In [16]:
```
import pandas as pd
from scipy.sparse import coo_matrix
import numpy as np

ratingData = pd.read_csv('Ratings.csv', delimiter=';', dtype={'User-ID': str}, low_memory=False) #
userData = pd.read_csv('Users.csv', delimiter=';', dtype={'User-ID': str}, low_memory=False) #



userRatingData = pd.merge(ratingData, userData, on='User-ID', how='left') # Merge ratings and user
```

In [18]:
```
# Separating users based on age
userAgeData = userRatingData[~userRatingData['Age'].isna()]
userNoAgeData = userRatingData[userRatingData['Age'].isna()]

def createSparseMatrix(userRating, fileName):

    userRating = userRating[userRating['Rating'] > 0]

    userIds = userRating['User-ID']
    bookIds = userRating['ISBN']
    ratings = userRating['Rating']

    userIdMapping = {}
    bookIdMapping = {}
    userCodes = []
    bookCodes = []

    userCounter = 1
    bookCounter = 1

    for userId, bookId in zip(userIds, bookIds):
        if userId not in userIdMapping:
            userIdMapping[userId] = userCounter
            userCounter += 1
        userCodes.append(userIdMapping[userId])

        if bookId not in bookIdMapping:
            bookIdMapping[bookId] = bookCounter
            bookCounter += 1
        bookCodes.append(bookIdMapping[bookId])

    userCodes = np.array(userCodes)
    bookCodes = np.array(bookCodes)

    sparse_matrix = coo_matrix((ratings.values, (userCodes, bookCodes)), dtype=int)
    sparse_matrix_csr = sparse_matrix.tocsr()

    with open(fileName, 'w') as f:
        for userCode in range(1, sparse_matrix_csr.shape[0]):
            row = sparse_matrix_csr.getrow(userCode)
            features = [f"{bookCode}:{row[0, bookCode]}" for bookCode in row.indices]
            if features:
                line = "0 "+" ".join(features) + "\n"
                f.write(line)

# Generate sparse matrices for both groups
createSparseMatrix(userAgeData, 'Userbook_ratings_with_age.libsvm')
createSparseMatrix(userNoAgeData, 'Userbook_ratings_without_age.libsvm')

print("Sparse matrices created for users with and without age information.")
```

```
Sparse matrices created for users with and without age information.
```

In [ ]: