answered before you submit the form.

You must be signed in with your SNU email to access and submit this form.

If you want to change your response, please submit this form again. If you submit multiple responses, only your last response will be used.

hyunwoopark@snu.ac.kr [Switch account](#) Draft saved

Your email will be recorded when you submit this form

**\* Required**

**SNU Email \***  
Your response must end with [@snu.ac.kr](#).

hyunwoopark@snu.ac.kr

**Student Number \***

2002-11992

**Name (in English) \***  
The same name as in eTL. Please use [Last, First] format.

Park, Hyunwoo

**Gender \***

☐ Female

☒ Male

☐ Prefer not to say

☐ Other: \_\_\_\_\_

**Age Group \***

☐ 0 ~ 19

☐ 20 ~ 24

☐ 25 ~ 29

☐ 30 ~ 34

☒ 35 ~ 39

☐ 40 and above

☐ Prefer not to say

**Undergraduate Major (If multiple, separate by semicolon.) \***

Electrical Engineering

**Graduate Major (If multiple, separate by semicolon. If none, say None.) \***

Information Management and Systems; Indust

**Choose languages that you have experience programming more than 100 lines. \***

☒ Python

☒ JavaScript

☒ HTML

☒ CSS

☐ Other: \_\_\_\_\_

**What is your self-evaluated proficiency in Python? \***

1 2 3 4 5 6 7

No experience ☐ ☐ ☐ ☐ ☐ ☒ ☐ Very comfortable

**What is your self-evaluated proficiency in web programming? \***

1 2 3 4 5 6 7

No experience ☐ ☐ ☐ ☐ ☒ ☐ ☐ Very comfortable

**Choose your five favorite Kaggle datasets with weights. \***

Your answer to this question should be five lines. Each line should be URL [space] weight. The weights should sum to 100. Example response is as attached. Your score for this class survey will be deducted if your response does not follow the instructions.

**Choose your five favorite Kaggle datasets with weights. \***

Your answer to this question should be five lines. Each line should be URL [space] weight. The weights should sum to 100. Example response is as attached.

<https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings> 20

<https://www.kaggle.com/neuromusic/avocado-prices> 30

<https://www.kaggle.com/jessical9530/honey-production> 15

<https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009> 5

<https://www.kaggle.com/nickhould/craft-cans> 30

<https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings> 20

<https://www.kaggle.com/neuromusic/avocado-prices> 30

<https://www.kaggle.com/jessical9530/honey-production> 15

<https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009> 5

<https://www.kaggle.com/nickhould/craft-cans> 30

**Submit** [Clear form](#)

Never submit passwords through Google Forms.

This form was created inside of 서울대학교. [Report Abuse](#)

Google Forms

## 📌 Things to do / Reminders

- Class Survey (5%)
  - Due on 9/13 (Tue) before class 2pm KST.
- Review
  - Check out Chrome Developer Tools.
  - Watch Data Wranger and OpenRefine videos.
  - Try regular expression using re.
- eTL boards
  - Q&A Board for questions and answers
  - Discussion Board for resources or ideas to share
- See you Tuesday, 9/14.
  - Univariate Analysis and Visualization
  - Zoom only