

```
In [1]: import pandas as pd
```

```
In [14]: current_employee_data=pd.read_csv("C:/Users/malav/Downloads/input.csv")
current_employee_data
```

Out[14]:

	Employee Code	Manager Employee Code	Date of Joining	Date of Exit	Compensation	Compensation 1	Compensation 1 date	Compensation 2	Compensation 2 date	Review 1	Review 1 date	Review 2
0	1	NaN	2021-01-01	NaN	20000	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	2	1.0	2021-01-01	NaN	20000	10000.0	2022-01-01	20000.0	2023-01-01	9.0	2021-06-01	9.5
2	3	1.0	2021-01-01	2023-12-31	20000	10000.0	2022-01-01	20000.0	2023-01-01	9.0	2021-06-01	9.5


```

In [24]: # Create an empty List to store historical versions
historical_employee_data = []

# Iterate through each row of the current employee data
for index, row in current_employee_data.iterrows():
    employee_code = row['Employee Code']
    manager_employee_code = row['Manager Employee Code']
    date_of_joining = row['Date of Joining']
    date_of_exit = row['Date of Exit']
    compensation = row['Compensation']
    compensation1 = row['Compensation 1']
    compensation1_date = row['Compensation 1 date']
    compensation2 = row['Compensation 2']
    compensation2_date = row['Compensation 2 date']
    review1 = row['Review 1']
    review1_date = row['Review 1 date']
    review2 = row['Review 2']
    review2_date = row['Review 2 date']
    engagement1 = row['Engagement 1']
    engagement1_date = row['Engagement 1 date']
    engagement2 = row['Engagement 2']
    engagement2_date = row['Engagement 2 date']

    # Create historical versions for compensation changes
    if not pd.isnull(compensation1):
        historical_employee_data.append({
            'employee_code': employee_code,
            'date': compensation1_date,
            'compensation': compensation1
        })
    if not pd.isnull(compensation2):
        historical_employee_data.append({
            'employee_code': employee_code,
            'date': compensation2_date,
            'compensation': compensation2
        })

    # Create historical versions for review changes
    if not pd.isnull(review1):
        historical_employee_data.append({
            'employee_code': employee_code,
            'date': review1_date,
            'review': review1
        })
    if not pd.isnull(review2):
        historical_employee_data.append({
            'employee_code': employee_code,
            'date': review2_date,
            'review': review2
        })

    # Create historical versions for engagement changes
    if not pd.isnull(engagement1):
        historical_employee_data.append({
            'employee_code': employee_code,
            'date': engagement1_date,
            'engagement': engagement1
        })
    if not pd.isnull(engagement2):
        historical_employee_data.append({
            'employee_code': employee_code,
            'date': engagement2_date,
            'engagement': engagement2
        })

    # Create historical versions for manager changes
    if not pd.isnull(manager_employee_code):
        historical_employee_data.append({
            'employee_code': employee_code,
            'date': date_of_joining, # Assuming the change in manager occurred on the joining date
            'manager': manager_employee_code
        })

    # Create historical versions for date of exit
    if not pd.isnull(date_of_exit):
        historical_employee_data.append({
            'employee_code': employee_code,
            'date': date_of_exit,
            'exit_date': date_of_exit
        })

# Convert the historical employee data to a DataFrame
historical_employee_df = pd.DataFrame(historical_employee_data)

# Sort the DataFrame by employee code and date
historical_employee_df = historical_employee_df.sort_values(by=['employee_code', 'date'])

```

```

# Reset index
historical_employee_df = historical_employee_df.reset_index(drop=True)

# Save the historical employee data to a CSV file
historical_employee_df.to_csv('historical_employee_data.csv', index=False)

print("Historical employee data saved successfully.\n",historical_employee_df)

```

Historical employee data saved successfully.

	employee_code	date	compensation	review	engagement	manager \
0	2	2021-01-01	NaN	NaN	NaN	1.0
1	2	2021-03-01	NaN	NaN	4.0	NaN
2	2	2021-06-01	NaN	9.0	NaN	NaN
3	2	2022-01-01	10000.0	NaN	NaN	NaN
4	2	2022-03-01	NaN	NaN	5.0	NaN
5	2	2022-06-01	NaN	9.5	NaN	NaN
6	2	2023-01-01	20000.0	NaN	NaN	NaN
7	3	2021-01-01	NaN	NaN	NaN	1.0
8	3	2021-03-01	NaN	NaN	4.0	NaN
9	3	2021-06-01	NaN	9.0	NaN	NaN
10	3	2022-01-01	10000.0	NaN	NaN	NaN
11	3	2022-03-01	NaN	NaN	5.0	NaN
12	3	2022-06-01	NaN	9.5	NaN	NaN
13	3	2023-01-01	20000.0	NaN	NaN	NaN
14	3	2023-12-31	NaN	NaN	NaN	NaN

	exit_date
0	NaN
1	NaN
2	NaN
3	NaN
4	NaN
5	NaN
6	NaN
7	NaN
8	NaN
9	NaN
10	NaN
11	NaN
12	NaN
13	NaN
14	2023-12-31