**PART 1: PYTHON & PANDAS SUMMARY**

**ESSENTIAL COMMANDS**

*read\_csv()* – Load a CSV file into a DataFrame for analysis. Useful for local or remote files. Add sep, index\_col, or encoding for better control.  
*head()*, shape, dtypes, columns – Preview top records, row/column count, data types, and column names. Helps understand structure before processing.  
*loc[]*, *iloc[]* – Extract rows/columns by label or index. Crucial for slicing and dicing data sets based on location or names.

**ANALYSIS & CLEANING**

*value\_counts()*, *unique()*, *mean()* – Quickly summarize categorical or numeric features. Understand distributions and central tendencies.  
*describe()*, *quantile()* – Return statistics such as min, max, mean, percentiles. Great for exploratory data analysis (EDA).  
*fillna()*, *drop()*, *replace()* – Handle missing values, remove rows/columns, or replace specific values to clean your dataset.

**TRANSFORMATIONS & EXPORT**

*astype()* – Convert columns to a specific data type (e.g., category, int). Helps reduce memory or enforce consistency.  
*str.lower()*, *apply()* – Normalize string columns or apply functions to Series/DataFrame. Essential for data standardization.  
*to\_csv()* – Save the cleaned DataFrame as a CSV file. Use arguments to exclude index or choose delimiter.

**GROUPING & MERGING**

*groupby()*, *pivot\_table()* – Aggregate data by categories, with optional custom functions. Allows powerful summaries (e.g., mean per region).  
*merge()*, *concat()* – Combine datasets either by common key or stacking. Use joins (inner, left, right, outer) for precision.

**FUNCTIONS**

*lambda*, *def*, *apply*, *map* – Use quick anonymous functions or reusable named ones. Apply transformations row-by-row or element-wise. Essential for feature engineering.

**COMMON PATTERNS**

*isnull().sum()* – Count missing values column-wise. Crucial for early diagnostics.  
*get\_dummies()* – Convert categorical columns to multiple binary (0/1) columns for ML models.  
*df[df["Age"] > 30]* – Filter rows by condition. Combine multiple conditions using & or | as needed.

**PART 2: GOVERNANCE, SECURITY & LIFECYCLE**

**DATA GOVERNANCE**

**Principles:** Confidentiality (data privacy), Integrity (data accuracy), Availability (data access when needed)  
**Enforced via:** Encryption, access controls, backup systems, audits, compliance reviews  
**Roles:**

* **Data Owner:** Defines usage and policies
* **Data Steward:** Maintains data accuracy and standards
* **Data Analyst:** Uses data for reporting and decision-making

**DATA LIFECYCLE (Lab 7)**

Phases:

* **Create:** Collect or generate data
* **Process:** Clean, transform, validate
* **Store:** Save securely (cloud or local)
* **Use:** Analytics, dashboards, decision support
* **Archive:** Long-term storage for inactive data
* **Destroy:** Secure deletion when obsolete  
  **Tools:** Informatica for ETL, AWS Data Lifecycle Manager for cloud, IBM InfoSphere for governance

**SECURITY & PRIVACY (Lab 8)**

**Tools:**

* VPNs for secure transmission
* Data masking during cleaning
* Encryption at rest & in transit
* Role-based access control
* Differential privacy for ML datasets  
  **Risks:** Data breaches, unauthorized access, policy non-compliance, loss of trust  
  **Responses:** Minimize data use, audit logs, encryption standards (AES-256), security policies

**PART 3: ETHICS IN TECH & AI**

**ETHICAL PRINCIPLES**

**Codes Studied:** ACM, BCS, CIPS, EU AI Guidelines, GenAI Voluntary Code  
**Shared Values:** Transparency, fairness, public good, privacy, responsibility  
**Needs Unique to AI:** Detect & fix algorithmic bias, explainability (why a model made a decision), fairness across demographics, accountability in automation

**FRAMEWORKS**

* **Utilitarianism:** Ethical if it produces the most benefit to most people. Relevant in AI for large-scale decision-making.
* **Deontology:** Based on rights and duties—certain actions are wrong regardless of outcomes (e.g., using personal data without consent).

**BEST PRACTICES**

* Build explainability into models using XAI techniques (LIME, SHAP)
* Human-in-the-loop systems for high-risk decisions (healthcare, finance)
* Internal ethical review boards
* Documentation: model cards, data sheets for datasets

**CASE STUDIES**

* **Cambridge Analytica:** Misuse of Facebook user data for political influence
* **COMPAS Tool:** Bias in predicting recidivism, with unequal treatment by race
* **Amazon Facial Recognition Moratorium:** Voluntary halt due to fairness & surveillance concerns

**PART 4: MIDTERM 2 THEORY RECAP**

**DATA TYPES**

**Transactional Data:** Real-time operations (e.g., sales logs, ATM usage). Volatile and frequently updated.  
**Analytical Data:** Summarized over time, used for insights and forecasting (e.g., marketing reports, BI dashboards).

**CLOUD SECURITY**

**Risks:** Shared resources, offsite data, lack of control  
**Controls:** Encryption, secure credentials, regulatory compliance (e.g., GDPR, HIPAA), vendor audits

**ZERO TRUST & IDENTITY CONTROL**

**Zero Trust Model:** Never trust, always verify. Access is evaluated per request based on credentials and behavior.  
**Identity-based Access Control (IBAC):** Users get permissions based on role/identity. Supports least privilege principle.

**LINEAGE & QUALITY**

**Data Lineage:** Shows origin, movement, and transformation of data. Helps trace errors and validate transformations.  
**Quality Dimensions:** Accuracy, completeness, consistency, timeliness. Enforced through data profiling and validation.

**MONITORING & ROLES**

**Monitoring Types:**

* **Usage Monitoring:** Who accessed what and when
* **Compliance Monitoring:** Alignment with data policies
* **Performance Monitoring:** Uptime, load times, failures  
  **Roles Recap:** Owners define access, Stewards ensure quality, Analysts interpret data

**ETHICAL AI**

* AI is high-stakes: it influences employment, healthcare, justice, etc.
* Needs ethical codes addressing bias, autonomy, data misuse, and transparency
* Encourage proactive regulation, model audits, and fairness metrics