

# BOP Complaints

Data Librarianship and Management  
Final Project Check-in #2  
Fall 2024  
Emily Cunningham

< > 2024-10-21\_federal-inmate-com... ┌─────────┐ ┌─────────┐ ┌─────────┐ ┌─────────┐

Name	Size	Kind
data -- Folder	--	Folder
BOPDocuments -- Folder	--	Folder
adremedysubject.pdf	68 KB	PDF Document
adremvarlabels.xlsx	11 KB	Microso...k (.xlsx)
facility-codes.csv	4 KB	CSV Document
FacilityList1.pdf	651 KB	PDF Document
FacilityList2.pdf	74 KB	PDF Document
Federal Inmate Complaints — Data Documentation.pdf	308 KB	PDF Document
fields.csv	3 KB	CSV Document
reason-codes.csv	5 KB	CSV Document
CDFC_FacilityCodes.csv	19 KB	CSV Document
CDSTATUS_CaseStatusCodes.csv	94 bytes	CSV Document
cdsub1cb_ConcatSubjectCodes.csv	11 KB	CSV Document
CDSUB1PR_PrimarySubjectCodes.csv	960 bytes	CSV Document
ColumnCodes.csv	4 KB	CSV Document
ComplaintFilings.csv	201.4 MB	CSV Document
ITERLVL_OrgLevelCodes.csv	41 bytes	CSV Document
STATRSN_StatusReasonCodes.csv	6 KB	CSV Document
docs -- Folder	--	Folder
LICENSE	1 KB	Plain Text
README.md	32 bytes	Markdo...ument
results -- Folder	--	Folder
NYSubmissions.csv	5.2 MB	CSV Document
NYSubmissionsEnriched.csv	6.5 MB	CSV Document
NYSubmissionsEnrichedExpanded.csv	18.4 MB	CSV Document
output_percentages_primaryreason.csv	992 bytes	CSV Document
output_percentages_secondaryreason.csv	12 KB	CSV Document
UniqueNYSubmissionEnriched.csv	3.7 MB	CSV Document
UniqueNYSubmissionsEnrichedExpanded.csv	10.5 MB	CSV Document
src -- Folder	--	Folder
Create_NYComplaintFilings.ipynb	25 KB	Document
R.ipynb	233 KB	Document
R.txt	1 KB	Plain Text
top_complaints_pie_chart.png	116 KB	PNG image

```
# STEP 1: Create NY subset of data (NYSubmissions.csv)
# Find all NY Facilities in the CDFC_FacilityCodes.csv and save as a set
# Use the set to filter ComplaintFilings.csv so that we are left with only complaints made at NY facilities
# Save results to NYSubmissions.csv

import pandas as pd

# Load facilities data, filter for NY, and extract facility codes
allfacilities = pd.read_csv('../data/CDFC_FacilityCodes.csv')
nyfacilitycodes = set(allfacilities.loc[allfacilities['State'] == 'NY', 'Facility_Code'])

# Load submissions data and filter where CDFCLEVN is in NY facility codes
allsubmissions = pd.read_csv('../data/ComplaintFilings.csv')
nysubmissions = allsubmissions[allsubmissions['CDFCLEVN'].isin(nyfacilitycodes)]

# Save filtered submissions to a CSV file
nysubmissions.to_csv('../results/NYSubmissions.csv', index=False)
```

```

# STEP 2: Create a new enriched dataset (NYSubmissionsEnriched) by adding columns
# Add column (subcount) that counts total number of submissions associated with the CASENBR
# Add column (appcourt) that counts total number of appeals
# Add column (earliest_sitdtrcv) that has earliest Remedy Case Submission date received (sitdtrcv) for that CASENBR
# Add column (latest_sdtstat) that has Date latest status assigned (sdtstat) for that CASENBR
# Add column (days_between) that has number of days between earliest_sitdtrcv and latest_sdtstat. This is the amount of time either between first complaint and closure OR first comp

# Load NYComplaintFilings.csv dataset
nysubmissions2 = pd.read_csv(r'../results/NYSubmissions.csv')

# add column that counts the total number of submissions associated with that one CASENBR
counts = nysubmissions2['CASENBR'].value_counts()
nysubmissions2['subcount'] = nysubmissions2['CASENBR'].map(counts)

# Add a column to count the total number of "Rejected" submissions for each CASENBR
rejected_counts = nysubmissions2[nysubmissions2['CDSTATUS'] == 'REJ'].groupby('CASENBR').size()
nysubmissions2['rejected_count'] = nysubmissions2['CASENBR'].map(rejected_counts).fillna(0).astype(int)

# Add a column to count the total number of "Closed Denied" and "Closed Other" submissions for each CASENBR
closed_denied_other_counts = nysubmissions2[nysubmissions2['CDSTATUS'].isin(['CLD', 'CLO'])].groupby('CASENBR').size()
nysubmissions2['closed_denied_other_count'] = nysubmissions2['CASENBR'].map(closed_denied_other_counts).fillna(0).astype(int)

# Add a column to count the total number of "Closed Granted" and "Accepted" submissions for each CASENBR
closed_denied_other_counts = nysubmissions2[nysubmissions2['CDSTATUS'].isin(['CLG', 'ACC'])].groupby('CASENBR').size()
nysubmissions2['closed_granted_accepted_count'] = nysubmissions2['CASENBR'].map(closed_denied_other_counts).fillna(0).astype(int)

# ensure 'sitdtrcv' and 'sdtstat' are in datetime format
nysubmissions2['sitdtrcv'] = pd.to_datetime(nysubmissions2['sitdtrcv'], errors='coerce')
nysubmissions2['sdtstat'] = pd.to_datetime(nysubmissions2['sdtstat'], errors='coerce')

# find the earliest sitdtrcv date for each CASENBR
earliest_dates = nysubmissions2.groupby('CASENBR')['sitdtrcv'].min().reset_index()
earliest_dates.rename(columns={'sitdtrcv': 'earliest_sitdtrcv'}, inplace=True)

# find the latest sdtstat date for each CASENBR
latest_dates = nysubmissions2.groupby('CASENBR')['sdtstat'].max().reset_index()
latest_dates.rename(columns={'sdtstat': 'latest_sdtstat'}, inplace=True)

# merge the earliest and latest dates
merged_dates = pd.merge(earliest_dates, latest_dates, on='CASENBR')

# calculate the number of days between the earliest sitdtrcv and the latest sdtstat
merged_dates['days_between'] = (merged_dates['latest_sdtstat'] - merged_dates['earliest_sitdtrcv']).dt.days

# merge the calculated dates back into the original dataset
nysubmissionsenriched = pd.merge(
    nysubmissions2,
    merged_dates,
    on='CASENBR',
    how='left' # Retain all original rows
)

# save the updated dataset as NYSubmissionsEnriched.csv
nysubmissionsenriched.to_csv("../results/NYSubmissionsEnriched.csv", index=False)

```

CASENBR	subcount	rejected_count	closed_denied_other_count	closed_granted_accepted_count	earliest_sitdtrcv	latest_sdtstat	days_between
220246	1	0	0	1	2000-08-28	2000-09-13	16
220264	1	0	1	0	2000-08-16	2000-08-30	14
220279	4	3	1	0	2000-08-24	2000-11-14	82
220279	4	3	1	0	2000-08-24	2000-11-14	82
220279	4	3	1	0	2000-08-24	2000-11-14	82
220279	4	3	1	0	2000-08-24	2000-11-14	82
220320	1	1	0	0	2000-08-25	2000-08-29	4
220367	1	0	1	0	2000-08-24	2000-09-21	28
220432	2	1	1	0	2000-08-30	2000-10-10	41
220432	2	1	1	0	2000-08-30	2000-10-10	41
220454	4	2	2	0	2000-08-29	2001-02-01	156
220454	4	2	2	0	2000-08-29	2001-02-01	156
220454	4	2	2	0	2000-08-29	2001-02-01	156
220454	4	2	2	0	2000-08-29	2001-02-01	156
220455	7	7	0	0	2000-08-28	2000-12-12	106
220455	7	7	0	0	2000-08-28	2000-12-12	106
220455	7	7	0	0	2000-08-28	2000-12-12	106
220455	7	7	0	0	2000-08-28	2000-12-12	106
220455	7	7	0	0	2000-08-28	2000-12-12	106
220455	7	7	0	0	2000-08-28	2000-12-12	106
220459	4	2	2	0	2000-08-28	2001-01-03	128
220459	4	2	2	0	2000-08-28	2001-01-03	128
220459	4	2	2	0	2000-08-28	2001-01-03	128
220459	4	2	2	0	2000-08-28	2001-01-03	128
220478	1	0	1	0	2000-08-29	2000-09-12	14

```

# STEP 3: Create expanded dataset of NYSubmissionsEnriched.csv with codes translated for easier use

# import code csvs
complaintcodes = pd.read_csv(r'../data/cdsub1cb_ConcatSubjectCodes.csv')
facilitycodes = pd.read_csv(r'../data/CDFC_FacilityCodes.csv')
statuscodes = pd.read_csv(r'../data/CDSTATUS_CaseStatusCodes.csv')
orglevelcodes = pd.read_csv(r'../data/ITERLVL_OrgLevelCodes.csv')
statusreasoncodes = pd.read_csv(r'../data/STATRSN_StatusReasonCodes.csv')
columnncodes = pd.read_csv(r'../data/ColumnCodes.csv')

nysubmissionsrenriched2 = nysubmissionsenriched.copy()

# duplicate the cdsub1cb column to create cdsub1cbTEXT
nysubmissionsrenriched2.loc[:, 'cdsub1cbTEXT'] = nysubmissionsrenriched2['cdsub1cb']

# find and replace
nysubmissionsrenriched2.loc[:, 'cdsub1cbTEXT'] = nysubmissionsrenriched2['cdsub1cbTEXT'].map(complaintcodes.set_index('Code').get)
nysubmissionsrenriched2.loc[:, 'CDFCLEVN'] = nysubmissionsrenriched2['CDFCLEVN'].map(facilitycodes.set_index('FacID').get)
nysubmissionsrenriched2.loc[:, 'CDFCLRCV'] = nysubmissionsrenriched2['CDFCLRCV'].map(facilitycodes.set_index('FacID').get)
nysubmissionsrenriched2.loc[:, 'CDOFCRCV'] = nysubmissionsrenriched2['CDOFCRCV'].map(facilitycodes.set_index('FacID').get)
nysubmissionsrenriched2.loc[:, 'ITERLVL'] = nysubmissionsrenriched2['ITERLVL'].map(orglevelcodes.set_index('Code').get)
nysubmissionsrenriched2.loc[:, 'CDSTATUS'] = nysubmissionsrenriched2['CDSTATUS'].map(statuscodes.set_index('Code').get)
nysubmissionsrenriched2.loc[:, 'STATRSN1'] = nysubmissionsrenriched2['STATRSN1'].map(statusreasoncodes.set_index('Reason').get)
nysubmissionsrenriched2.loc[:, 'STATRSN2'] = nysubmissionsrenriched2['STATRSN2'].map(statusreasoncodes.set_index('Reason').get)
nysubmissionsrenriched2.loc[:, 'STATRSN3'] = nysubmissionsrenriched2['STATRSN3'].map(statusreasoncodes.set_index('Reason').get)
nysubmissionsrenriched2.loc[:, 'STATRSN4'] = nysubmissionsrenriched2['STATRSN4'].map(statusreasoncodes.set_index('Reason').get)
nysubmissionsrenriched2.loc[:, 'STATRSN5'] = nysubmissionsrenriched2['STATRSN5'].map(statusreasoncodes.set_index('Reason').get)

nysubmissionsrenriched2.loc[:, 'sdtdue'] = nysubmissionsrenriched2['sdtdue'].fillna('rejected')

nysubmissionsrenriched2.loc[:, 'accept'] = nysubmissionsrenriched2['accept'].replace({0: 'no', 1: 'yes'})
nysubmissionsrenriched2.loc[:, 'reject'] = nysubmissionsrenriched2['reject'].replace({0: 'no', 1: 'yes'})
nysubmissionsrenriched2.loc[:, 'deny'] = nysubmissionsrenriched2['deny'].replace({0: 'no', 1: 'yes'})
nysubmissionsrenriched2.loc[:, 'grant'] = nysubmissionsrenriched2['grant'].replace({0: 'no', 1: 'yes'})
nysubmissionsrenriched2.loc[:, 'other'] = nysubmissionsrenriched2['other'].replace({0: 'no', 1: 'yes'})
nysubmissionsrenriched2.loc[:, 'submit'] = nysubmissionsrenriched2['submit'].replace({0: 'no', 1: 'yes'})
nysubmissionsrenriched2.loc[:, 'filed'] = nysubmissionsrenriched2['filed'].replace({0: 'no', 1: 'yes'})
nysubmissionsrenriched2.loc[:, 'closed'] = nysubmissionsrenriched2['closed'].replace({0: 'no', 1: 'yes'})

columnncodes_dict = dict(zip(columnncodes['Code'], columnncodes['Text']))

nysubmissionsrenriched2.rename(columns=columnncodes_dict, inplace=True)

nysubmissionsrenriched2.to_csv('../results/NYSubmissionsEnrichedExpanded.csv', index=False)

```

Status Code	Reason Code	Text
CLD	DNY	DENIED: REQUEST OR APPEAL DENIED SUBSTANTIALLY
CLG	GRT	GRANTED: REQUEST OR APPEAL GRANTED SUBSTANTIALLY
CLG	PAR	GRANTED: REQUEST OR APPEAL PARTIALLY GRANTED
CLG	PVG	GRANTED: REQUEST OR APPEAL PREVIOUSLY GRANTED
CLO	ISJ	CLOSED: IMPROPER SUBJECT MATTER.
CLO	MOT	CLOSED: REQUEST OR APPEAL IS MOOT.
CLO	OTH	OTHER/SEE REMARKS
CLO	REP	CLOSED: REQUEST OR APPEAL DENIED AS REPEATED
CLO	WDN	CLOSED: WITHDRAWN AT INMATE'S REQUEST.
CLO	XPL	CLOSED: INFORMATION OR EXPLANATION ONLY.
REJ	ATT	REJECTED: You did not submit the correct number
REJ	CDC	REJECTED: YOU MUST WAIT FOR THE CCM'S DECISION
REJ	CON	REJECTED: YOU DID NOT SUBMIT YOUR REQUEST
REJ	CPG	REJECTED: YOU DID NOT SUBMIT THE PROPER INFORMATION
REJ	DHO	REJECTED: YOU DID NOT PROVIDE A COPY OF THE DOCUMENT
REJ	DHR	REJECTED: YOU MUST WAIT FOR THE DHO'S DECISION
REJ	DIR	REJECTED: Concur with rationale of Regional Office
REJ	ENG	REJECTED: You may request staff assistance in preparing your request
REJ	EXH	REJECTED: YOU DID NOT SUBMIT THE PROPER INFORMATION
REJ	FRM	REJECTED: YOU DID NOT SUBMIT YOUR REQUEST
REJ	IAT	REJECTED: YOU DID NOT PROVIDE A COPY OF THE DOCUMENT
REJ	IMP	REJECTED: IMPROPER SUBJECT MATTER (FOI/POLICY)
REJ	INF	REJECTED: YOU DID NOT ATTEMPT INFORMAL RESOLUTION
REJ	INS	REJECTED: YOU MUST FIRST FILE A BP-9 REQUEST
REJ	IPO	REJECTED: YOU DID NOT PROVIDE A COPY OF THE DOCUMENT

Facility_Code	Facility_Name
CAC	ADAMS COUNTY CORR CTR CCM
CAT	ATLANTA CCM
CBR	BALTIMORE CCM
CBS	BIG SPRING CCM
CBN	BOSTON CCM
CCF	CALIFORNIA CITY CCM
CCH	CHICAGO CCM
CCB	CIBOLA COUNTY CCM
CCN	CINCINNATI CCM
CRJ	D. RAY JAMES CORR FCL CCM
CDL	DALBY CCM
CDA	DALLAS CCM
CDE	DENVER CCM
CDT	DETROIT CCM
CDC	DIST OF COLUMBIA CCM
CED	EDEN CCM
CEP	EL PASO CCM
COP	GREATER PLAINING CCM

Code	Text
ACC	Accepted
CLD	Closed Denied
CLG	Closed Granted
CLO	Closed Other
REJ	Rejected



Remedy Case Number	Organizational level of submission	Facility of Occurrence	Facility Received	Receiving Office	Case Status	Status Reason 1	Status Reason 2
157212	Agency	BROOKLYN MDC	BROOKLYN MDC	BOP	Rejected	REJECTED: YOU SUBMITTED YOUR REQUEST OR APPEAL TO THE WRONG LEVEL OR WRONG OFFICE.	OTHER/SEE REMARKS
182837	Region	OTISVILLE FCI	OTISVILLE FCI	NER	Rejected	REJECTED: YOUR APPEAL IS UNTIMELY. REGIONAL APPEALS MUST BE RECEIVED WITHIN 20 DAYS OF THE WARDEN'S OR CCM'S RESPONSE. THIS TIME LIMIT INCLUDES MAIL TIME.	
182837	Agency	OTISVILLE FCI	OTISVILLE FCI	BOP	Rejected	REJECTED: YOUR APPEAL IS UNTIMELY. REGIONAL APPEALS MUST BE RECEIVED WITHIN 20 DAYS OF THE WARDEN'S OR CCM'S RESPONSE. THIS TIME LIMIT INCLUDES MAIL TIME.	
184239	Region	NEW YORK MCC	BROOKLYN MDC	NER	Rejected	OTHER/SEE REMARKS	
184239	Agency	NEW YORK MCC	BROOKLYN MDC	BOP	Rejected	OTHER/SEE REMARKS	REJECTED: YOU SUBMITTED YOUR REQUEST OR APPEAL TO THE WRONG LEVEL OR WRONG OFFICE.
185448	Region	RAY BROOK FCI	RAY BROOK FCI	NER	Rejected	REJECTED: YOUR APPEAL IS UNTIMELY. REGIONAL APPEALS MUST BE RECEIVED WITHIN 20 DAYS OF THE WARDEN'S OR CCM'S RESPONSE. THIS TIME LIMIT INCLUDES MAIL TIME.	REJECTED: YOU SUBMITTED YOUR REQUEST OR APPEAL TO THE WRONG LEVEL OR WRONG OFFICE.
185448	Agency	RAY BROOK FCI	RAY BROOK FCI	BOP	Rejected	REJECTED: YOUR APPEAL IS UNTIMELY. REGIONAL APPEALS MUST BE RECEIVED WITHIN 20 DAYS OF THE WARDEN'S OR CCM'S RESPONSE. THIS TIME LIMIT INCLUDES MAIL TIME.	
193467	Facility	RAY BROOK FCI	RAY BROOK FCI	RAY BROOK FCI	Rejected	REJECTED: YOUR REQUEST IS UNTIMELY. INSTITUTION AND CCC REQUESTS MUST BE RECEIVED WITHIN 20 DAYS OF THE EVENT COMPLAINED ABOUT.	OTHER/SEE REMARKS
194199	Agency	RAY BROOK FCI	LEWISBURG USP	BOP	Rejected	REJECTED: YOUR REQUEST IS UNTIMELY. INSTITUTION AND CCC REQUESTS MUST BE RECEIVED WITHIN 20 DAYS OF THE EVENT COMPLAINED ABOUT.	OTHER/SEE REMARKS

Status Reason for 20523 Submissions with "Rejected" Status

Status Reason	Count
OTHER/SEE REMARKS	8306
REJECTED: YOU MAY RESUBMIT YOUR APPEAL IN PROPER FORM WITHIN 10 DAYS OF THE DATE OF THIS REJECTION NOTICE.	5534
REJECTED: YOU DID NOT ATTEMPT INFORMAL RESOLUTION PRIOR TO SUBMISSION OF ADMINISTRATIVE REMEDY, OR YOU DID NOT PROVIDE THE NECESSARY EVIDENCE OF YOUR ATTEMPT AT INFORMAL RESOLUTION.	3162
REJECTED: YOU MAY RESUBMIT YOUR REQUEST IN PROPER FORM WITHIN 5 DAYS OF THE DATE OF THIS REJECTION NOTICE.	3154
REJECTED: YOU MUST FIRST FILE A BP-9 REQUEST THROUGH THE INSTITUTION FOR THE WARDEN'S REVIEW AND RESPONSE BEFORE FILING AN APPEAL AT THIS LEVEL.	2442
REJECTED: YOUR APPEAL IS UNTIMELY. REGIONAL APPEALS MUST BE RECEIVED WITHIN 20 DAYS OF THE WARDEN'S OR CCM'S RESPONSE. THIS TIME LIMIT INCLUDES MAIL TIME.	2089
REJECTED: YOU SUBMITTED YOUR REQUEST OR APPEAL TO THE WRONG LEVEL OR WRONG OFFICE.	1772
REJECTED: YOU MAY RESUBMIT YOUR APPEAL IN PROPER FORM WITHIN 15 DAYS OF THE DATE OF THIS REJECTION NOTICE.	1688
REJECTED: YOU MAY ONLY SUBMIT UP TO ONE LETTER-SIZE (8 1/2" X 11") CONTINUATION PAGE.	1584
REJECTED: YOU DID NOT PROVIDE A COPY OF YOUR INSTITUTION ADMINISTRATIVE REMEDY REQUEST (BP-9), OR A RECEIPT, OR YOU DID NOT PROVIDE A VERIFIED PHOTOCOPY.	1501
REJECTED: YOUR REQUEST IS UNTIMELY. INSTITUTION AND CCC REQUESTS MUST BE RECEIVED WITHIN 20 DAYS OF THE EVENT COMPLAINED ABOUT.	1427
REJECTED: YOU DID NOT SUBMIT YOUR REQUEST OR APPEAL ON THE PROPER FORM (BP-9, BP-10, BP-11).	1220
REJECTED: YOU DID NOT SUBMIT YOUR REQUEST THROUGH YOUR COUNSELOR, OR OTHER AUTHORIZED PERSON.	1216
REJECTED: YOU DID NOT SUBMIT THE PROPER NUMBER OF CONTINUATION PAGES. YOU MUST SUBMIT ONE COPY AT THE WARDEN'S LEVEL; TWO COPIES AT THE REGIONAL DIRECTOR'S LEVEL; AND THREE COPIES AT THE CENTRAL OFFICE LEVEL.	1186
REJECTED: All four pages of your (BP-9) (BP-10) (BP-11) form must be legible and worded the same. Photocopies of the form will not be accepted.	1112
REJECTED: YOU DID NOT SUBMIT A COMPLETE SET (4 CARBONIZED COPIES) OF THE REQUEST OR APPEAL FORM.	1022
REJECTED: THE ISSUE YOU RAISE IS NOT A SENSITIVE ISSUE. YOUR REQUEST/APPEAL IS NOT BEING RETURNED TO YOU IN ACCORDANCE WITH POLICY.	955

Status Reason	Count
REJECTED: YOU DID NOT PROVIDE A COPY OF THE DHO REPORT; OR YOU DID NOT OTHERWISE IDENTIFY THE CHARGES AND DATE OF THE DHO ACTION YOU ARE APPEALING.	795
REJECTED: YOU MUST PROVIDE MORE SPECIFIC INFORMATION ABOUT YOUR REQUEST/APPEAL SO THAT IT MAY BE CONSIDERED.	794
REJECTED: YOU DID NOT SIGN YOUR REQUEST OR APPEAL.	736
REJECTED: YOU ARE APPEALING MORE THAN ONE INCIDENT REPORT (INCIDENT NUMBER) ON A SINGLE APPEAL FORM. YOU MUST FILE A SEPARATE APPEAL FOR EACH INCIDENT REPORT (INCIDENT NUMBER) YOU WISH TO APPEAL.	578
REJECTED: YOU DID NOT PROVIDE A COPY OF THE REGIONAL APPEAL, OR A RECEIPT, OR YOU DID NOT PROVIDE A VERIFIED PHOTOCOPY.	573
REJECTED: Provide staff verification stating reason untimely filing was not your fault.	564
REJECTED: Concur with rationale of Regional Office and/or institution for rejection. Follow directions provided on prior rejection notices.	504
REJECTED: YOU DID NOT PROVIDE A COPY OF THE ATTACHMENT(S) TO YOUR INSTITUTION ADMINISTRATIVE REMEDY REQUEST (BP-9).	471
REJECTED: YOUR APPEAL IS UNTIMELY. CENTRAL OFFICE APPEALS MUST BE RECEIVED WITHIN 30 DAYS OF THE REGIONAL DIRECTOR'S RESPONSE. THIS TIME LIMIT INCLUDES MAIL TIME.	342
REJECTED: Your appeal of the rejection is untimely. Resubmissions are due within: 5 days (institution); 10 days (CCM or Regional Office); 15 days (Central Office). Submit staff memo on BOP letterhead stating reason untimely filing wasn't your fault.	284
REJECTED: You did not submit the correct number of copies of the attachments (new documentation not considered by lower levels). 2 at institution; 3 at region; and 4 at Central Office.	242
REJECTED: YOU MUST WAIT FOR THE DHO'S DECISION OF THE DHO HEARING BEFORE YOU MAY APPEAL TO THE REGIONAL OFFICE. RESUBMIT YOUR APPEAL TO THIS OFFICE WITHIN 20 DAYS OF THE DATE YOU RECEIVE THE DHO'S DECISION.	149
REJECTED: YOUR REQUEST CONTAINS GRATUITOUS OBSCENE OR ABUSIVE LANGUAGE.	34
REJECTED: IMPROPER SUBJECT MATTER (FOI/PA REQUEST,ETC.)	23
REJECTED: You may request staff assistance in preparing your request or appeal in English.	21
REJECTED: Due to your allegations, your appeal is being forwarded to another department for review; however, your appeal was retained in accordance with policy.	20

```

# STEP 4: Create subset of data which only has unique complaints. It will be the record with the most recent status
# load dataset
nysubmissionedenriched3 = pd.read_csv(r'../results/NYSubmissionsEnriched.csv')

# sort 'sitedtrcv' (submission date) in descending order
nysubmissionedenriched3 = nysubmissionedenriched3.sort_values(by='sitedtrcv', ascending=False)

# Drop duplicate CASENBRs, keeping the most recent (highest sitedtrcv)
unique_nysubmissionedenriched1 = nysubmissionedenriched3.drop_duplicates(subset='CASENBR', keep='first')

unique_nysubmissionedenriched1.to_csv("../results/UniqueNYSubmissionedenriched.csv", index=False)

```

Python

```

# STEP 5: Create expanded dataset of NYSubmissionsEnriched.csv with codes translated for easier use

# import code csvs
complaintcodes = pd.read_csv('../data/cdsuibcb_ConcatSubjectCodes.csv')
facilitycodes = pd.read_csv('../data/CDFC_FacilityCodes.csv')
statuscodes = pd.read_csv('../data/CDSTATUS_CaseStatusCodes.csv')
orglevelcodes = pd.read_csv('../data/ITERLVL_OrgLevelCodes.csv')
statusreasoncodes = pd.read_csv('../data/STATRSN_StatusReasonCodes.csv')
columncodes = pd.read_csv('../data/ColumnCodes.csv')

unique_nysubmissionsrenriched2 = unique_nysubmissionedenriched1.copy()

# duplicate the cdsuibcb column to create cdsuibcbTEXT
unique_nysubmissionsrenriched2.loc[:, 'cdsuibcbTEXT'] = unique_nysubmissionsrenriched2['cdsuibcb']

# find and replace
unique_nysubmissionsrenriched2.loc[:, 'cdsuibcbTEXT'] = unique_nysubmissionsrenriched2['cdsuibcbTEXT'].map(complaintcodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'CDFCLEVN'] = unique_nysubmissionsrenriched2['CDFCLEVN'].map(facilitycodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'CDFCLRCV'] = unique_nysubmissionsrenriched2['CDFCLRCV'].map(statuscodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'CDFCRCV'] = unique_nysubmissionsrenriched2['CDFCRCV'].map(facilitycodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'ITERLVL'] = unique_nysubmissionsrenriched2['ITERLVL'].map(orglevelcodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'CDSTATUS'] = unique_nysubmissionsrenriched2['CDSTATUS'].map(statuscodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'STATRSN1'] = unique_nysubmissionsrenriched2['STATRSN1'].map(statusreasoncodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'STATRSN2'] = unique_nysubmissionsrenriched2['STATRSN2'].map(statusreasoncodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'STATRSN3'] = unique_nysubmissionsrenriched2['STATRSN3'].map(statusreasoncodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'STATRSN4'] = unique_nysubmissionsrenriched2['STATRSN4'].map(statusreasoncodes.set_index('Code').Text)
unique_nysubmissionsrenriched2.loc[:, 'STATRSN5'] = unique_nysubmissionsrenriched2['STATRSN5'].map(statusreasoncodes.set_index('Code').Text)

unique_nysubmissionsrenriched2.loc[:, 'sdtdue'] = unique_nysubmissionsrenriched2['sdtdue'].fillna('rejected')

unique_nysubmissionsrenriched2.loc[:, 'accept'] = unique_nysubmissionsrenriched2['accept'].replace({0: 'no', 1: 'yes'})
unique_nysubmissionsrenriched2.loc[:, 'reject'] = unique_nysubmissionsrenriched2['reject'].replace({0: 'no', 1: 'yes'})
unique_nysubmissionsrenriched2.loc[:, 'deny'] = unique_nysubmissionsrenriched2['deny'].replace({0: 'no', 1: 'yes'})
unique_nysubmissionsrenriched2.loc[:, 'grant'] = unique_nysubmissionsrenriched2['grant'].replace({0: 'no', 1: 'yes'})
unique_nysubmissionsrenriched2.loc[:, 'other'] = unique_nysubmissionsrenriched2['other'].replace({0: 'no', 1: 'yes'})
unique_nysubmissionsrenriched2.loc[:, 'submit'] = unique_nysubmissionsrenriched2['submit'].replace({0: 'no', 1: 'yes'})
unique_nysubmissionsrenriched2.loc[:, 'filed'] = unique_nysubmissionsrenriched2['filed'].replace({0: 'no', 1: 'yes'})
unique_nysubmissionsrenriched2.loc[:, 'closed'] = unique_nysubmissionsrenriched2['closed'].replace({0: 'no', 1: 'yes'})

columncodes_dict = dict(zip(columncodes['Code'], columncodes['Text']))

unique_nysubmissionsrenriched2.rename(columns=columncodes_dict, inplace=True)

unique_nysubmissionsrenriched2.to_csv('../results/UniqueNYSubmissionsEnrichedExpanded.csv', index=False)

```

Python

```
freq_table <- table(uniquedata$Remedy.Subject.Code.Translation)

percentages <- prop.table(freq_table) * 100

result <- data.frame(
  Value = names(freq_table),
  Count = as.vector(freq_table),
  Percentage = round(as.vector(percentages), 2)
)

result <- result[order(result$Count, decreasing = TRUE), ]

write.csv(result, "../results/output_percentages_secondaryreason.csv", row.names = FALSE)
```

✓ 0.0s

## output\_percentages\_primaryreason

Value	Count	Percentage
DHO/CDC/CONT. HOUSING APPEALS	6092	23.32
STAFF/OTHERS - COMPLAINTS	3033	11.61
MEDICAL-EXC. FORCED TREATMENT	2965	11.35
UDC/RRC ACTIONS	2813	10.77
OPERATIONS, INSTITUTION	1498	5.74
SENTENCE COMPUTATION	1164	4.46
SPECIAL HOUSING UNITS	1083	4.15
CLASSIFICATION MATTERS	986	3.78
LEGAL MATTERS	977	3.74
PROGRAMS, INSTITUTION	677	2.59
TRANSFER - REQUEST/OBJECTION	628	2.4
MAIL COMMUNICATION	607	2.32
JAIL TIME CREDIT	595	2.28
COMMUNITY/PRE-RELEASE PROGRAMS	561	2.15
VISITING	389	1.49
FOOD	378	1.45
DENTAL CARE	289	1.11
WORK ASSIGNMENTS	289	1.11
DISABILITY-PHYSICAL OR MENTAL	244	0.93
COMMUNICATION-OTHER THAN MAIL	208	0.8
EDUCATION, RECREATION, LEISURE	206	0.79
RECORDS MANAGEMENT	169	0.65
MENTAL HEALTH CARE	86	0.33
SEARCHES AND USE OF RESTRAINTS	83	0.32
OTHER STATUTORILY-MANDATED PROCEDURES	63	0.24
PREA (SEXUAL ABUSE BY STAFF OR INMATE)	17	0.07
CONTROL UNIT	11	0.04
MEDICAL TREATMENT - FORCED	7	0.03

## output\_percentages\_secondaryreason

Value	Count	Percentage
DHO/CDC/CONT. HOUSING APPEALS: DHO APPEAL - COMBINED (PROCEDURES, EVIDENCE & SANCTIONS)	3351	12.83
UDC/RRC ACTIONS: UDC ACTION	2790	10.68
STAFF/OTHERS - COMPLAINTS: UNPROFESSIONAL, INAPPROPRIATE CONDUCT OR MISCONDUCT BY STAFF	1488	5.7
MEDICAL-EXC. FORCED TREATMENT: MEDICAL CARE - DELAY OR ACCESS TO	1327	5.08
STAFF/OTHERS - COMPLAINTS: OTHER COMPLAINT AGAINST STAFF	1096	4.2
DHO/CDC/CONT. HOUSING APPEALS: CONTRACT COMM. CORR. CTR. DISCIPLINE - MAJOR SANCTIONS	1031	3.95
SENTENCE COMPUTATION: OTHER SENTENCE COMPUTATION	990	3.79
DHO/CDC/CONT. HOUSING APPEALS: DHO APPEAL - SANCTION	689	2.64
MEDICAL-EXC. FORCED TREATMENT: MEDICAL CARE - IMPROPER OR INADEQUATE	583	2.23
SPECIAL HOUSING UNITS: ADMINISTRATIVE DETENTION - PLACEMENT, REVIEWS, RELEASE	496	1.9
DHO/CDC/CONT. HOUSING APPEALS: DHO APPEAL - EVIDENCE	482	1.85
OPERATIONS, INSTITUTION: PERSONAL PROPERTY - INCL. CONFISCATION OR DESTRUCTION	427	1.63
OPERATIONS, INSTITUTION: SAFETY, SANITATION, ENVIRONMENTAL CONDITIONS	399	1.53
TRANSFER - REQUEST/OBJECTION: TRANSFER - OTHER	372	1.42
MEDICAL-EXC. FORCED TREATMENT: OTHER MEDICAL MATTERS	358	1.37
DHO/CDC/CONT. HOUSING APPEALS: DHO APPEAL - PROCEDURES	350	1.34
COMMUNITY/PRE-RELEASE PROGRAMS: RESIDENTIAL REENTRY CENTER REFERRALS	316	1.21
OPERATIONS, INSTITUTION: OTHER OPERATIONS	310	1.19
JAIL TIME CREDIT: OTHER JAIL TIME CREDIT	293	1.12
SPECIAL HOUSING UNITS: OTHER SPECIAL HOUSING UNIT MATTERS	291	1.11
CLASSIFICATION MATTERS: COMPASSIONATE RELEASE REQUEST	289	1.11
MAIL COMMUNICATION: OTHER MAIL COMPLAINTS	282	1.08
JAIL TIME CREDIT: CREDIT FOR TIME SPENT IN JAIL	279	1.07
MEDICAL-EXC. FORCED TREATMENT: PRESCRIPTIONS, MEDICATION	272	1.04
LEGAL MATTERS: ADMINISTRATIVE REMEDY PROCEDURES	260	1

```

unique_submissions <- n_distinct(data$Remedy.Case.Number)
print(unique_submissions)

# Count unique submissions by Facility
submission_count_by_facility <- data %>%
  group_by(Facility.of.Occurrence) %>%
  summarise(unique_submissions = n_distinct(Remedy.Case.Number))

# Create a bar plot
ggplot(submission_count_by_facility, aes(x = Facility.of.Occurrence, y = unique_submissions)) +
  geom_bar(stat = "identity") +
  labs(title = "Unique Submissions by Facility", x = "Facility", y = "Unique Submissions") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))

```

/ 0.2s

