🐼 Pandas – Complete Recall Notes (AI/ML Focus)

1️⃣ Core Data Structures

🔹 Series

- 1D labeled array
- Like a single column

```
s = pd.Series([10,20,30])
```

🔹 DataFrame

- 2D labeled table
- Rows + Columns

```
df = pd.DataFrame({"A":[1,2], "B":[3,4]})
```

2️⃣ Data Import / Export

```
pd.read_csv()
pd.read_excel()
pd.read_json()
pd.read_sql()
```

Export:

```
df.to_csv()
df.to_excel()
```

3️⃣ Basic Inspection (ALWAYS FIRST STEP)

```
df.head()
df.tail()
df.shape
df.columns
df.index
df.dtypes
```

```
df.info()
df.describe()
df.value_counts()
```

🔥 In ML → `df.info()` + `df.describe()` are mandatory.

4️⃣ Selection & Indexing

🔹 Column Selection

```
df["col"]
df[["col1","col2"]]
```

🔹 Row Selection

loc → label-based

```
df.loc[0]
df.loc[0:5, ["A","B"]]
```

iloc → position-based

```
df.iloc[0]
df.iloc[0:5, 0:2]
```

5️⃣ Boolean Filtering (VERY IMPORTANT)

```
df[df["age"] > 25]
df[(df["age"]>25) & (df["salary"]>50000)]
```

Used heavily in EDA.

6️⃣ Missing Values Handling

```
df.isnull()
df.isnull().sum()
df.dropna()
df.fillna(0)
df.fillna(method="ffill")
```

ML Rule:

- Numeric → mean/median
- Categorical → mode

---

7 Data Cleaning

Rename columns

```
df.rename(columns={"old":"new"}, inplace=True)
```

Change data type

```
df["col"] = df["col"].astype(int)
```

Remove duplicates

```
df.drop_duplicates()
```

---

8 Sorting

```
df.sort_values("col")
df.sort_values("col", ascending=False)
```

---

9 Aggregation & Grouping (CORE ML SKILL)

Basic aggregation

```
df["col"].mean()
df["col"].sum()
df["col"].median()
df["col"].count()
```

GroupBy

```
df.groupby("col").mean()
df.groupby("col")["salary"].mean()
df.groupby("col").agg(["mean","sum"])
```

🔥 Used in feature analysis.

---

10 Apply / Map / Lambda

map (Series only)

```
df["col"].map({"M":0,"F":1})
```

apply

```
df["col"].apply(lambda x: x*2)
```

apply on DataFrame

```
df.apply(np.sum)
```

---

11 Merge / Join / Concat

Merge (SQL style)

```
pd.merge(df1, df2, on="id", how="inner")
```

how = inner / left / right / outer

Concat

```
pd.concat([df1, df2])
```

---

12 Pivot & Crosstab

```
df.pivot_table(values="salary", index="dept", aggfunc="mean")
pd.crosstab(df["gender"], df["dept"])
```

---

13 Working with Dates

```
df["date"] = pd.to_datetime(df["date"])
df["year"] = df["date"].dt.year
df["month"] = df["date"].dt.month
```

Important for time-series ML.

1️⃣4️⃣ String Operations

```
df["col"].str.lower()
df["col"].str.upper()
df["col"].str.contains("abc")
```

1️⃣5️⃣ Correlation (For ML Feature Insight)

```
df.corr()
```

Used before modeling.

1️⃣6️⃣ Sampling

```
df.sample(5)
df.sample(frac=0.2)
```

1️⃣7️⃣ Performance & Vectorization

Never use loops ❌
Use vectorized operations ✅

```
df["new"] = df["A"] + df["B"]
```