Destiny Dump Database

Developers

Brey Rivera – Adam Spera – Tyler Mui – Dylan Alexander

What it does...

Destiny Dump is a database that contains all of the weapons in the popular game Destiny 2.

Every weapon in Destiny has over 10 unique attribute that make them all different.

Due to the amount of weapons in the game, over its ongoing life of over 6 years of active support, the total weapons in the game is now over 1,000.

We wanted to develop this database and interface because there is currently no public database of all this information available, due to the api being very exclusive.

Table of Contents

01

API Queries without API Tokens

Beautiful Soup

Due to the security aspect of a live service game, the content api is very exclusive.

Luckily, some professional services offer quick views of the api data, which we can take advantage of to scrape.

We will be using the python package BEautifulSoup for web scraping.

Scraping the Data

Light.gg () => Destiny Dump

Getting links to each weapon:

www.light.gg/db/items/507038823

As seen above, light.gg stores its weapon entries with unpredictable, unique ids. Luckily, the weapon browsing page (limit 50 weapons per page) has a predictably incrementing link. From this, we can scrape the <a> tags on each page, and collect its href attribute (the unique unpredictable links):

www.light.gg/db/category/1/weapons/?page=4

```
def getWeaponLinks():
  f = open("weaponLinks.txt", "a")

  counter = 1
  while (counter <=  1154):
    print(counter)
    url = 'https://www.light.gg/db/category/1?page='+ str(counter)
    response = requests.get(url)
    soup = BeautifulSoup(response.content, 'lxml')

    for link in soup.find_all('a'):
      if '/db/items/' in link.get('href') and 'compare' not in link.get('href'):
        f.write('https://www.light.gg' + link.get('href') + '\n')

    counter += 1

  f.close()

  lines = open('weaponLinks.txt', 'r').readlines()
  lines_set = set(lines)
  out = open('weaponLinks.txt', 'a')
  for line in lines_set:
    out.write(line)
  lines.close()

  return
```

This scraping gives us a text file list of every unique link to every weapon on light.gg.

Getting data from each weapon:

Now that we have links to every weapon, we can start scraping for each weapon's stats.

*Note after data is compiled, the file type must be changed from .txt to .json.

Code explained:

1. Iterate through every link in file.
2. Get the lxml and begin parsing it.
3. Check specific cases where errors could be entered into our data.
4. Find all of each type of tag that we know contains the data, like how each weapons name is the only <h2> tag on the page.
5. Add all the data collected to an object and add it to the output text file.

```
def getWeaponStats():
  weaponLinksFile = open('weaponLinks.txt', 'r')
  links = weaponLinksFile.readlines()

  weaponStatsFile = open('weaponStats-Dec6.txt', 'a')
  weaponStatsFile.write("[\n")

  tempDict = {}
  counter = 0

  for link in links:

    print(str(counter) + ' ------------------------------------------------------------------------------')
    counter += 1

    response = requests.get(link.strip())
    soup = BeautifulSoup(response.content, 'lxml')

    if len(soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')) >= 2:
      if soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')[2].strip() == 'Weapon Ornament':
        continue

    tempDict['weapon_id'] = link.split('items/')[1].strip()
    tempDict['Name'] = soup.find_all('h2')[0].text.strip()
    tempDict['Rarity'] = soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')[0].strip()

    if len(soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')) == 4:
      tempDict['Class'] = soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')[1].strip()
      tempDict['Element'] = soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')[2].strip()
      tempDict['Type'] = soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')[3].strip()
    else:
      tempDict['Class'] = 'Any'
      tempDict['Element'] = soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')[1].strip()
      tempDict['Type'] = soup.find_all('span', class_="weapon-type")[0].text.strip().split('/')[2].strip()

    for linkTable in soup.find_all('table', class_="stat-visualizer"):
      for row in linkTable.find_all('tr'):
        tds = row.find_all('td')
        tempDict[tds[0].text.strip()] = tds[-1].text.strip()

    perksArray = []
    for img in soup.find_all("img", {"class": "mod"}):
      if 'Ornament' not in img.get('alt') and img.get('alt').isascii():
        perksArray.append(img.get('alt'))

    tempDict['Perks'] = perksArray
    perksArray = []

    print(str(tempDict))
    weaponStatsFile.write(str(tempDict) + ",\n")
    tempDict = {}

  weaponStatsFile.write("]")

  return
```

We've stepped into a war with scraped data

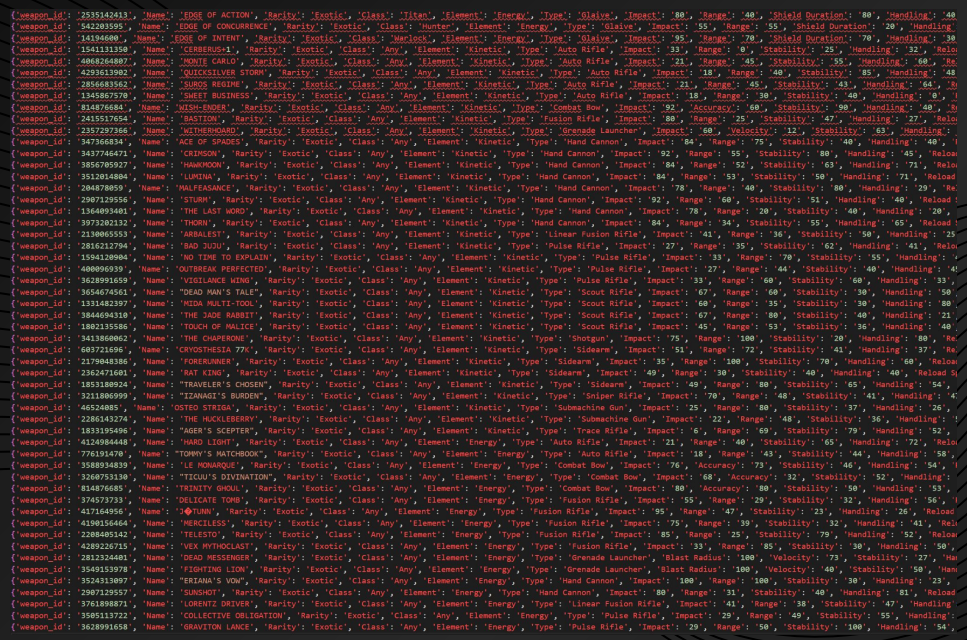
Scraped Data Before Initial Parsing


```
weapon_id,Name,Rarity,Class,Element,Type,Impact,Range,Shield Duration,Handling,Reload Speed,Aim Assistance,Inventory Size,Airborne Effectiveness,Rounds Per Minute,Charge Time,Magazine,Perks/0,Perks/1,Perks/2,Perks/3,Perks/4,Perks/5,Perks/6,Perks/7,Perks/8,Perks/9,Perks/10,Perks/11,
2535142413,EDGE OF ACTION,Exotic,Titan,Energy,Glaive,80,40,80,40,25,30,30,7,55,80,4,Edge of Action,Low-Impedance Windings,Appended Mag,Remote Shield,Edge of Action,Low-Impedance Windings,Ballistic Tuning,Tempered Truss Rod,Supercooled Accelerator,Auxiliary Reserves,Lightweight Emit
542203595,EDGE OF CONCURRENCE,Exotic,Hunter,Energy,Glaive,55,55,20,55,80,75,70,9,80,20,6,Edge of Concurrence,Tempered Truss Rod,Accurized Rounds,Lightning Seeker,Edge of Concurrence,Low-Impedance Windings,Ballistic Tuning,Tempered Truss Rod,Supercooled Accelerator,Auxiliary Reserve
14194600,EDGE OF INTENT,Exotic,Warlock,Energy,Glaive,95,70,70,30,45,50,50,12,45,70,3,Edge of Intent,Supercooled Accelerator,Alloy Magazine,Restorative Turret,Edge of Intent,Low-Impedance Windings,Ballistic Tuning,Tempered Truss Rod,Supercooled Accelerator,Auxiliary Reserves,Lightwe
1541131350,CERBERUS+1,Exotic,Any,Kinetic,Auto Rifle,33,0,,32,35,60,59,16,360,,31,Four-Headed Dog,Rifled Barrel,Fluted Barrel,Full Bore,Spread Shot Package,Four-Headed Mongrel,,,,,,,,,,,,,,,,,25,16,86,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
4068264807,MONTE CARLO,Exotic,Any,Kinetic,Auto Rifle,21,45,,60,79,50,52,29,600,,43,Monte Carlo Method,Fluted Barrel,High-Caliber Rounds,Markov Chain,Composite Stock,What If,Merry Maker,Western Front,,,,,,,,,,,,,,,55,15,80,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
4293613902,QUICKSILVER STORM,Exotic,Any,Kinetic,Auto Rifle,18,40,,48,54,83,59,26,720,,51,Rocket Tracers,Corkscrew Rifling,High-Caliber Rounds,Grenade Chaser,Hand-Laid Stock,,,,,,,,,,,,,,,,,,85,16,88,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2856683562,SUROS REGIME,Exotic,Any,Kinetic,Auto Rifle,21,45,,64,63,68,0,31,600,,25,SUROS Legacy,Hammer-Forged Rifling,Tactical Mag,Spinning Up,Dual Speed Receiver,Hand-Laid Stock,Coup de Main,Gray Nitrite,SUROS Chrome,Standard Bearer,,,,,,,,,,,,,43,16,50,,,,,,,,,,,,,,,,,,,,,,,,,,,,
1345867570,SWEET BUSINESS,Exotic,Any,Kinetic,Auto Rifle,18,30,,0,0,80,80,32,360,,45,Payday,Polygonal Rifling,Armor-Piercing Rounds,Business Time,Composite Stock,Down to Business,Go About Your Business,Corporate Sponsorship,,,,,,,,,,,,,,,40,16,100,,,,,,,,,,,,,,,,,,,,,,,,,,,,
814876684,WISH-ENDER,Exotic,Any,Kinetic,Combat Bow,92,,,40,30,51,50,19,,,,Queen's Wrath,High Tension String,Anti-Taken Fletching,Broadhead,Author of Devastation,,,,,,,,,,,,,,,,,,90,18,60,60,780,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2415517654,BASTION,Exotic,Any,Kinetic,Fusion Rifle,80,25,,27,24,65,30,3,,780,5,Saint's Fists,Hammer-Forged Rifling,Liquid Coils,Breakthrough,Composite Stock,Bloodline Memorial,,,,,,,,,,,,,,,,47,15,75,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2357297366,WITHERHOARD,Exotic,Any,Kinetic,Grenade Launcher,60,,,48,33,84,46,6,90,,6,Primeval's Torment,Countermass,Black Powder,Break the Bank,Composite Stock,White Collar Crime,Drowned in Strife,Cedar and Ash,Splendid Vidua,,,,,,,,,,,,,63,12,96,,12,,,,,,,,,,,,,,,,,,,,,,,,,,,,
347366834,ACE OF SPADES,Exotic,Any,Kinetic,Hand Cannon,84,75,,40,42,70,56,14,140,,13,Memento Mori,Corkscrew Rifling,High-Caliber Rounds,Firefly,Smooth Grip,Big Blind,All In,The Vanguard Dare,Last Hand,Salute to the Colonel,,,,,,,,,,,,40,14,100,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
3437746471,CRIMSON,Exotic,Any,Kinetic,Hand Cannon,92,55,,45,60,75,100,15,415,,14,Banned Weapon,Smallbore,Accurized Rounds,Cruel Remedy,Heavy Grip,Silver Bullet,Iridescent Death,Black Death,,,,,,,,,,,,,,80,14,55,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
3856705927,HAWKMOON,Exotic,Any,Kinetic,Hand Cannon,84,52,,71,59,93,52,26,140,,8,Paracausal Shot,Corkscrew Rifling,Alloy Magazine,Transformative,Smooth Grip,Paracausal Shot,Corkscrew Rifling,Extended Barrel,Fluted Barrel,Hammer-Forged Rifling,Polygonal Rifling,Smallbore,Arrowhead Br
3512014804,LUMINA,Exotic,Any,Kinetic,Hand Cannon,84,53,,71,61,88,58,27,140,,13,Noble Rounds,Chambered Compensator,Accurized Rounds,Blessing of the Sky,Polymer Grip,Lucid Melody,,,,,,,,,,,,,,,,50,14,95,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
204878059,MALFEASANCE,Exotic,Any,Kinetic,Hand Cannon,78,40,,29,80,50,57,53,180,,14,Explosive Shadow,Corkscrew Rifling,Extended Mag,Taken Predator,Heavy Grip,Aim to Misbehave,Red String of Fate,The Pink Mist,,,,,,,,,,,,,,,80,14,98,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2907129556,STURM,Exotic,Any,Kinetic,Hand Cannon,92,60,,40,50,50,30,12,120,,9,Accomplice,Extended Barrel,Extended Mag,Storm and Stress,Combat Grip,Symbiosis,Sudden Squall,Dream of a New World,,,,,,,,,,,,,,51,14,80,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
1364093401,THE LAST WORD,Exotic,Any,Kinetic,Hand Cannon,78,20,,20,80,40,40,19,225,,8,Fan Fire,Corkscrew Rifling,Accurized Rounds,Hip-Fire Grip,Textured Grip,Laconic,Heated Exchange,The Stagecoach,End of an Era,,,,,,,,,,,,,40,11,100,,,,,,,,,,,,,,,,,,,,,,,,,,,,
3973202132,THORN,Exotic,Any,Kinetic,Hand Cannon,84,34,,65,40,85,60,21,140,,11,Mark of the Devourer,Corkscrew Rifling,Accurized Rounds,Soul Devourer,Textured Grip,Wishes of Sorrow,Heretic Robe,For the King,,,,,,,,,,,,,,55,14,100,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2130065553,ARBALEST,Exotic,Any,Kinetic,Linear Fusion Rifle,41,36,,25,28,31,34,12,,533,5,Compounding Force,Extended Barrel,Projection Fuse,Disruption Break,Composite Stock,Barren Wastes,Violent Exorcism,Electromagnetic Execution,,,,,,,,,,,,,,50,25,77,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2816212794,BAD JUJU,Exotic,Any,Kinetic,Pulse Rifle,27,35,,41,47,60,15,28,450,,27,String of Curses,Hammer-Forged Rifling,Steady Rounds,Hip-Fire Grip,Short-Action Stock,Aurelian Decree,Beast Magic,,,,,,,,,,,,,,,62,17,85,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
1594120904,NO TIME TO EXPLAIN,Exotic,Any,Kinetic,Pulse Rifle,33,70,,42,60,40,57,24,340,,24,Rewind Again,Fluted Barrel,Accurized Rounds,Time-Slip,Feeding Frenzy,Frozen In Time,Golden Days,,,,,,,,,,,,,,,55,17,73,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
400096939,OUTBREAK PERFECTED,Exotic,Any,Kinetic,Pulse Rifle,27,44,,45,45,65,54,29,450,,34,The Corruption Spreads,Extended Barrel,Accurized Rounds,Outlaw,Parasitism,Patient Zero,Pandemic,Mercurial Affliction,Matterscourge,,,,,,,,,,,,,40,17,80,,,,,,,,,,,,,,,,,,,,,,,,,,,,
3628991659,VIGILANCE WING,Exotic,Any,Kinetic,Pulse Rifle,33,60,,33,40,60,55,21,530,,30,Harsh Truths,Corkscrew Rifling,Alloy Magazine,Last Stand,Composite Stock,Book of the Dead,Heaven's Vigil,Vigil for Saint-14,,,,,,,,,,,,,,60,17,79,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
3654674561,DEAD MAN'S TALE,Exotic,Any,Kinetic,Scout Rifle,67,60,,50,50,60,55,17,120,,14,Cranial Spike,Fluted Barrel,Accurized Rounds,Transformative,Hand-Laid Stock,Cranial Spike,Chambered Compensator,Corkscrew Rifling,Fluted Barrel,Polygonal Rifling,Smallbore,Hammer-Forged Rifling,
1331482397,MIDA MULTI-TOOL,Exotic,Any,Kinetic,Scout Rifle,60,35,,80,80,100,60,25,200,,16,MIDA Multi-Tool,Corkscrew Rifling,High-Caliber Rounds,MIDA Radar,Hand-Laid Stock,Titanium Alloy,MIDA Tactical,Desert Camo,Digital Ops,,,,,,,,,,,,,,30,20,80,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
3844694310,THE JADE RABBIT,Exotic,Any,Kinetic,Scout Rifle,67,80,,21,54,60,30,29,150,,10,The Fate of All Fools,Polygonal Rifling,High-Caliber Rounds,Zen Moment,Jade Jester,Raging Lepus,Qingming Offering,,,,,,,,,,,,,,,40,20,100,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
1802135586,TOUCH OF MALICE,Exotic,Any,Kinetic,Scout Rifle,45,53,,40,46,40,58,25,260,,11,Touch of Malice,Polygonal Rifling,Flared Magwell,Charged with Blight,Hand-Laid Stock,Threatening Breath,,,,,,,,,,,,,,,,36,20,85,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
3413860062,THE CHAPERONE,Exotic,Any,Kinetic,Shotgun,75,100,,80,30,70,45,1,70,,6,Precision Slug,Hammer-Forged Rifling,Accurized Rounds,The Roadborn,Hand-Laid Stock,Guardian Angel,The Panama Ravine,Ain't My First Rodeo,The Queensguard,,,,,,,,,,,,,20,12,50,,,,,,,,,,,,,,,,,,,,,,,,,,,,
603721696,CRYOSTHESIA 77K,Exotic,Any,Kinetic,Sidearm,51,72,,37,46,93,50,28,260,,12,LN2 Burst,Polygonal Rifling,Particle Repeater,Liquid Cooling,Polymer Grip,Deletion Protocol,,,,,,,,,,,,,,,,,41,12,92,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2179048386,FORERUNNER,Exotic,Any,Kinetic,Sidearm,35,100,,60,80,60,56,30,200,,15,Full Stop,Smallbore,Flared Magwell,Pace Yourself,Smooth Grip,BrayTech M6,Gilded Honors,,,,,,,,,,,,,,,70,20,96,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2362471601,RAT KING,Exotic,Any,Kinetic,Sidearm,49,30,,40,40,60,80,31,300,,15,Rat Pack,Smallbore,Tactical Mag,Vermin,Smooth Grip,Black Plague,Catacombs,Climate Change,,,,,,,,,,,,,,40,12,80,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
1853180924,TRAVELER'S CHOSEN,Exotic,Any,Kinetic,Sidearm,49,80,,54,40,80,44,35,300,,15,Gathering Light,QuickDot SAS,Armor-Piercing Rounds,Gift of the Traveler,Short-Action Stock,Sand and Sun,Conscripted,Dream of the Sky,,,,,,,,,,,,,,65,12,85,,,,,,,,,,,,,,,,,,,,,,,,,,,,
3211806999,IZANAGI'S BURDEN,Exotic,Any,Kinetic,Sniper Rifle,70,48,,47,40,62,60,6,90,,4,Honed Edge,Chambered Compensator,Accurized Rounds,No Distractions,Composite Stock,Ronin,Father of Islands,Worldshaper,,,,,,,,,,,,,,41,45,73,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
46524085,OSTEO STRIGA,Exotic,Any,Kinetic,Submachine Gun,25,80,,26,18,55,30,27,600,,27,Screaming Swarm,Toxic Overload,Screaming Swarm,Polygonal Rifling,Chambered Compensator,Full Bore,Corkscrew Rifling,Fluted Barrel,Extended Barrel,Hammer-Forged Rifling,Arrowhead Brake,Smallbore,Ste
2286143274,THE HUCKLEBERRY,Exotic,Any,Kinetic,Submachine Gun,22,48,,56,0,49,100,33,750,,37,Ride the Bull,Fluted Barrel,Ricochet Rounds,Rampage,Short-Action Stock,The Devil's Work,Snakebite,,,,,,,,,,,,,,,36,15,100,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
1833195496,AGER'S SCEPTER,Exotic,Any,Kinetic,Trace Rifle,6,69,,52,55,100,44,15,1000,,97,Ager's Call,Arrowhead Brake,Particle Repeater,Rega's Refrain,Hand-Laid Stock,Symbol of Rule,Springtime Scales,Cold Cheer,,,,,,,,,,,,,,79,16,96,,,,,,,,,,,,,,,,,,,,,,,,,,,,
4124984448,HARD LIGHT,Exotic,Any,Energy,Auto Rifle,21,40,,72,64,100,40,31,600,,49,Volatile Light,Polygonal Rifling,Alloy Magazine,The Fundamentals,Composite Stock,The Future is Chrome,Lethal System,Brumal Dawn,Foundational Structure,,,,,,,,,,,,,65,16,100,,,,,,,,,,,,,,,,,,,,,,,,,,,,
```

We've stepped into a war with scraped data

The .txt to .json conversion process:

Start by changing the file type from txt to json, easy enough... but then... the errors come...



Next step is to use regex on VS Codes find and replace feature to replace all the single quotes with double quotes.

AFTER PARSING


```
{
  "weapon_id": "2535142413",
  "Name": "EDGE OF ACTION",
  "Rarity": "Exotic",
  "Class": "Titan",
  "Element": "Energy",
  "Type": "Glaive",
  "Impact": "80",
  "Range": "40",
  "Shield Duration": "80",
  "Handling": "40",
  "Reload Speed": "25",
  "Aim Assistance": "30",
  "Inventory Size": "30",
  "Airborne Effectiveness": "7",
  "Rounds Per Minute": "55",
  "Charge Time": "80",
  "Magazine": "4",
  "Perks": [
    "Edge of Action",
    "Low-Impedance Windings",
    "Appended Mag",
    "Remote Shield",
    "Edge of Action",
    "Low-Impedance Windings",
    "Ballistic Tuning",
    "Tempered Truss Rod",
    "Supercooled Accelerator",
    "Auxiliary Reserves",
    "Lightweight Emitter",
    "Alloy Magazine",
    "Extended Mag",
    "Appended Mag",
    "Accurized Rounds",
    "Swap Mag",
    "Light Mag",
    "Remote Shield",
    "Hand-Laid Stock",
    "Composite Stock",
    "Fitted Stock",
    "Short-Action Stock"
  ]
},
```

02

Oh how we love Python...

SQLITE3

Due to some technical issues with importing data in Java, we decided to make a DBMS utilizing Python :)

DBMS.py

Creating and configuring the database.
TLDR: lots of pandas and sqlite3!

```python
# insert weapons table data
for index, row in weapons_df.iterrows():
  record = (row['weapon_id'], row['Name'], row['Rarity'], row['Class'], row['Element'], row['Type'])
  insert_into_weapons(conn, record)

# insert weapon perk data
for index, row in weapons_df.iterrows():
  id = row['weapon_id']
  for perk in [*set(row['Perks'])]:
    record = (id, perk)
    insert_into_perks(conn, record)

# insert weapon stat data
for index, row in weapons_df.iterrows():
    record = (row['weapon_id'], row['Impact'], row['Range'], row['Shield Duration'], row['Handling'],
row['Reload Speed'], row['Aim Assistance'], row['Inventory Size'], row['Airborne Effectiveness'],
row['Rounds Per Minute'], row['Charge Time'], row['Magazine'], row['Stability'], row['Zoom'],
row['Recoil'], row['Accuracy'], row['Draw Time'], row['Velocity'], row['Blast Radius'], row['Swing
Speed'], row['Guard Efficiency'], row['Guard Resistance'], row['Charge Rate'], row['Ammo Capacity'])
  insert_into_stats(conn, record)
```

```python
import sqlite3
import json
import pandas as pd

"""
this file creates a database called destinyWeapons.db
"""

def create_connection(db_file):
  conn = None
  try:
    conn = sqlite3.connect(db_file)
    conn.execute('PRAGMA foreign_keys = ON;')
    return conn
  except Error as e:
    print(e)
  return conn

def create_table(conn, create_table_sql):
  try:
    c = conn.cursor()
    c.execute(create_table_sql)
  except Error as e:
    print(e)

def insert_into_weapons(conn, record):
  sql = """INSERT INTO WEAPONS(weapon_id, Name, Rarity, Class, Element, Type) VALUES(?,?,?,?,?,?)"""
  try:
    c = conn.cursor()
    c.execute(sql, record)
    conn.commit()
  except Error as e:
    print(e)

def insert_into_perks(conn, record):
  sql = """INSERT INTO PERKS(weapon_id, Perk) VALUES(?,?)"""
  try:
    c = conn.cursor()
    c.execute(sql, record)
    conn.commit()
  except Error as e:
    print(e)

def insert_into_stats(conn, record):
  sql = """INSERT INTO STATS VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)"""
  try:
    c = conn.cursor()
    c.execute(sql, record)
    conn.commit()
  except Error as e:
    print(e)
```

DBMS.py

Table Breakdown:

1. **WEAPONS**
   a. The weapons table holds basic information about the weapon
      i. Primary Key - weapon_id

2. **PERKS**
   a. The perks table holds perks for each weapon
      i. Primary Key - (weapon_id, Perk)
      ii. Foreign Key - weapon_id

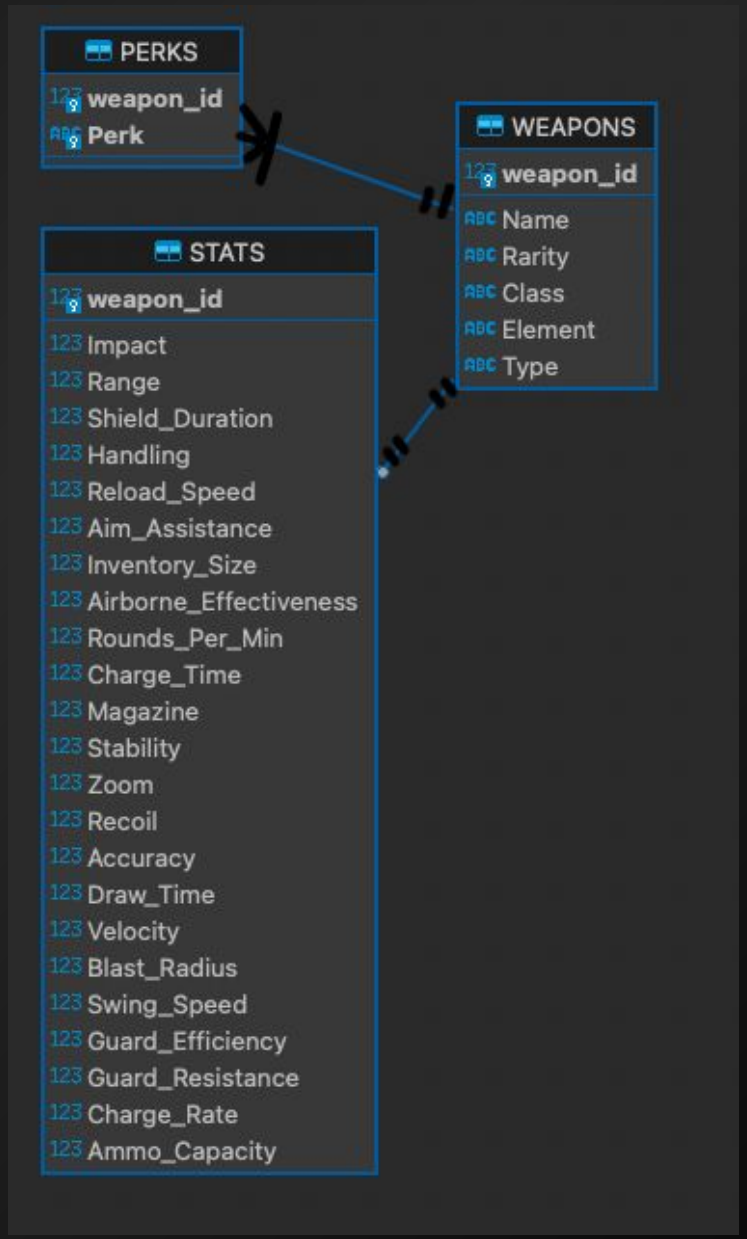
3. **STATS**
   a. The stats table holds one record for all stats a weapon holds
      i. Primary Key - weapon_id
      ii. Foreign Key - weapon_id

THE SQL TABLES

```
createWeaponsTable = """
      CREATE TABLE IF NOT EXISTS WEAPONS (
        weapon_id INTEGER NOT NULL,
        Name VARCHAR(255) NOT NULL,
        Rarity VARCHAR(255) NOT NULL,
        Class VARCHAR(255) NOT NULL,
        Element VARCHAR(255) NOT NULL,
        Type VARCHAR(255) NOT NULL,

        PRIMARY KEY (weapon_id)
      );
      """

createPerksTable = """
      CREATE TABLE IF NOT EXISTS PERKS (
        weapon_id INTEGER NOT NULL,
        Perk VARCHAR(55) NOT NULL,

        PRIMARY KEY (weapon_id, Perk),
        FOREIGN KEY (weapon_id) REFERENCES WEAPONS(weapon_id)
          ON UPDATE CASCADE
          ON DELETE CASCADE
      );
      """

createStatsTable = """
      CREATE TABLE IF NOT EXISTS STATS (
        weapon_id INTEGER NOT NULL,
        Impact INTEGER,
        Range INTEGER,
        Shield_Duration INTEGER,
        Handling INTEGER,
        Reload_Speed INTEGER,
        Aim_Assistance INTEGER,
        Inventory_Size INTEGER,
        Airborne_Effectiveness INTEGER,
        Rounds_Per_Min INTEGER,
        Charge_Time INTEGER,
        Magazine INTEGER,
        Stability INTEGER,
        Zoom INTEGER,
        Recoil INTEGER,
        Accuracy INTEGER,
        Draw_Time INTEGER,
        Velocity INTEGER,
        Blast_Radius INTEGER,
        Swing_Speed INTEGER,
        Guard_Efficiency INTEGER,
        Guard_Resistance INTEGER,
        Charge_Rate INTEGER,
        Ammo_Capacity INTEGER,

        PRIMARY KEY (weapon_id),
        FOREIGN KEY (weapon_id) REFERENCES WEAPONS(weapon_id)
          ON UPDATE CASCADE
          ON DELETE CASCADE
      );
      """
```

ER Diagram

** Disclaimer, made by a Data Science undergraduate, not professional graphic designer*



DBMS_interact.py

The code for database interaction is around 340 lines, more can be seen on the GitHub repository. This snippet shows the main driver in **DBMS_interact.py**. Listed are the options for interaction:

1. Select Weapon
2. Update Weapon
3. Insert Weapon
4. Delete Weapon
5. View All Weapons
6. View All Weapon Types
7. View All Weapons of Specific Type
8. List top 10 weapons with best value for selected stat

```python
from DBMS import create_connection
import inquirer
import re
import sys

def menu():
  questions = [
      inquirer.List('initialChoice', message="Welcome to Destiny Dump! Select from one of the options below",
choices=['Select Weapon', 'Update Weapon', 'Delete Weapon', 'Insert Weapon', 'View All Weapons', 'View All
Weapon Types', 'View All Weapons of Specific Type', 'List top 10 weapons with best value for selected stat'])
    ]
  answers = inquirer.prompt(questions)

  if answers['initialChoice'] == 'Select Weapon':
    # shows weapon name, stats, and available perks
    weapon_name = input('Enter the name of the weapon you wish to view: ')
    select_weapon(conn, weapon_name)
    restartOption()
  elif answers['initialChoice'] == 'Update Weapon':
    # update any aspect of a weapon
    weapon_name = input('Which weapon would you like to update? (Enter weapon name): ').upper()
    update_weapon(conn, weapon_name)
    restartOption()
  elif answers['initialChoice'] == 'Delete Weapon':
    # delete any weapon
    weapon_name = input('What is the name of the weapon you wish to delete? ')
    delete_weapon(conn, weapon_name)
    restartOption()
  elif answers['initialChoice'] == 'Insert Weapon':
    # insert any weapon
    insert_weapon(conn)
    restartOption()
  elif answers['initialChoice'] == 'View All Weapons':
    # displays all weapons
    view_all_weapons(conn)
    restartOption()
  elif answers['initialChoice'] == 'View All Weapon Types':
    # displays all the weapon types that are in the database
    view_all_types(conn)
    restartOption()
  elif answers['initialChoice'] == 'View All Weapons of Specific Type':
    # displays all the weapon types that are in the database
    w_type = input('What type of weapon do you wish to view? ')
    view_all_weapons_of_type(conn, w_type)
    restartOption()
  elif answers['initialChoice'] == 'List top 10 weapons with best value for selected stat':
    print() # buffer
    # print stats
    for stat in STATS_COLS[1:]:
      print(stat)
    s_stat = input('\nWhich stat would you like to view the top 10 weapons for? ')
    list_weapon_for_stat(conn, s_stat)
    restartOption()
```

03

UX, UI, and U-everything else

Inquirer

We use the python package Inquirer to allow an easy interface through a console program.

Inquirer provides an easy to read, and understandable interface to make querying easier than ever.

Plus, pretty colors!

Inquirer them all!

Inquire what it looks like? You're in luck!

```
[?] Welcome to Destiny Dump! Select from one of the options below: Select Weapon
 > Select Weapon
   Update Weapon
   Delete Weapon
   Insert Weapon
   View All Weapons
   View All Weapon Types
   View All Weapons of Specific Type
   List top 10 weapons with best value for selected stat
```

Oh no, our table, its broken!

Bugs Encountered:

When working with foreign keys in SQLITE3, make sure to enable foreign keys each time a connection is made.

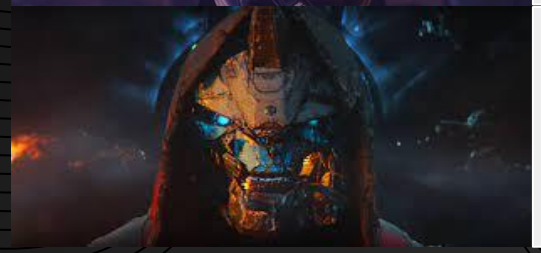
Special thanks to the_triangle_dude on reddit!

DEMO

Click on box to play video

Memes made during production!

Excited to work on a final project that involves coding



15 minutes into web scraping

Thank You