# DATASHIELD – ENSURING PRIVACY WITH K-ANONYMITY PROJECT TUTORIAL

**OF**

**DATABASE AND ONLINE SOCIAL MEDIA SECURITY**

**(CSLM 654)**

**MASTER OF TECHNOLOGY**

**In**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted By**
**ARWAZ KHAN & ARYAM SHRIVASTVA (242210005 & 242210006)**

**Submitted To**
**DR. SHELLY SACHDEVA (ASSOCIATE PROFESSOR, DoCSE)**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**
**NATIONAL INSTITUTE OF TECHNOLOGY DELHI MAY 2025**

# K-ANONYMITY

## 1. What is k-Anonymity?

k-Anonymity is a privacy-preserving technique used in data publishing to prevent the re-identification of individuals in datasets. It ensures that any individual cannot be distinguished from at least k − 1 other individuals based on a set of quasi-identifiers (QIDs). A dataset satisfies k-anonymity if every combination of quasi-identifier attributes occurs in at least k records.

- Quasi-identifiers: Attributes like age, ZIP code, or gender that may not uniquely identify someone on their own but can do so when combined.
- Anonymized records: By generalizing or suppressing QIDs, the dataset ensures that each person's record is indistinguishable from at least k−1 others.

### 1.1 When is k-Anonymity Used?

- When releasing datasets for research or statistical purposes while preserving user privacy.
- In healthcare, finance, or government records, where sensitive data must be protected from re-identification.
- To comply with privacy regulations like GDPR or HIPAA.
- When publishing public datasets for data mining, machine learning, or academic use.

### 1.2 How does k-Anonymity Work?

1. **Identify Quasi-Identifiers (QIDs):**
   - Detect which attributes could be used to identify individuals when combined with external information.
2. **Generalization and Suppression:**
   - Generalize specific values (e.g., age 28 → 20–30).
   - Suppress values where generalization is insufficient.

3. **Group Records:**
    - Modify the dataset such that for every set of QIDs, there are at least **k** identical records.
4. **Check Anonymity:**
    - Ensure that every record is indistinguishable from at least $\mathbf{k-1}$ others based on QIDs.

### 1.3 Example of k-Anonymity Work

| Age | ZIP Code | Disease |
|---|---|---|
| 25 | 13053 | Flu |
| 27 | 13068 | Cold |
| 29 | 13053 | Cancer |

After 3-Anonymity:

| Age | ZIP Code | Disease |
|---|---|---|
| 25 | 13*** | Flu |
| 27 | 13*** | Cold |
| 29 | 13*** | Cancer |

Now, any individual cannot be re-identified since each row shares QID values with at least two others (k=3).

### 1.4 Limitations of k-Anonymity

- **Homogeneity Attack:** All records in a group have the same sensitive value, making inference easy.
- **Background Knowledge Attack:** If an attacker knows additional information, k-anonymity may still leak data.

- Does not protect against attribute disclosure, only identity disclosure.

To address these, more advanced techniques like l-diversity and t-closeness have been introduced.

## 2. Setup

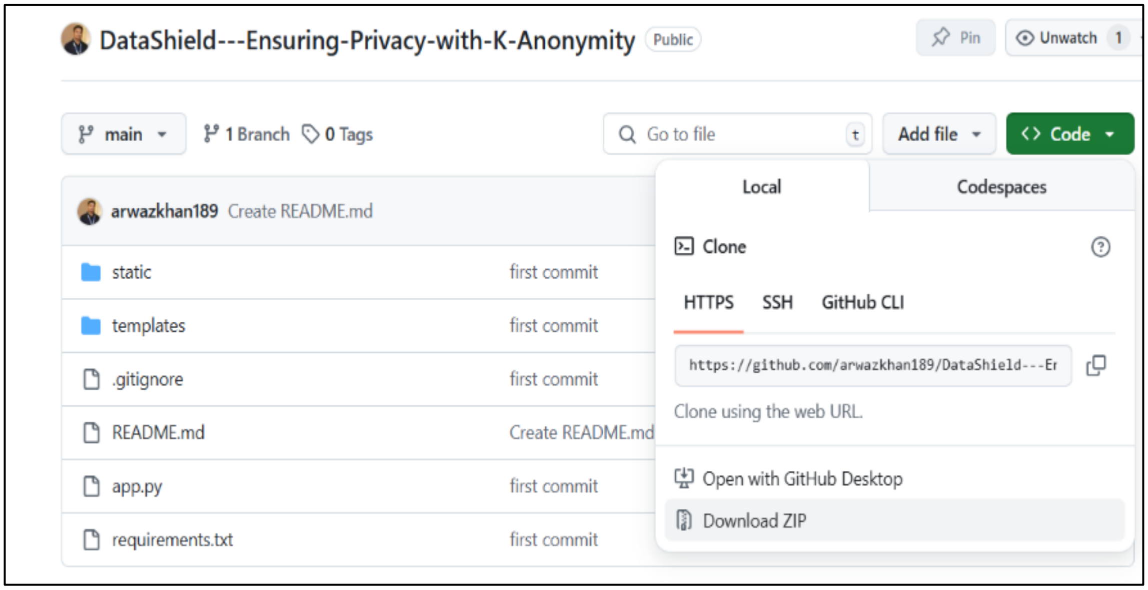
- Download & Install VS Code https://code.visualstudio.com/download
- Download & Install Python https://www.python.org/downloads/
- Install Python libraries in command prompt  
  pip install flask, pandas, numpy

## 3. Create the file structure

- ➢ app.py
- ➢ requirements.txt
- ➢ static / styles.css
- ➢ static/ script.js
- ➢ templates / index.html

## 4. Steps to start the project

**Step 1:** Download the ZIP file from the following link and extract its contents: https://github.com/arwazkhan189/DataShield---Ensuring-Privacy-with-K-Anonymity



arwazkhan189 Create README.md

| | |
|---|---|
| static | first commit |
| templates | first commit |
| .gitignore | first commit |
| README.md | Create README.md |
| app.py | first commit |
| requirements.txt | first commit |

```
https://github.com/arwazkhan189/DataShield---Er
```

**Fig 4.1:** GitHub Repository Page

**Step 2:** Launch VS Code and open the extracted project folder.

**Step 3:** Open the terminal in VS Code and run the application using the command:

```
py app.py
```

**Step 4:** Once the server starts, open the localhost URL displayed in the terminal in your web browser.

**Step 5:** The web application will now be displayed in your browser.

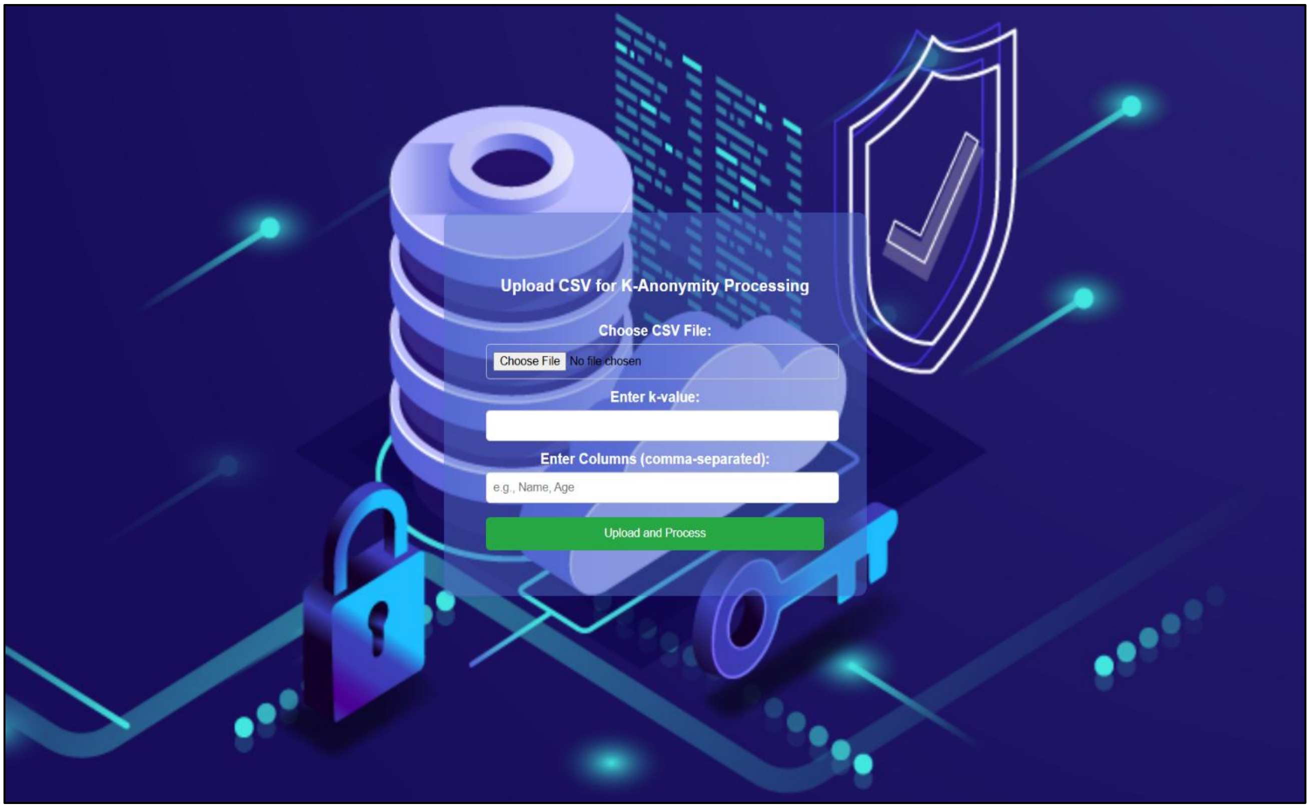
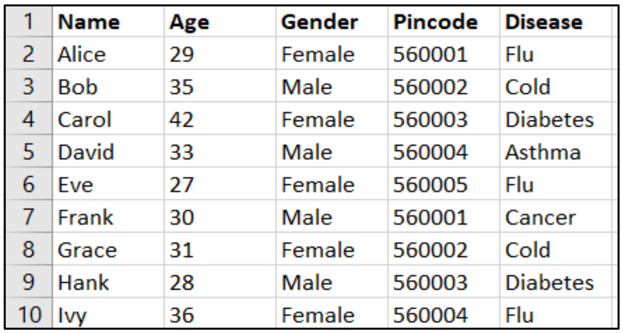


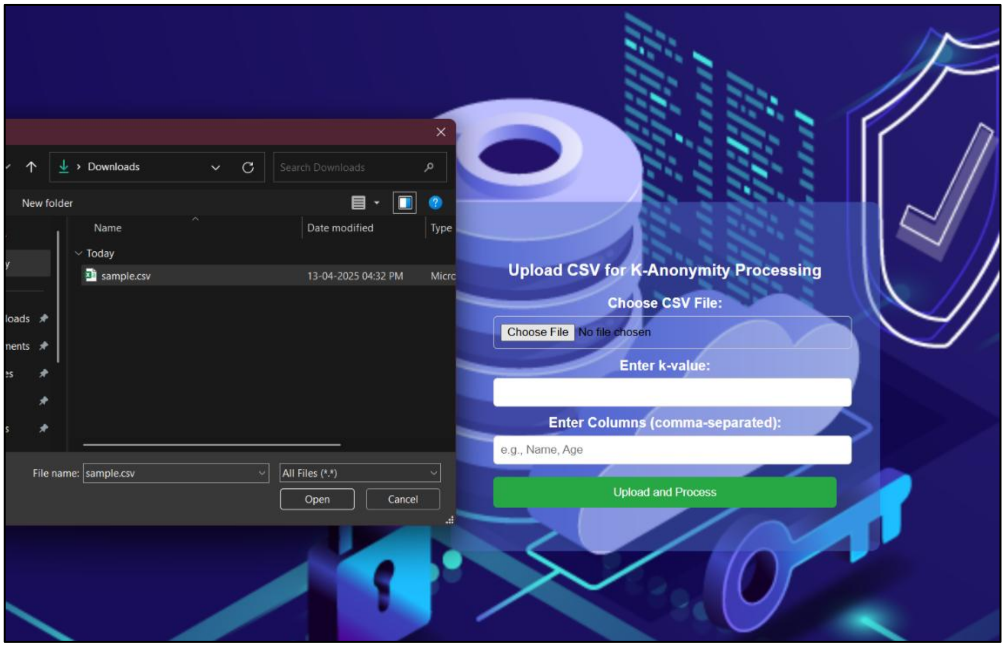
**Fig 4.2:** Web Application Interface

**Step 6:** Choose a sample dataset on which you want to apply k-anonymity.



| 1 | Name | Age | Gender | Pincode | Disease |
|---|---|---|---|---|---|
| 2 | Alice | 29 | Female | 560001 | Flu |
| 3 | Bob | 35 | Male | 560002 | Cold |
| 4 | Carol | 42 | Female | 560003 | Diabetes |
| 5 | David | 33 | Male | 560004 | Asthma |
| 6 | Eve | 27 | Female | 560005 | Flu |
| 7 | Frank | 30 | Male | 560001 | Cancer |
| 8 | Grace | 31 | Female | 560002 | Cold |
| 9 | Hank | 28 | Male | 560003 | Diabetes |
| 10 | Ivy | 36 | Female | 560004 | Flu |

**Fig 4.3:** Sample Dataset Used for K-Anonymity



sample.csv 13-04-2025 04:32 PM Micro

File name: sample.csv

All Files (*.*)

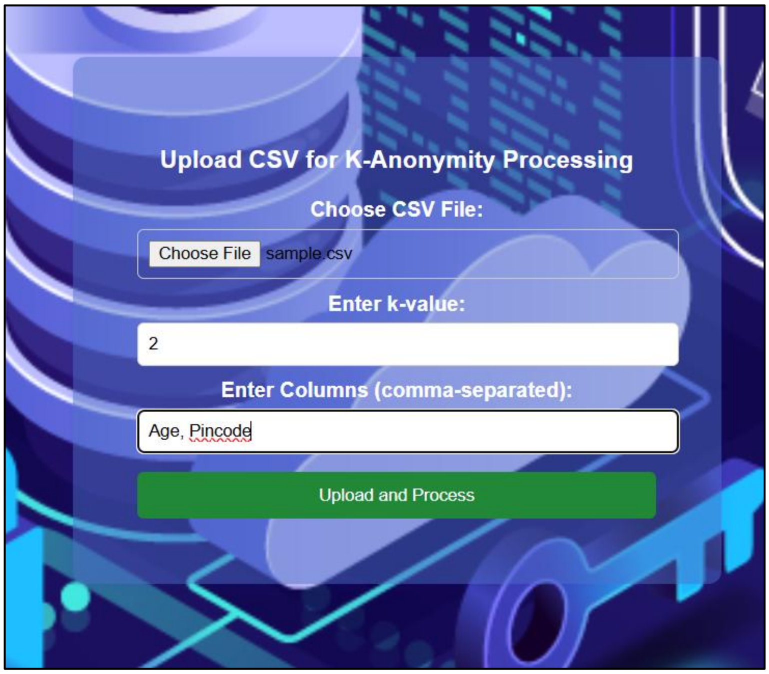
No file chosen

e.g., Name, Age

**Fig 4.4:** Selecting the Sample Dataset for Processing

**Step 7:** Specify the k-value to define the level of anonymity.

**Step 8:** Provide the column names, separated by commas, that should be considered for anonymization.



Upload CSV for K-Anonymity Processing
Choose CSV File:
Choose File
Enter k-value:
Enter Columns (comma-separated):
Upload and Process

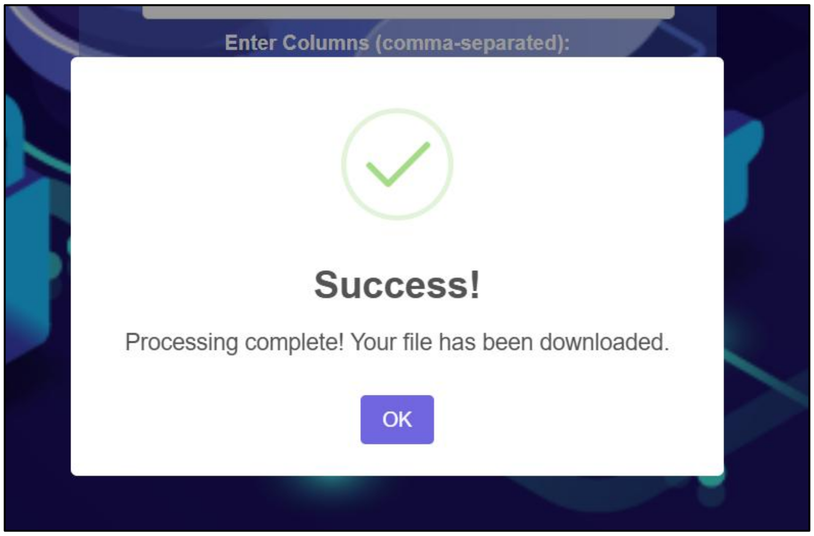
sample.csv

2

Age, Pincode

**Fig 4.5:** Defining the K-Value and Specifying Column Names for Anonymization

**Step 9:** Click on the "Upload and Process" button to process the dataset and download the anonymized output.



Success!

Processing complete! Your file has been downloaded.

**Fig 4.6:** Processed Dataset Downloaded Successfully

**Step 10:** Navigate to the Downloads folder and open the file named anonymized_file.csv to view the anonymized dataset.

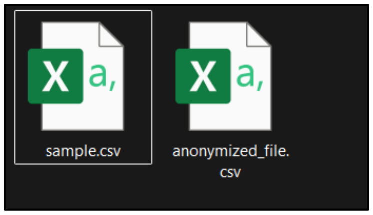
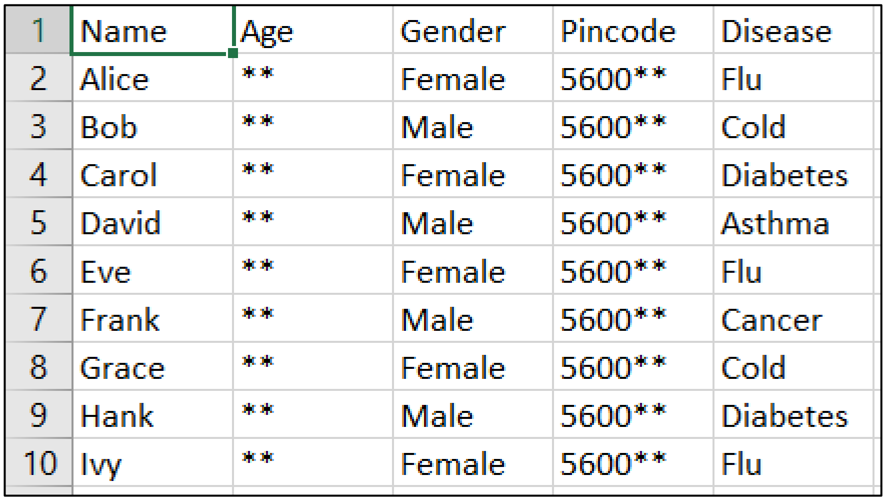


**Fig 4.7:** Downloaded Anonymized Dataset



| 1 | Name | Age | Gender | Pincode | Disease |
|---|---|---|---|---|---|
| 2 | Alice | ** | Female | 5600** | Flu |
| 3 | Bob | ** | Male | 5600** | Cold |
| 4 | Carol | ** | Female | 5600** | Diabetes |
| 5 | David | ** | Male | 5600** | Asthma |
| 6 | Eve | ** | Female | 5600** | Flu |
| 7 | Frank | ** | Male | 5600** | Cancer |
| 8 | Grace | ** | Female | 5600** | Cold |
| 9 | Hank | ** | Male | 5600** | Diabetes |
| 10 | Ivy | ** | Female | 5600** | Flu |

**Fig 4.8:** View of the Anonymized Dataset in CSV Format

## Future Work

- Integrate advanced models like l-diversity and t-closeness to improve privacy.
- Enable real-time anonymization for streaming data.
- Improve scalability to handle large datasets efficiently.
- Allow user-defined privacy levels for flexible control.
- Incorporate privacy-preserving machine learning techniques.
- Enhance user interface and data visualization tools.
- Add evaluation metrics to balance privacy and utility.
- Ensure compliance with privacy laws like GDPR (General Data Protection Regulation).

## References

[1] L. Sweeney, “k-Anonymity: A model for protecting privacy,” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 5, pp. 557–570, 2002.

[2] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, “l-Diversity: Privacy beyond k-anonymity,” ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 1, no. 1, pp. 3–es, 2007.

[3] R. Elmasri and S. B. Navathe, Fundamentals of Database Systems, 7th ed. Pearson, 2015.