**Class #16: Data Analysis Project using NumPy and Pandas**

1. Project: Download a CSV file from Kaggle and perform analysis using NumPy and Pandas. **Assignment #16:**

<https://www.kaggle.com/datasets/chicago/chicago-citywide-payroll-data/data>

**Data Overview (ডেটার সার্বিক তথ্য):**

**১. DataFrame**: এই ডেটাটি একটি **pandas DataFrame** যেটি ৩২,৬৫৮টি রেকর্ড এবং ৮টি কলাম ধারণ করে। এর মানে হল, ডেটাটিতে ৩২,৬৫৮টি সারি (rows) এবং ৮টি ভিন্ন ভিন্ন কলাম রয়েছে।

**Columns**: ডেটার ৮টি কলাম রয়েছে:

* 1. **Name**: কর্মচারীর নাম
  2. **Job Titles**: কর্মচারীর পদবী
  3. **Department**: কর্মচারী যে বিভাগের সাথে যুক্ত
  4. **Full or Part-Time**: কর্মচারী পূর্ণকালীন (Full-time) না অংশকালীন (Part-time)
  5. **Salary or Hourly**: কর্মচারীর বেতন পদ্ধতি (Salary বা Hourly)
  6. **Typical Hours**: কর্মচারীর সাধারণ কাজের সময় (এটি কিছু ক্ষেত্রে না থাকতে পারে)
  7. **Annual Salary**: বার্ষিক বেতন (Salary পদ্ধতির জন্য)
  8. **Hourly Rate**: প্রতি ঘণ্টায় বেতন (Hourly পদ্ধতির জন্য)

**২. Missing Data (নাল বা অনুপস্থিত তথ্য):**

* **Typical Hours** কলামে **৭,৮৮৩টি** মান রয়েছে, অর্থাৎ অনেক রেকর্ডে এই কলামে ডেটা অনুপস্থিত।
* **Annual Salary** কলামে **২৪,৭৭৫টি** মান রয়েছে, মানে কিছু রেকর্ডে বার্ষিক বেতন নেই। এগুলি সম্ভবত **Hourly Rate** পদ্ধতির জন্য সেগুলিতে নেই।
* **Hourly Rate** কলামে **৭,৮৮৩টি** মান রয়েছে, অর্থাৎ অনেক রেকর্ডে ঘণ্টায় বেতন নেই। এ ক্ষেত্রেও সম্ভবত **Salary** পদ্ধতির জন্য এই কলামে ডেটা নেই।

এটি সম্ভবত কারণ যে, **Salary** এবং **Hourly Rate** দুটি আলাদা বেতন পদ্ধতি। যেখানে **Salary** পদ্ধতিতে **Annual Salary** দেওয়া হয়, এবং **Hourly Rate** পদ্ধতিতে **Hourly Rate** দেওয়া হয়, কিন্তু একসাথে দুইটি তথ্য একটি রেকর্ডে থাকতে পারে না। তাই কিছু কলামে ডেটা অনুপস্থিত (null)।

**৩. Data Types (ডেটার ধরন):**

* **float64**: Typical Hours কলামে কিছু মান রয়েছে, কিন্তু অনেক জায়গায় এটি অনুপস্থিত (NaN), তাই এটি float64 ডেটাটাইপ ধারণ করে।
* **object**: বাকি সব কলামে ডেটা **object** টাইপে রয়েছে, কারণ সেখানে স্ট্রিং (টেক্সট) ধরনের মান রয়েছে (যেমন, নাম, পদবী, বিভাগ, বেতন ইত্যাদি)।

**৪. Memory Usage (মেমরি ব্যবহারের তথ্য):**

* এই ডেটাফ্রেমের মোট মেমরি ব্যবহার প্রায় **২.০ MB**। এটি মোট রেকর্ডের সংখ্যা এবং ডেটা টাইপ অনুযায়ী স্বাভাবিক।

**৫. Output-এর কারণ:**

* **Missing Data**: কিছু কলামে (যেমন Typical Hours, Annual Salary, Hourly Rate) অনুপস্থিত (missing) তথ্য রয়েছে, যা মূলত কর্মচারীর বেতন পদ্ধতির কারণে ঘটে। **Salary** পদ্ধতিতে **Annual Salary** থাকে, কিন্তু **Hourly Rate** থাকেনা, আর **Hourly Rate** পদ্ধতিতে এর বিপরীতে **Hourly Rate** থাকে, কিন্তু **Annual Salary** থাকেনা।

ডেটায় কিছু কলামে অনুপস্থিত মান (missing data) থাকার কারণে সেই কলামগুলোতে **NaN** দেখা যাচ্ছে। উদাহরণস্বরূপ, যদি কর্মচারী **Salary** পদ্ধতিতে থাকে, তবে **Hourly Rate** ফাঁকা থাকবে, এবং এর বিপরীতে **Hourly Rate** পদ্ধতিতে থাকা কর্মচারীর জন্য **Annual Salary** ফাঁকা থাকবে।

এটি হল প্রকৃত ডেটার গঠন এবং ডেটার স্বাভাবিক আচরণ, এবং কখনো কখনো এই ধরনের তথ্য অনুপস্থিত থাকতে পারে।

|  |
| --- |
| import numpy as np  import pandas as pd  # Read City\_of\_Chicago\_Payroll\_Data.csv as a dataframe in pay.  pay = pd.read\_csv('City\_of\_Chicago\_Payroll\_Data.csv')  # Show first five and last five records in your data.  print(f"First five records in the data:\n{pay}")  # Get the overview of your data.  print(f"Overview of the data:\n{pay.info()}") |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | **Name** | **Job Titles** | **Department** | **Full or Part-Time** | **Salary or Hourly** | **Typical Hours** | **Annual Salary** | **Hourly Rate** | | **0** | ALLISON, PAUL W | LIEUTENANT | FIRE | F | Salary | NaN | $107790.00 | NaN | | **1** | BRUNO, KEVIN D | SERGEANT | POLICE | F | Salary | NaN | $104628.00 | NaN | | **2** | COOPER, JOHN E | LIEUTENANT-EMT | FIRE | F | Salary | NaN | $114324.00 | NaN | | **3** | CRESPO, VILMA I | STAFF ASST | LAW | F | Salary | NaN | $76932.00 | NaN | | **4** | DOLAN, ROBERT J | SERGEANT | POLICE | F | Salary | NaN | $111474.00 | NaN | | **...** | ... | ... | ... | ... | ... | ... | ... | ... | | **32653** | ZYGOWICZ, PETER J | POLICE OFFICER | POLICE | F | Salary | NaN | $90024.00 | NaN | | **32654** | ZYLINSKA, KATARZYNA | POLICE OFFICER | POLICE | F | Salary | NaN | $48078.00 | NaN | | **32655** | ZYMANTAS, MARK E | POLICE OFFICER | POLICE | F | Salary | NaN | $87006.00 | NaN | | **32656** | ZYRKOWSKI, CARLO E | POLICE OFFICER | POLICE | F | Salary | NaN | $93354.00 | NaN | | **32657** | ZYSKOWSKI, DARIUSZ | CHIEF DATA BASE ANALYST | DoIT | F | Salary | NaN | $115932.00 | NaN |   32658 rows × 8 columns  <class 'pandas.core.frame.DataFrame'>  RangeIndex: 32658 entries, 0 to 32657  Data columns (total 8 columns):  # Column Non-Null Count Dtype  --- ------ -------------- -----  0 Name 32658 non-null object  1 Job Titles 32658 non-null object  2 Department 32658 non-null object  3 Full or Part-Time 32658 non-null object  4 Salary or Hourly 32658 non-null object  5 Typical Hours 7883 non-null float64  6 Annual Salary 24775 non-null object  7 Hourly Rate 7883 non-null object  dtypes: float64(1), object(7)  memory usage: 2.0+ MB |

এই কোডটি দিয়ে আপনি একটি CSV ফাইলকে একটি pandas ডেটাফ্রেমে লোড করছেন এবং কিছু তথ্য প্রদর্শন করছেন:

1. **লাইব্রেরি ইম্পোর্ট**: numpy এবং pandas লাইব্রেরি ইম্পোর্ট করা হয়েছে। pandas ডেটা ম্যানিপুলেশন এবং অ্যানালাইসিসে ব্যবহৃত হয়।
2. **CSV ফাইল পড়া**: pd.read\_csv() ফাংশন ব্যবহার করে City\_of\_Chicago\_Payroll\_Data.csv নামক ফাইলটি pay ডেটাফ্রেমে পড়া হয়েছে।
3. **প্রথম পাঁচটি এবং শেষ পাঁচটি রেকর্ড**: কোডে pay মুদ্রিত করা হয়েছে, তবে এটি শুধু প্রথম পাঁচটি এবং শেষ পাঁচটি রেকর্ড দেখাবে ।

* প্রথম অংশে ডেটাফ্রেমের প্রথম ৫টি সারি দেখা যাবে (যেমন: ALLISON, PAUL W BRUNO, KEVIN D COOPER, JOHN E CRESPO, VILMA I DOLAN, ROBERT J, ইত্যাদি)।
* শেষ অংশে ডেটাফ্রেমের শেষ ৫টি সারি দেখা যাবে (যেমন: ZYGOWICZ, PETER J ZYLINSKA, KATARZYNA ZYMANTAS, MARK E ZYRKOWSKI, CARLO E ZYSKOWSKI, DARIUSZ)।

1. **ডেটা সম্পর্কে সার্বিক তথ্য**: pay.info() ফাংশনটি ডেটাফ্রেমের কলাম সংখ্যা, নাল (missing) ভ্যালু এবং ডেটার ধরন সম্পর্কে তথ্য দেখায়।

Columns: মোট ৮টি কলাম

* কলামের বিস্তারিত: Name, Job Titles, Department, Full or Part-Time, Salary or Hourly, Annual Salary, Hourly Rate: অবজেক্ট টাইপ (স্ট্রিং)।
* Typical Hours: ফ্লোট বা সংখ্যাসূচক ডেটা, তবে কিছু মান NaN (মিসিং ডেটা)।

1----10 important

df.describe() দিয়ে শুধুমাত্র numerial value এ column এ কাজ করবে ।

df.describe(include='all') সব গুলো column এ কাজ করবে।

**What are the maximum, minimum and average Typical Hours? (use 'Typical Hours' column)**

Slicing করলে নির্দিষ্ট কলামে পাওয়া যাবে । df['Typical Hours'].max(), df['Typical Hours'].min(), df['Typical Hours'].mean(),

df.groupby("Salary or Hourly").count()[['Name']]

df['Salary or Hourly'].value\_counts()

দুইটাই একই

df.groupby("Department").count()['Name'].idxmax()

df[df['Department'] == 'POLICE'].groupby('Salary or Hourly').count()['Name']

|  |
| --- |
|  |

df[df['Department'] == 'POLICE']['Salary or Hourly'].value\_counts()

যেহেতু, annual salay এর আগে $ আছে, তাই সে object. তাকে str.replace(old, new) দিয়ে $ এর জায়গায় ফাকা স্থান বানাতে হবে ।

df ['Salary'] = df['Annual Salary'].str.replace('$', '').astype(float) কে object থেকে float এ রুপান্তরিত করতে হবে ।