Data Masking:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Employee ID | Name | User ID | Gender | Age | Phone Number | Date of Birth |
| 57861 | Ralph Kendrick | dem55 | O | 35 | 7758956347 | 20-05-1988 |
| 18953 | Dorothy Leadbetter | har83 | M | 20 | 9150357745 | 07-07-2003 |
| 33291 | Greg Rm | kja96 | F | 18 | 7865489007 | 04-10-2005 |
| 96272 | Craig King | jsm77 | F | 44 | 9374674890 | 16-08-1979 |
| 7195 | Linda Bahn | pnx31 | M | 65 | 7689988542 | 29-09-1958 |
| 94757 | Liane Lavin | jdb81 | F | 43 | 9347897140 | 08-07-1980 |
| 48167 | Mary James | bwg34 | O | 57 | 9445678912 | 13-04-1966 |
| 50730 | Kenneth Bass | fsk57 | F | 45 | 9946771966 | 28-02-1978 |
| 46122 | Matthew Lara | rdv50 | M | 28 | 9994034782 | 25-03-1995 |
| 78912 | Greig Smith | grs10 | O | 32 | 9387361103 | 01-01-1991 |

|  |  |  |  |
| --- | --- | --- | --- |
| Driver's License Number | Passport Number | Bank Account Number | Email address |
| US2320217789012 | A3903463 | 15432964088 | [kend634@gmail.com](mailto:kend634@gmail.com) |
| US1420201474654 | B5238914 | 11653894257 | [lead774@gmail.cmm](mailto:lead774@gmail.cmm) |
| US8520228548916 | P8539510 | 12546729086 | [graig634@gmail.com](mailto:graig634@gmail.com) |
| US3420211174590 | A8239058 | 26789259789 | [king489@gmail.com](mailto:king489@gmail.cfm) |
| US2120213382901 | U6452890 | 85372897256 | [behn854@gmail.com](mailto:behn854@gmail.com) |
| US9020226739034 | N6349025 | 67387098734 | [lavi714@gmail.com](mailto:lavi714@gmail.com) |
| US1120194590763 | S7450928 | 52890156238 | [jame891@gmail.com](mailto:jame891@gmail.com) |
| US2720237320165 | P6428904 | 62296086118 | [bass196@gmail.com](mailto:bass196@gmail.com) |
| US3420212890123 | C6328942 | 45396802167 | [lara478@gmail.com](mailto:lara478@gmail.com) |
| US2720196219054 | F7501358 | 16724390568 | [smit110@gmail.com](mailto:smit110@gmail.com) |

|  |  |  |  |
| --- | --- | --- | --- |
| Weight | D.NO | Street name | Pin code |
| 89.74 | c-20 | Park Avenue | 625016 |
| 56.56 | E-85 | Washing Tone Street | 624902 |
| 66.5 | S-60 | Crosby Street | 623917 |
| 94.69 | W-30 | Fifth Avenue Street | 623908 |
| 70.86 | D-9 | Broadway | 536894 |
| 80.78 | F-8 | St Mark's | 456789 |
| 61.65 | G-9 | Madison Avenue | 345678 |
| 70.74 | C-22 | NYC street | 536478 |
| 50.67 | D-59 | Jones Street | 657383 |
| 68.39 | R-20 | Walls Street | 643589 |

\*The above data which we are working on contains all the data types for performing data masking.

**PHONE NUMBER/EMP ID**

1. Problem statement: To mask a list of Phone number.
2. Inputs: A list of Phone number (Integers).
3. Outputs: To make a list of masked Phone number. (Mask the real one)
4. Pseudocode:
   * In the given Phone number, Keep the first 5 digits.
   * Mask the rest 5 digits with asterisk (\*).
   * Return the Masked Phone number.

**1.RANDOM DIGIT MASKING:**

Random digit masking is a technique for masking phone numbers by replacing some of the digits with X's or \*'s. The number of digits to be masked and their positions are chosen randomly.

In this example, three digits were chosen at random to be masked with X's. The resulting masked phone number is '55X-12X-45X7'.

INPUT:

555-123-4567

OUTPUT:

55X-12X-45X7

**2.REVERSAL MASKING:**

Reversal masking is a technique for masking phone numbers by first reversing the digits in the phone number and then masking some of them with X's or \*'s.

The number of digits to be masked and their positions are chosen randomly.

INPUT:

9840128511

AFTER REVERSING:

1158210489

OUTPUT:

1\*58\*1\*0\*4\*9

**3. VOWEL MASKING**

Vowel masking is a technique for masking phone numbers by replacing the vowels in the phone number with X's or \*'s.

INPUT:

9840128511

OUTPUT:

98XXX28511

**4.CONSONANT MASKING:**

Consonant masking is a technique for masking phone numbers by replacing the consonants in the phone number with X's or \*'s.

INPUT:

9840128511

OUTPUT:

984XXXXXXX

**5.PATTERN MASKING:**

Pattern masking is a technique for masking phone numbers by replacing specific patterns in the phone number with X's or \*'s.

For example, you might want to mask the area code of a phone number while leaving the rest of the number intact.

INPUT:

5550128511

OUTPUT:

XXX0128511

**6.RANDOMIZED OFFSET MASKING**

Randomized offset masking is a technique for masking phone numbers where a random number of digits is masked from a random offset.

In this example, a random offset is chosen between 0 and the length of the phone number minus 1, and a random number of digits to mask is chosen between 1 and the remaining length of the phone number.

The selected digits are then masked with X's. The resulting masked phone number is something like '5XX-123-4567'.

INPUT:

555-123-4567

OUTPUT:

555-123-4567

**7.SOUNDEX MASKING**

Soundex masking is a technique for masking phone numbers where each digit in the phone number is replaced with a Soundex code, and then some of the Soundex codes are masked with X's or \*'s.

INPUT:

555-123-4567

OUTPUT:

555X

**8.BINARY MASKING:**

In this example, each digit in the phone number is converted to binary format using four bits, and then a random number of bits are masked with X's.

The masked binary number is then converted back to decimal format and formatted as a phone number. The resulting masked phone number is something like '5XX-12X-4X67'.

INPUT:

555-123-4567

OUTPUT:

5XX-12X-4X67

**9.ASCII MASKING:**

converts each digit in the phone number to its ASCII code (using **ord()**), and then masks a random number of the resulting characters.

The masked characters are replaced with X's, and the resulting masked string is converted back to a phone number format by converting each pair of ASCII characters back to their corresponding character using **chr()**.

1**0.BASE64 MASKING:**

Function takes a phone number as input, encodes it in Base64 format, randomly masks some of the Base64 characters with X's or \*'s, and then decodes the masked Base64 string back to the original phone number format.

**DOOR NUMBER**

**Problem statement:To mask a list of door number**

**Input:**A list of door numbers

**Outputs:**To make a list of fake numbers(mask as real data)

**1.SHUFFLING:**

\*Total we have 9 rows split into each three columns

\*suffle among itself randomly

INPUT:

C-20

E-85

C-18

**OUTPUT:**

C-18

E-85

C-20

**2.GENERATING THE SECOND NUMBER:**

**\***Here the second number from the door number itself randomly generated

INPUT:

C-20

OUTPUT:

C-40

**3.INTERCHANGING THE FIRST LETTER:**

\*From the door number first letter will get interchanged among itself

\*interchanging occurs among the column of door number

INPUT:

C-20

E-85

C-40

OUTPUT:

E-20

C-85

D-40

**5.SHUFFLING AMONG ALL:**

\*In this algorithm entire column itself get shuffled among randomly

\*And it will get assinged to some respective columns finally

INPUT:

C-20

E-85

C-90

D-40

OUTPUT:

D-40

E-85

C-20

C-90

**6.MATHAMETICAL ALGORITHM:**

\*In this algorithm some two door number has been taken and the frst door number gets added by the second door number value print some output at the frst door number place

INPUT :

C-20

E-40

OUTPUT:

C-60

\*Then in second door place output of the first door number will get added with the original value of the second door number.

INPUT:

C-60

E-40

OUTPUT:

E-100

FINAL OUTPUT:

C-60

E-100

**7.RANDOMIZATION:**

\*In this algorithm enitre second column has been generated randomly

\*In other, the entire first column has been generated randomly

**8.TRUNCATION ALGORITHM:**

\*This algorithm removes part of the door number, such as the last one or two digits, and replaces them with a symbol.

\*For example, the door number "1234" might be truncated to "12XX" or "1XXX".

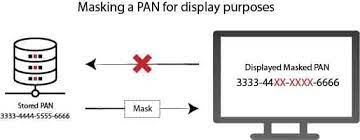
\*This approach can be useful for preserving the overall shape of the door number while obscuring some of its specific details.

INPUT:

C-20

OUTPUT:

C-XX



**9. SUBTITUTION ALGORITHM:**

This algorithm replaces the digits of the door number with other symbols or characters.

For example, the door number "12-34" might be masked as "\*\*-\*\*".

Alternatively, the algorithm could substitute each digit with a letter or other symbol, such as "D###" or "XO-XO".

This approach can be useful for preserving the length and format of the original door number.

INPUT

C-20

OUTPUT

\*\* - \*\*

INPUT:

C-20

OUTPUT:

C\*\*\*

**10.RANDOMIZATION :**

This algorithm randomly shuffles the digits of the door number to create a new, randomized number.

For example, if the door number is "C-34", the algorithm might create "34-C" or "C-43". This approach can help protect the privacy of the door number by making it more difficult to guess.

**EMAIL ADDRESS**

## Problem statement: To mask a list of email address.

## Inputs: A list of email address.

## Outputs: To make a list of fake data (Mask the real one)

## Pseudocode:

Reverse the string characters in the given email address.

Return the masked data.

**1.SPLITING:**

John.doe@gmail.com - j\*\*\*\*\*\*@\*\*\*\*\*m.

**2.CAPITALISATION:**

John.doe@gmail.com -jOhN.dOe@GmaIl.com

**3.HASH MASKING ALGORITHM:**

John.doe@example.com-'73c9a89e2492a2b7aa41a1f0ce7f350ab2e5ba01e282077d1c45637c5b5f5c5f'

**4.FULL MASKING:**

'john.doe@example.com'-\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**5.OBFUSCATION MASKING:**

John.deo@example.com- j0hn.d03@3xampl3.c0m

**6.REPETITION:**

John.deo@example.com- jjjooohhnn...ddooeee@eexxaammppllee.coom

**7.INSERTION:**

John.deo@example.com- [joh\*ndoe@exa\*mple.com](mailto:joh*ndoe@exa*mple.com)

**8.REVERSING:**

John.deo@example.com- [moc.elpmaxe@eod.nhoj](mailto:moc.elpmaxe@eod.nhoj)

**9.CONVERSTION :**

[John.deo@example.com](mailto:John.deo@example.com)

\*First convert to ascii values- 106 111 104 110 46 100 101 111 64 101 120 97 109 112 108 101 46 99 111 109

John

J-106- 1101010

o-111- 1101111

h-104- 1101000

n-110- 1101110

\*Count the 1’s if it is odd do the and operation with random number

\*Or if it is even do the or operation with random number

**PINCODE:**

## Problem statement: To mask a list of Bank Account number.

## Inputs: A list of numbers.

## Outputs: To make a list of fake numbers (Mask the real one)

## Pseudocode:

## Keep the first 5 integers and reverse the next 2 integers.

* Mask the balance integers with asterisk (\*).
* Return the masked data

**1.REVERSING:**

\*625016-610526

**2.SORTING**

\*625016-012566

**3.BITWISE MASKING**

\*INPUT=625016

\*Convert into ascii numbers

6-54-110110

2-50-110010

5-53-110101

0-48-110000

1-49-110001

6-54-110110

\*Count the number of one’s if it is odd do the and operation else if it is even do the or operation

OUTPUT=3A3E9639329334=5862150575014758

4.Shuffling among itself

5.Divide the number, first three numbers remind same and next 3 numbers will get hashed

6. Divide the number, next three numbers remind same, and first 3 numbers will get hashed

7.Adding random value and print the same characters of number as output.

8.remind the first and last number and swap the between numbers.

9.Multiply some random value to the middle portion and remind the same characters as output.

10.divide the number, keep the first three numbers and shuffle the balance among itself

11. divide the number, keep the first three numbers and print the balance among itself.

12. divide the number, keep the next three numbers and print the first among itself.

**GENDER**

Problem statement:To mak a list of gender in the column gender

Inputs:A list of gender

Outputs:To make a list of fake gender(masked as real one)

Pseudo Code: