CA2: NI Postcode and Crime data

Section-1:

Introduction:

The NI post code and crime data are given to work on the CA2. NI post code datasets consists of the postcode details of the Northern Ireland like Organisation name, Building name, number, Primary thorfare, locality, town etc. Crime data consist of the information on crimes of the Northern Ireland. Assignment has some task to do the for the analysis in order to obtain the final output.

Section 1:

A) The first task of the assignment is to show the total number of rows, the structure of the data and the first 10 rows of the data frame containing all of the NI postcode data. The code for the above task is shown in the fig.1.

```
# Loading the CSV file by placing in the working directory
NIPOSTCODE = read.csv("NIPostcodes.csv", header = FALSE)
# Displaying the total number of rows
nrow(NIPOSTCODE)
# Structure of the data
str(NIPOSTCODE)
# Displaying the fisrt 10 rows of the data frame
head(NIPOSTCODE,10)
```

Fig.1: Loading the CSV file, showing total number of rows and first 10 rows of the data frame

The CSV file is placed in the working directory to access publicly. The dataset is loaded by using the read.csv function and number of rows are displayed using the function nrow. Head function gives you limited number of rows in this task we need only 10. The structure and outputs of the task are shown in the below figure. The Header = false in the first line of the code is used because in the figure.2 we can observe that the first column is treated as the header to ignore this header must be false.

```
Console   Terminal ×   Jobs ×
~/L00151024/
> # Displaying the total number of rows
> nrow(NIPOSTCODE)
[1] 943033
> # Displaying the fisrt 10 rows of the data frame
> head(NIPOSTCODE,10)
   X    X.1 X.2 X17         HIGH.ROAD X.3 X.4       X.5 MULLAGHACALL.NORTH PORTSTEWART LONDONDERRY BT557BG X281855 X438598 X1
1                15 CONVENTION AVENUE                   MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557BW  281892  438228  2
2                13      STATION ROAD                   MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557HH  282306  438587  3
3                99    OLD COACH ROAD                   MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557HW  282419  438387  4
4                20       BREDA COURT                                BREDA     BELFAST        DOWN  BT86JB  335367  369985  5
5                11  UPPER HEATHMOUNT                   MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557AR  281719  438366  6
6                86        LEVER ROAD                   MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557EE  282080  438424  7
7               112    OLD COACH ROAD                   MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557HW  282524  438243  8
8               134    WHITEPARK ROAD         BALLINTOY  BALLINTOY DEMESNE BALLYCASTLE      ANTRIM BT546ND  303527  444150  9
9    FLAT 3      16      STATION ROAD                   MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557DA  282128  438612 10
10               15 HEATHERLEA AVENUE                   MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557HF  282220  438467 11
```

Fig.3: Error in the output

```
> str(NIPOSTCODE)
'data.frame':   943033 obs. of  15 variables:
 $ X                 : Factor w/ 40859 levels "","" ASCERT"," BALLYMAC HOTEL",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ X.1               : Factor w/ 6186 levels "","'RETURN APARTMENT' B",..: 1 1 1 1 1 1 1 1 3166 1 ...
 $ X.2               : Factor w/ 12215 levels "","'ARDEEVIN'",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ X17               : Factor w/ 5831 levels ""," 25C",".",..: 804 564 5076 1368 298 4701 348 612 926 804 ...
 $ HIGH.ROAD         : Factor w/ 24540 levels "","ABBACY ROAD",..: 6072 21646 18024 3392 23334 14504 18024 23992 21646 11945 ...
 $ X.3               : Factor w/ 453 levels "","AN BEALACH LEATHAN",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ X.4               : Factor w/ 288 levels "","ABBEY ROAD",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ X.5               : Factor w/ 676 levels "","ABBEY BUSINESS PARK",..: 1 1 1 1 1 1 1 55 1 1 ...
 $ MULLAGHACALL.NORTH: Factor w/ 7705 levels "ABBEY PARK","ABOCURRAGH",..: 6145 6145 6145 1532 6145 6145 6145 482 6145 6145 ...
 $ PORTSTEWART       : Factor w/ 314 levels "","AGHAGALLON",..: 271 271 271 46 271 271 271 27 271 271 ...
 $ LONDONDERRY       : Factor w/ 6 levels "ANTRIM","ARMAGH",..: 5 5 5 3 5 5 5 1 5 5 ...
 $ BT557BG           : Factor w/ 47931 levels "","BR925BN","BT00BT",..: 30877 30962 30972 43705 30854 30904 30972 30557 30881 30960 ...
 $ X281855           : int  281892 282306 282419 335367 281719 282080 282524 303527 282128 282220 ...
 $ X438598           : int  438228 438587 438387 369985 438366 438424 438243 444150 438612 438467 ...
 $ X1                : int  2 3 4 5 6 7 8 9 10 11 ...
```

Fig.4: Structure of the data frame

Output:

```
> head(NIPOSTCODE,10)
   V1     V2 V3  V4                V5 V6 V7        V8                V9         V10         V11     V12    V13    V14 V15
1                17          HIGH ROAD                MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557BG 281855 438598   1
2                15 CONVENTION AVENUE                MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557BW 281892 438228   2
3                13      STATION ROAD                MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557HH 282306 438587   3
4                99    OLD COACH ROAD                MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557HW 282419 438387   4
5                20       BREDA COURT                             BREDA     BELFAST        DOWN  BT86JB 335367 369985   5
6                11  UPPER HEATHMOUNT                MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557AR 281719 438366   6
7                86        LEVER ROAD                MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557EE 282080 438424   7
8               112    OLD COACH ROAD                MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557HW 282524 438243   8
9               134    WHITEPARK ROAD       BALLINTOY  BALLINTOY DEMESNE BALLYCASTLE      ANTRIM BT546ND 303527 444150   9
10    FLAT 3     16      STATION ROAD                MULLAGHACALL NORTH PORTSTEWART LONDONDERRY BT557DA 282128 438612  10
```

Fig.5: Output without error in the headers

CSV files has 943033 row in total and the first 10 rows of the data frame are showed in the above figure. Now we can see that the headers are changed by keeping false.

B) The next task is to assign the relevant titles to each attribute in the data frame. Initially the header of the columns is named as x, x.1, x.2 and so on. We have changed with the relevant titles as shown in the below code and stored in the data frame col_names. Modified data is then moved to NIPOSTCODE and output can be displayed by using view(NIPOSTCODE).

```
# Assigning the relevant title for each attribute of the data
col_names = c("Organisation Name",
              "Sub-building Name",
              "Building Name",
              "Number",
              "Primary Thorfare",
              "Alt Thorfare",
              "Secondary Thorfare",
              "Locality",
              "Townland",
              "Town",
              "County",
              "Postcode",
              "x-coordinates",
              "y-coordinates",
              "Primary Key (identifier)")
colnames(NIPOSTCODE) <- col_names
View(NIPOSTCODE)
```

Fig.3: Changing the names of each attributes

Output:

	Organisation Name	Sub-building Name	Building Name	Number	Primary Thorfare	Alt Thorfare	Secondary Thorfare	Locality	Townland	Town	Cou
1				17	HIGH ROAD				MULLAGHACALL NORTH	PORTSTEWART	
2				15	CONVENTION AVENUE				MULLAGHACALL NORTH	PORTSTEWART	
3				13	STATION ROAD				MULLAGHACALL NORTH	PORTSTEWART	
4				99	OLD COACH ROAD				MULLAGHACALL NORTH	PORTSTEWART	
5				20	BREDA COURT				BREDA	BELFAST	
6				11	UPPER HEATHMOUNT				MULLAGHACALL NORTH	PORTSTEWART	
7				86	LEVER ROAD				MULLAGHACALL NORTH	PORTSTEWART	
8				112	OLD COACH ROAD				MULLAGHACALL NORTH	PORTSTEWART	
9				134	WHITEPARK ROAD			BALLINTOY	BALLINTOY DEMESNE	BALLYCASTLE	
10		FLAT 3		16	STATION ROAD				MULLAGHACALL NORTH	PORTSTEWART	
11				15	HEATHERLEA AVENUE				MULLAGHACALL NORTH	PORTSTEWART	
12				30	STATION ROAD				MULLAGHACALL NORTH	PORTSTEWART	
13				21	HIGH ROAD				MULLAGHACALL NORTH	PORTSTEWART	
14				1	PROSPECT ROAD				TULLAGHMURRY WEST	PORTSTEWART	

Fig.4: Change in the titles

C) In this task we need to replace and recode all the missing entries with a suitable identifier. In order to do this, we need to check the count of missing entries in each attribute by writing the code as shown in below figure.5

	Organisation Name	Sub-building Name	Building Name	Number	Primary Thorfare	Alt Thorfare	Secondary Thorfare	Locality	Townland	Tow
1				17	HIGH ROAD				MULLAGHACALL NORTH	P
2				15	CONVENTION AVENUE				MULLAGHACALL NORTH	P
3				13	STATION ROAD				MULLAGHACALL NORTH	P
4				99	OLD COACH ROAD				MULLAGHACALL NORTH	P
5				20	BREDA COURT				BREDA	B
6				11	UPPER HEATHMOUNT				MULLAGHACALL NORTH	P
7				86	LEVER ROAD				MULLAGHACALL NORTH	P
8				112	OLD COACH ROAD				MULLAGHACALL NORTH	P
9				134	WHITEPARK ROAD			BALLINTOY	BALLINTOY DEMESNE	B
10		FLAT 3		16	STATION ROAD				MULLAGHACALL NORTH	P
11				15	HEATHERLEA AVENUE				MULLAGHACALL NORTH	P
12				30	STATION ROAD				MULLAGHACALL NORTH	P
13				21	HIGH ROAD				MULLAGHACALL NORTH	P
14				1	PROSPECT ROAD				TULLAGHMURRY WEST	P
15				29	LEVER ROAD				MULLAGHACALL NORTH	P
16		APARTMENT B		27	PRINCESS STREET				PORT RUSH	P
17				94	OLD COACH ROAD				MULLAGHACALL NORTH	P
18				1	HARRYVILLE				MULLAGHACALL NORTH	P

Fig.6: Initial missing entries in the dataset

Now we need to check the count of missing entries of each attributes. The lappy is used in the code to iterate each column and check the missing entries.

```
# Checking the count of missing entries in each attribute
data.frame(lapply(NIPOSTCODE, function(x)sum(x ==""")))
```

Fig.5:

Output:

```
> # Checking the count of missing entries in each attribute
> data.frame(lapply(NIPOSTCODE, function(x)sum(x ==""")))
  Organisation.Name Sub.building.Name Building.Name Number Primary.Thorfare Alt.Thorfare Secondary.Thorfare Locality Townland  Town
1            890537             884099        895540  28753              470       921788             938400   856789        0 19872
  County Postcode x.coordinates y.coordinates Primary.Key..identifier.
1      0     8900             0             0                       0
>
```

Fig.6: Count of missing entries

Plotting a graph in order to justify my decision in choosing the best. The below code is used to plot the graph by using the summary function.

```
# Using library VIM
library(VIM)
# Plotting a graph to choose the best consideration for the missing entries
missing_values <- aggr(NIPOSTCODE, prop= FALSE, numbers = TRUE)
summary(missing_values)
```

Fig.7: using library vim

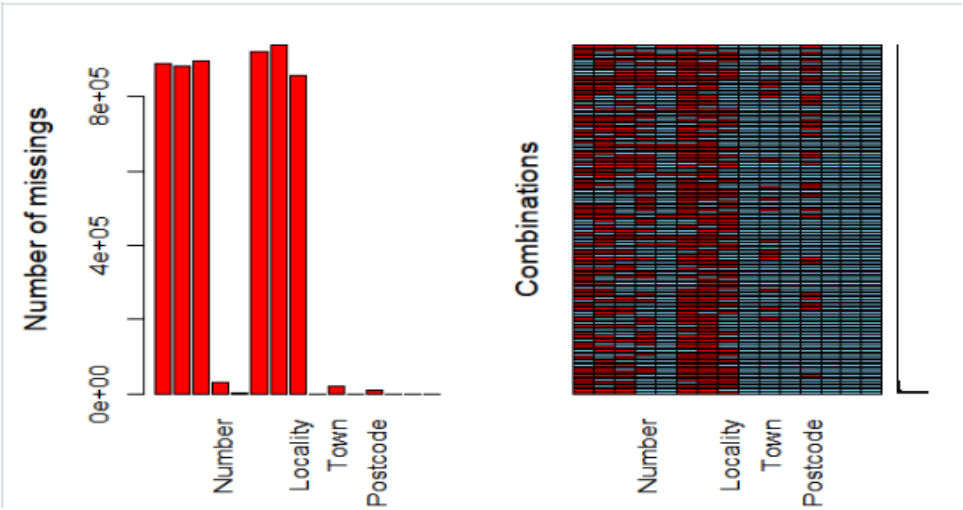


Fig.8: bar plot

While observing the graph I have decided to replace with NA to the missing entries because we cannot hardcode the data and some of the columns have missing values more than 50%. We cannot even remove the rows of the missing entries as they effect the other rows and lead to data loss.

```
# Replacing the missing values with the NA
NIPOSTCODE[NIPOSTCODE == ""] <- NA
# To view the data frame
View(NIPOSTCODE)
```

Fig.9: Recode and replace the missing entries

Output:

	Organisation Name	Sub-building Name	Building Name	Number	Primary Thorfare	Alt Thorfare	Secondary Thorfare	Locality	Townland
1	NA	NA	NA	17	HIGH ROAD	NA	NA	NA	MULLAGH
2	NA	NA	NA	15	CONVENTION AVENUE	NA	NA	NA	MULLAGH
3	NA	NA	NA	13	STATION ROAD	NA	NA	NA	MULLAGH
4	NA	NA	NA	99	OLD COACH ROAD	NA	NA	NA	MULLAGH
5	NA	NA	NA	20	BREDA COURT	NA	NA	NA	BREDA
6	NA	NA	NA	11	UPPER HEATHMOUNT	NA	NA	NA	MULLAGH
7	NA	NA	NA	86	LEVER ROAD	NA	NA	NA	MULLAGH
8	NA	NA	NA	112	OLD COACH ROAD	NA	NA	NA	MULLAGH
9	NA	NA	NA	134	WHITEPARK ROAD	NA	NA	BALLINTOY	BALLINTOY
10	NA	FLAT 3	NA	16	STATION ROAD	NA	NA	NA	MULLAGH
11	NA	NA	NA	15	HEATHERLEA AVENUE	NA	NA	NA	MULLAGH
12	NA	NA	NA	30	STATION ROAD	NA	NA	NA	MULLAGH
13	NA	NA	NA	21	HIGH ROAD	NA	NA	NA	MULLAGH

Fig.7: Dataset replaced with NA

D) As I have replaced with NA the count of missing entries and after replacing with NA has same number of count. The function used in the code is used to iterate each column of the dataset and check the count of the NA values in each attribute.

```
# Checking the count of NA in each attribute
data.frame(sapply(NIPOSTCODE, function(x)sum(length(which(is.na(x))))))

```

Fig.10: Checking the count of NA

Output:

```
> data.frame(sapply(NIPOSTCODE, function(x)sum(length(which(is.na(x))))))
                                    sapply.NIPOSTCODE..function.x..sum.length.which.is.na.x.....
Organisation Name                                                                        890537
Sub-building Name                                                                        884099
Building Name                                                                            895540
Number                                                                                    28753
Primary Thorfare                                                                            470
Alt Thorfare                                                                             921788
Secondary Thorfare                                                                       938400
Locality                                                                                 856789
Townland                                                                                      0
Town                                                                                      19872
County                                                                                        0
Postcode                                                                                   8900
x-coordinates                                                                                 0
y-coordinates                                                                                 0
Primary Key (identifier)                                                                      0
```

Fig.11: structure of NIpostcode

E) In this task we need to shift the primary key identifier to the first of the dataset. We can see in the below figure the primary key identifier is at the last.

Filter

Alt Thorfare	Secondary Thorfare	Locality	Townland	Town	County	Postcode	x-coordinates	y-coordinates	Primary Key (identifier)
			MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557BG	281855	438598	1
			MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557BW	281892	438228	2
			MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557HH	282306	438587	3
			MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557HW	282419	438387	4
			BREDA	BELFAST	DOWN	BT86JB	335367	369985	5
			MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557AR	281719	438366	6
			MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557EE	282080	438424	7
			MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557HW	282524	438243	8
		BALLINTOY	BALLINTOY DEMESNE	BALLYCASTLE	ANTRIM	BT546ND	303527	444150	9
			MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557DA	282128	438612	10

Fig.12: Data frame before shifting the primary key

Now we need to move the primary key identifier to the first place. In the code I have changed the column values by shifting the last value to the first. As we can see in the below code the number 15 is shifted to the first place. In output we can see that primary key identifier is shifted to the first place.

```
# Changing the position of the primary key to the first
NIPOSTCODE <-NIPOSTCODE[, c(15,1,2,3,4,5,6,7,8,9,10,11,12,13,14)]
# Displaying the first 10 rows
head(NIPOSTCODE,10)
```

Fig.13: changing the position of primary key identifier

Output:

Functions.R × | CA2.R* × | NIPOSTCODE × | section2(CA2).R ×

Filter

	Primary Key (identifier)	Organisation Name	Sub-building Name	Building Name	Number	Primary Thorfare	Alt Thorfare	Secondary Thorfare	Locality	Townland
1	1	NA	NA	NA	17	HIGH ROAD	NA	NA	NA	MULLAC
2	2	NA	NA	NA	15	CONVENTION AVENUE	NA	NA	NA	MULLAC
3	3	NA	NA	NA	13	STATION ROAD	NA	NA	NA	MULLAC
4	4	NA	NA	NA	99	OLD COACH ROAD	NA	NA	NA	MULLAC
5	5	NA	NA	NA	20	BREDA COURT	NA	NA	NA	BREDA
6	6	NA	NA	NA	11	UPPER HEATHMOUNT	NA	NA	NA	MULLAC
7	7	NA	NA	NA	86	LEVER ROAD	NA	NA	NA	MULLAC
8	8	NA	NA	NA	112	OLD COACH ROAD	NA	NA	NA	MULLAC
9	9	NA	NA	NA	134	WHITEPARK ROAD	NA	NA	BALLINTOY	BALLINT
10	10	NA	FLAT 3	NA	16	STATION ROAD	NA	NA	NA	MULLAC

Fig.14:

F) In this task we need to create a new dataset called Limavady_data and store information only within it where locality, townland, town contain the name limavady. Count, display the rows and store the information in the csv file called limavady.

Functions.R × | CA2.R* × | limavady_data × | NIPOSTCODE × | section2(CA2).R ×

Filter

imary Thorfare	Alt Thorfare	Secondary Thorfare	Locality	Townland	Town	County	Postcode	x-coordinates	y-coordinates
GH ROAD	NA	NA	NA	MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557BG	281855	438598
ONVENTION AVENUE	NA	NA	NA	MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557BW	281892	438228
ATION ROAD	NA	NA	NA	MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557HH	282306	438587
D COACH ROAD	NA	NA	NA	MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557HW	282419	438387
EDA COURT	NA	NA	NA	BREDA	BELFAST	DOWN	BT86JB	335367	369985
PER HEATHMOUNT	NA	NA	NA	MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557AR	281719	438366
VER ROAD	NA	NA	NA	MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557EE	282080	438424
D COACH ROAD	NA	NA	NA	MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557HW	282524	438243
HITEPARK ROAD	NA	NA	BALLINTOY	BALLINTOY DEMESNE	BALLYCASTLE	ANTRIM	BT546ND	303527	444150
ATION ROAD	NA	NA	NA	MULLAGHACALL NORTH	PORTSTEWART	LONDONDERRY	BT557DA	282128	438612

Fig.15: Dataset before applying the changes

Code: In this task we have used subset function to check the columns within the NIPostcode dataset which contains the name limavady. In the code subset function uses the NI postcode data and checks the columns locality, townland, town and store only the information containing the name limavady. After that check the number of rows and store the information in the csv file and name it as limavady.

```
# To store only the data within name containing limavady
attach(NIPOSTCODE)
limavady_data <- subset(NIPOSTCODE, Locality=="LIMAVADY"|Townland == "LIMAVADY"|Town=="LIMAVADY")
View(limavady_data)
# To check the rows in the Limavady data
nrow(limavady_data)
# The modified data is loaded in to a csv file
write.csv(limavady_data, "Limavady.csv")
```

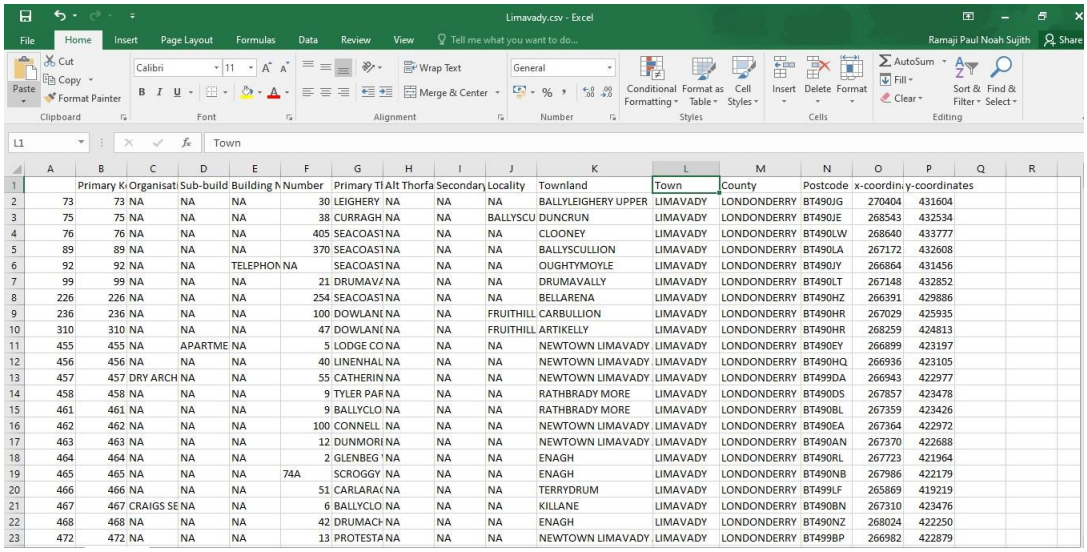
Fig.16: creating a data frame

Output:

As we can see in the output the name containing limavady is stored in the assigned columns in the dataset limavady. Number of rows are displayed in the below figure.17 output

```
> # To check the rows in the Limavady data
> nrow(limavady_data)
[1] 8467
>
```

Fig.17: Number of rows of limavady data



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1		Primary K	Organisati	Sub-build	Building N	Number	Primary Tl	Alt Thorfa	Secondary	Locality	Townland	Town	County	Postcode	x-coordina	y-coordinates
2	73	73	NA	NA	NA	30	LEIGHERY	NA	NA	NA	BALLYLEIGHERY UPPER	LIMAVADY	LONDONDERRY	BT490JG	270404	431604
3	75	75	NA	NA	NA	38	CURRAGH	NA	NA	BALLYSCU	DUNCRUN	LIMAVADY	LONDONDERRY	BT490JE	268543	432534
4	76	76	NA	NA	NA	405	SEACOAST	NA	NA	NA	CLOONEY	LIMAVADY	LONDONDERRY	BT490LW	268640	433777
5	89	89	NA	NA	NA	370	SEACOAST	NA	NA	NA	BALLYSCULLION	LIMAVADY	LONDONDERRY	BT490LA	267172	432608
6	92	92	NA	NA	TELEPHON	NA	SEACOAST	NA	NA	NA	OUGHTYMOYLE	LIMAVADY	LONDONDERRY	BT490JY	266864	431456
7	99	99	NA	NA	NA	21	DRUMAVA	NA	NA	NA	DRUMAVALLY	LIMAVADY	LONDONDERRY	BT490LT	267148	432852
8	226	226	NA	NA	NA	254	SEACOAST	NA	NA	NA	BELLARENA	LIMAVADY	LONDONDERRY	BT490HZ	266391	429886
9	236	236	NA	NA	NA	100	DOWLANI	NA	NA	FRUITHILL	CARBULLION	LIMAVADY	LONDONDERRY	BT490HR	267029	425935
10	310	310	NA	NA	NA	47	DOWLANI	NA	NA	FRUITHILL	ARTIKELLY	LIMAVADY	LONDONDERRY	BT490HR	268259	424813
11	455	455	NA	APARTME	NA	5	LODGE CO	NA	NA	NA	NEWTOWN LIMAVADY	LIMAVADY	LONDONDERRY	BT490EY	266899	423197
12	456	456	NA	NA	NA	40	LINENHAL	NA	NA	NA	NEWTOWN LIMAVADY	LIMAVADY	LONDONDERRY	BT490HQ	266936	423105
13	457	457	DRY ARCH	NA	NA	55	CATHERIN	NA	NA	NA	NEWTOWN LIMAVADY	LIMAVADY	LONDONDERRY	BT499DA	266943	422977
14	458	458	NA	NA	NA	9	TYLER PAR	NA	NA	NA	RATHBRADY MORE	LIMAVADY	LONDONDERRY	BT490DS	267857	423478
15	461	461	NA	NA	NA	9	BALLYCLO	NA	NA	NA	RATHBRADY MORE	LIMAVADY	LONDONDERRY	BT490BL	267359	423426
16	462	462	NA	NA	NA	100	CONNELL	NA	NA	NA	NEWTOWN LIMAVADY	LIMAVADY	LONDONDERRY	BT490EA	267364	422972
17	463	463	NA	NA	NA	12	DUNMORI	NA	NA	NA	NEWTOWN LIMAVADY	LIMAVADY	LONDONDERRY	BT490AN	267370	422688
18	464	464	NA	NA	NA	2	GLENBEG	NA	NA	NA	ENAGH	LIMAVADY	LONDONDERRY	BT490RL	267723	421964
19	465	465	NA	NA	NA	74A	SCROGGY	NA	NA	NA	ENAGH	LIMAVADY	LONDONDERRY	BT490NB	267986	422179
20	466	466	NA	NA	NA	51	CARLARA	NA	NA	NA	TERRYDRUM	LIMAVADY	LONDONDERRY	BT499LF	265869	419219
21	467	467	CRAIGS SE	NA	NA	6	BALLYCLO	NA	NA	NA	KILLANE	LIMAVADY	LONDONDERRY	BT490BN	267310	423476
22	468	468	NA	NA	NA	42	DRUMACH	NA	NA	NA	ENAGH	LIMAVADY	LONDONDERRY	BT490NZ	268024	422250
23	472	472	NA	NA	NA	13	PROTESTA	NA	NA	NA	NEWTOWN LIMAVADY	LIMAVADY	LONDONDERRY	BT499BP	266982	422879

Fig.18: Limavady.csv file

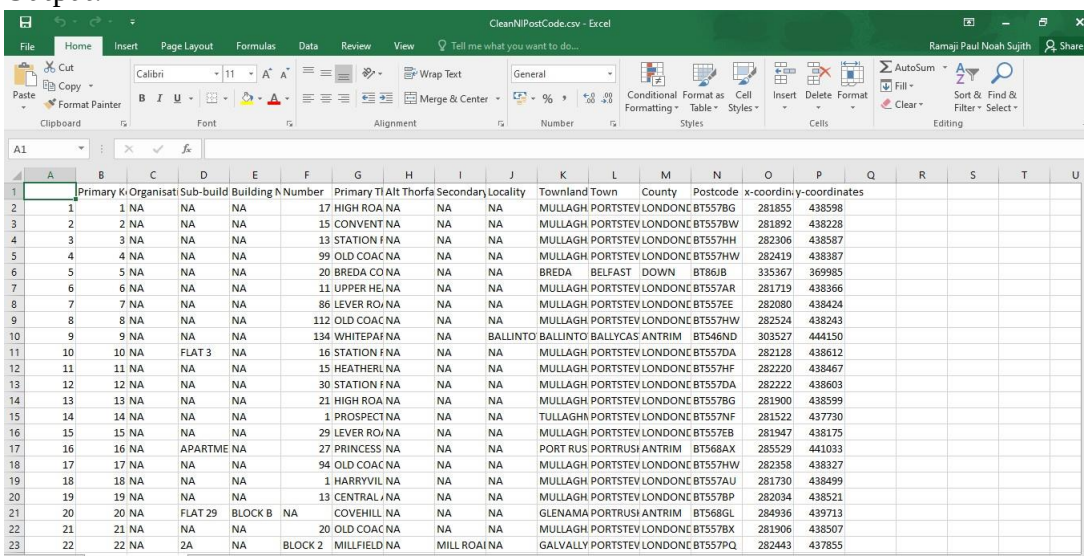
Now check the rows of the limavady data and

G) Now save the modified NIPostCode data into a csv file and name it as CleanNIPostCodeData. Write.csv file saves the data into a csv file.

```
# Saving the modified data in new csv file
write.csv(NIPOSTCODE, "CleanNIPostCode.csv")
```

Fig.19: writing the NIpostcodedata

Output:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1		Primary K	Organisati	Sub-build	Building N	Number	Primary Tl	Alt Thorfa	Secondary	Locality	Townland	Town	County	Postcode	x-coordina	y-coordinates
2	1	1	NA	NA	NA	17	HIGH ROA	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557BG	281855	438598
3	2	2	NA	NA	NA	15	CONVENT	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557BW	281892	438228
4	3	3	NA	NA	NA	13	STATION F	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557HH	282306	438587
5	4	4	NA	NA	NA	99	OLD COAC	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557HW	282419	438387
6	5	5	NA	NA	NA	20	BREDA CO	NA	NA	NA	BREDA	BELFAST	DOWN	BT86JB	335367	369985
7	6	6	NA	NA	NA	11	UPPER HE	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557AR	281719	438366
8	7	7	NA	NA	NA	86	LEVER RO	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557EE	282080	438424
9	8	8	NA	NA	NA	112	OLD COAC	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557HW	282524	438243
10	9	9	NA	NA	NA	134	WHITEPAF	NA	NA	BALLINTO	BALLINTO	BALLYCAS	ANTRIM	BT546ND	303527	444150
11	10	10	NA	FLAT 3	NA	16	STATION F	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557DA	282128	438612
12	11	11	NA	NA	NA	15	HEATHERL	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557HF	282220	438467
13	12	12	NA	NA	NA	30	STATION F	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557DA	282222	438603
14	13	13	NA	NA	NA	21	HIGH ROA	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557BG	281900	438599
15	14	14	NA	NA	NA	1	PROSPECT	NA	NA	NA	TULLAGHN	PORTSTEV	LONDONI	BT557NF	281522	437730
16	15	15	NA	NA	NA	29	LEVER RO	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557EB	281947	438175
17	16	16	NA	APARTME	NA	27	PRINCESS	NA	NA	NA	PORT RUS	PORTRUSH	ANTRIM	BT568AX	285529	441033
18	17	17	NA	NA	NA	94	OLD COAC	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557HW	282358	438327
19	18	18	NA	NA	NA	1	HARRYVIL	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557AU	281730	438499
20	19	19	NA	NA	NA	13	CENTRAL	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557BP	282034	438521
21	20	20	NA	FLAT 29	BLOCK B	NA	COVEHILL	NA	NA	NA	GLENAMA	PORTRUSH	ANTRIM	BT568GL	284936	439713
22	21	21	NA	NA	NA	20	OLD COAC	NA	NA	NA	MULLAGH	PORTSTEV	LONDONI	BT557BX	281906	438507
23	22	22	NA	2A	NA	BLOCK 2	MILLFIELD	NA	MILL ROA	NA	GALVALLY	PORTSTEV	LONDONI	BT557PQ	282443	437855

Fig.20: CleanNIPostCode.csv file

Section:2

To complete this section, they have given a dataset which contains all the crime data of Northern Ireland from 2015 to 2016 in different folders as shown in below figure

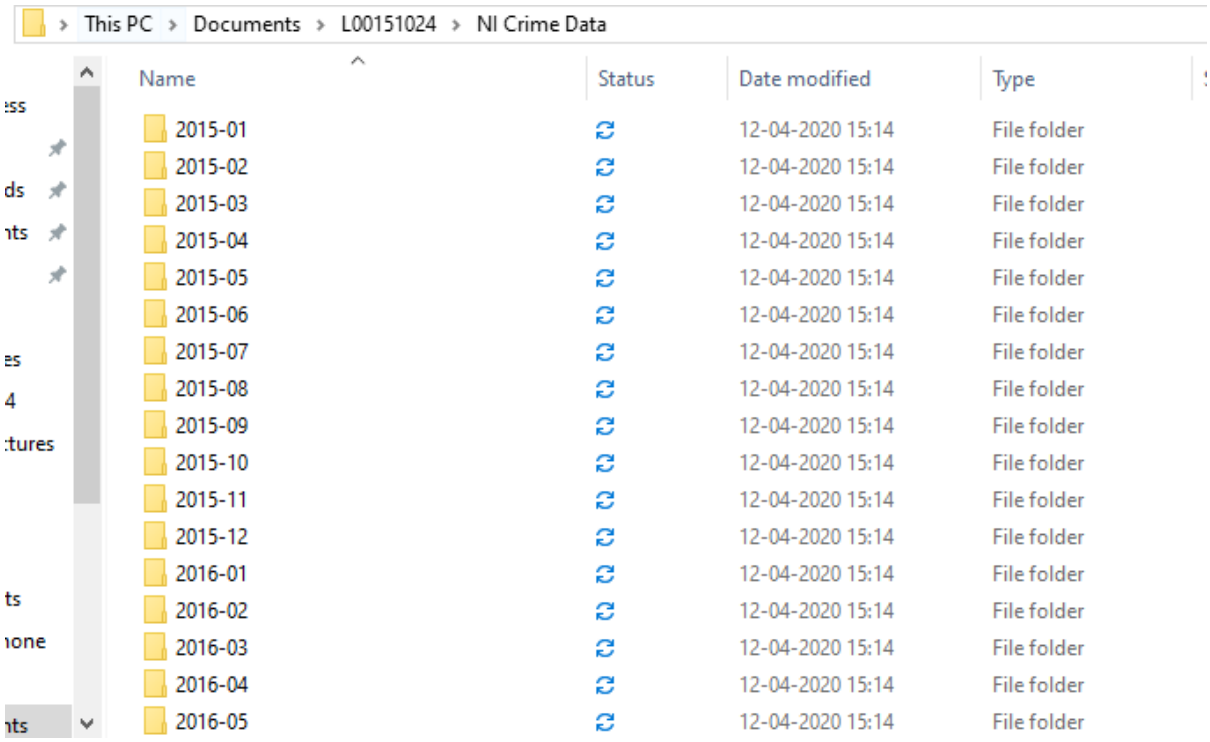


Fig.1: Checking the dataset folders

A) In this task we need to combine all the files in crime data from each csv file into one dataset and save this dataset into a csv file called AllNICrimeData. Show the count of number of rows in the AllNICrimeData.

Code: In this task we have used list function is used to extract the files within the folders in the NICrimeData by using the for loop. In this task cannot hardcode the data as it is a bad option to combine the files. Rbind is used to combine the data frames by rows. Finally saving the combined dataset into a csv file called AllNICrimeData.csv.

```r
# A)
# combining all of the crime data from each csv file into one data set
# creating an empty dataframe to store the data after looping
final_crime_data <- data.frame()
#files_list contains all the csv files from NI Crime Data
files_list <- list.files(recursive = TRUE)
for(file_list in files_list){
  crime_data <- read.csv(file_list)
  final_crime_data <- rbind(final_crime_data, crime_data)
}
# Getting the current working directory
getwd()
setwd("NI Crime Data/")
# Saving the combined dataset into a csv file
write.csv(final_crime_data, "AllNICrimeData.csv")
AllNICrimeData <- read.csv("AllNICrimeData.csv")
str(AllNICrimeData)
# showing the number of rows
nrow(AllNICrimeData)
```

Fig.1: code for task A

Output:

In the output we can see that all the files from the NI crime folders have been combined and count of rows is shown in the below figure.

```
> # Saving the combined dataset into a csv file
> write.csv(final_crime_data, "AllNICrimeData.csv")
> AllNICrimeData <- read.csv("AllNICrimeData.csv")
> str(AllNICrimeData)
'data.frame':   477696 obs. of  13 variables:
 $ X                 : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Crime.ID          : Factor w/ 11667 levels "","0009d3218c478283888080303fed14c46c61e6b3b8f55963a2671dac3afb3907",..: 1 1 1
 1 1 1 1 1 1 1 ...
 $ Month             : Factor w/ 36 levels "2015-01","2015-02",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Reported.by       : Factor w/ 1 level "Police Service of Northern Ireland": 1 1 1 1 1 1 1 1 1 1 ...
 $ Falls.within      : Factor w/ 1 level "Police Service of Northern Ireland": 1 1 1 1 1 1 1 1 1 1 ...
 $ Longitude         : num  -6 -5.71 -5.82 -6.39 -6.25 ...
 $ Latitude          : num  54.6 54.6 54.7 54.2 54.9 ...
 $ Location          : Factor w/ 14984 levels "No Location",..: 12359 2 9993 3624 13143 8121 11826 2 6913 7811 ...
 $ LSOA.code         : logi  NA NA NA NA NA NA ...
 $ LSOA.name         : logi  NA NA NA NA NA NA ...
 $ Crime.type        : Factor w/ 14 levels "Anti-social behaviour",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Last.outcome.category: logi  NA NA NA NA NA NA ...
 $ Context           : logi  NA NA NA NA NA NA ...
> nrow(AllNICrimeData)
[1] 477696
```

Fig.2: output for the structure and count of the AllNICrimeData

Functions.R | CA2.R* | limavady_data | NIPOSTCODE | section2(CA2).R* | AllNICrimeData | final_crime_data | random_sample_data

Filter

	X	Crime.ID	Month	Reported.by	Falls.within	Longitude	Latitude	Location	LSOA.code	LSOA.name
1	1		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-6.003289	54.55165	On or near Salisbury Place	NA	NA
2	2		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-5.707979	54.59231	On or near	NA	NA
3	3		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-5.815976	54.73161	On or near Milebush Park	NA	NA
4	4		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-6.393411	54.19788	On or near College Square North	NA	NA
5	5		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-6.251798	54.85970	On or near Staffa Drive	NA	NA
6	6		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-7.206893	54.62265	On or near Killyclogher Road	NA	NA
7	7		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-5.915793	54.59242	On or near Ravenhill Reach	NA	NA
8	8		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-5.535389	54.48792	On or near	NA	NA
9	9		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-7.322812	54.99940	On or near Great James Street	NA	NA
10	10		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-5.954670	54.61568	On or near Jamaica Road	NA	NA
11	11		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-6.707998	55.18703	On or near Station Road	NA	NA
12	12		2015-01	Police Service of Northern Ireland	Police Service of Northern Ireland	-6.049408	54.51354	On or near Antrim Street	NA	NA

Showing 1 to 13 of 477,696 entries, 13 total columns

Fig.3: output for the combined files placed in the AllNICrimeData file

B) In this task we need to modify newly created AllNICrimeData by removing the following attributed like Crime ID, Reported by, Falls within, LSOA code, LSOA name, last outcome and context.

Functions.R × | CA2.R* × | NIPOSTCODE × | section2(CA2).R* × | AllNICrimeData × | final_crime_data × | random_sample_data ×

Filter

Reported.by	Falls.within	Longitude	Latitude	Location	LSOA.code	LSOA.name	Crime.type	Last.o
Police Service of Northern Ireland	Police Service of Northern Ireland	-6.003289	54.55165	On or near Salisbury Place	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-5.707979	54.59231	On or near	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-5.815976	54.73161	On or near Milebush Park	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-6.393411	54.19788	On or near College Square North	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-6.251798	54.85970	On or near Staffa Drive	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-7.206893	54.62265	On or near Killyclogher Road	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-5.915793	54.59242	On or near Ravenhill Reach	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-5.535389	54.48792	On or near	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-7.322812	54.99940	On or near Great James Street	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-5.954670	54.61568	On or near Jamaica Road	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-6.707998	55.18703	On or near Station Road	NA	NA	Anti-social behaviour	NA
Police Service of Northern Ireland	Police Service of Northern Ireland	-6.049408	54.51354	On or near Antrim Street	NA	NA	Anti-social behaviour	NA

Fig.4: Before removing the attributes

Code: I have selected the required attributes and stored into the AllNICrimeData instead of removing the above given attributes.

```
# B)
# Selecting required attribute and removing other attributes according to the given task
AllNICrimeData <- final_crime_data[c(2,5,6,7,10)]
str(AllNICrimeData)
```

Fig.5:

Output:

Functions.R × | CA2.R* × | NIPOSTCODE × | section2(CA2).R* × | AllNICrimeData × | final_

Filter

	Month	Longitude	Latitude	Location	Crime.type
1	2015-01	-6.003289	54.55165	On or near Salisbury Place	Anti-social behaviour
2	2015-01	-5.707979	54.59231	On or near	Anti-social behaviour
3	2015-01	-5.815976	54.73161	On or near Milebush Park	Anti-social behaviour
4	2015-01	-6.393411	54.19788	On or near College Square North	Anti-social behaviour
5	2015-01	-6.251798	54.85970	On or near Staffa Drive	Anti-social behaviour
6	2015-01	-7.206893	54.62265	On or near Killyclogher Road	Anti-social behaviour
7	2015-01	-5.915793	54.59242	On or near Ravenhill Reach	Anti-social behaviour
8	2015-01	-5.535389	54.48792	On or near	Anti-social behaviour
9	2015-01	-7.322812	54.99940	On or near Great James Street	Anti-social behaviour
10	2015-01	-5.954670	54.61568	On or near Jamaica Road	Anti-social behaviour
11	2015-01	-6.707998	55.18703	On or near Station Road	Anti-social behaviour
12	2015-01	-6.049408	54.51354	On or near Antrim Street	Anti-social behaviour
13	2015-01	-5.709197	54.59706	On or near Westland Drive	Anti-social behaviour

Showing 1 to 14 of 477,696 entries, 5 total columns

Fig.6: Output after removing certain attributes

```
> str(AllNICrimeData)
'data.frame':   477696 obs. of  5 variables:
 $ Month     : Factor w/ 36 levels "2015-01","2015-02",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Longitude : num  -6 -5.71 -5.82 -6.39 -6.25 ...
 $ Latitude  : num  54.6 54.6 54.7 54.2 54.9 ...
 $ Location  : Factor w/ 14984 levels "No Location",..: 3507 2 2790 1022 3724 2263 3363 2 1909 2179 ...
 $ Crime.type: Factor w/ 14 levels "Anti-social behaviour",..: 1 1 1 1 1 1 1 1 1 1 ...
```

Fig.7: Structure of the modified file

C) In this task we need to change the crime type in AllNICrimeData by giving them short forms like for anti-social behavior be ASBO, Bicycle theft as BITH and so on as shown in the below code.

Functions.R × | CA2.R* × | NIPOSTCODE × | section2(CA2).R* × | AllNICrimeData × | final_cr

Filter

	Month	Longitude	Latitude	Location	Crime.type
1	2015-01	-6.003289	54.55165	On or near Salisbury Place	Anti-social behaviour
2	2015-01	-5.707979	54.59231	On or near	Anti-social behaviour
3	2015-01	-5.815976	54.73161	On or near Milebush Park	Anti-social behaviour
4	2015-01	-6.393411	54.19788	On or near College Square North	Anti-social behaviour
5	2015-01	-6.251798	54.85970	On or near Staffa Drive	Anti-social behaviour
6	2015-01	-7.206893	54.62265	On or near Killyclogher Road	Anti-social behaviour
7	2015-01	-5.915793	54.59242	On or near Ravenhill Reach	Anti-social behaviour
8	2015-01	-5.535389	54.48792	On or near	Anti-social behaviour
9	2015-01	-7.322812	54.99940	On or near Great James Street	Anti-social behaviour
10	2015-01	-5.954670	54.61568	On or near Jamaica Road	Anti-social behaviour
11	2015-01	-6.707998	55.18703	On or near Station Road	Anti-social behaviour
12	2015-01	-6.049408	54.51354	On or near Antrim Street	Anti-social behaviour
13	2015-01	-5.709197	54.59706	On or near Westland Drive	Anti-social behaviour

Fig.8: Data frame before shortening the crime type

Code: In the below code I have used function as.character to change the character and assign a short forms to the attributes.

```
# C)
# Giving short forms to the crime type attribute
AllNICrimeData$Crime.type <- as.character(AllNICrimeData$Crime.type)
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Anti-social behaviour"] <- "ASBO"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Bicycle theft"] <- "BITH"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Burglary"] <- "BURG"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Criminal damage and arson"] <- "CDAR"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "DRUGS"] <- "DRUG"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Other Theft"] <- "OTTH"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Public order"] <- "PUBO"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Robbery"] <- "ROBY"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Shoplifting"] <- "SHOP"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Theft from the person"] <- "THPR"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Vehicle crime"] <- "VECR"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Violence and sexual offences"] <- "VISO"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Other crime"] <- "OTCR"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Drugs"] <- "DRUG"
AllNICrimeData$Crime.type[AllNICrimeData$Crime.type == "Possession of weapons"] <- "POW"
```

Fig.8: code to shorten the crime type

Output:

Functions.R × | CA2.R* × | NIPOSTCODE × | section2(CA2).R* × | AllNICrimeData ×

Filter

	Month	Longitude	Latitude	Location	Crime.type
1	2015-01	-6.003289	54.55165	On or near Salisbury Place	ASBO
2	2015-01	-5.707979	54.59231	On or near	ASBO
3	2015-01	-5.815976	54.73161	On or near Milebush Park	ASBO
4	2015-01	-6.393411	54.19788	On or near College Square North	ASBO
5	2015-01	-6.251798	54.85970	On or near Staffa Drive	ASBO
6	2015-01	-7.206893	54.62265	On or near Killyclogher Road	ASBO
7	2015-01	-5.915793	54.59242	On or near Ravenhill Reach	ASBO
8	2015-01	-5.535389	54.48792	On or near	ASBO
9	2015-01	-7.322812	54.99940	On or near Great James Street	ASBO
10	2015-01	-5.954670	54.61568	On or near Jamaica Road	ASBO
11	2015-01	-6.707998	55.18703	On or near Station Road	ASBO
12	2015-01	-6.049408	54.51354	On or near Antrim Street	ASBO
13	2015-01	-5.709197	54.59706	On or near Westland Drive	ASBO

Showing 1 to 14 of 477,696 entries, 5 total columns

Fig.9: Output after shortening the crime type

D) In this task we need to show a bar plot of the frequency of each crime from the crime type field and specify relevant options such as labels and bar colors.

Code: In the below I have used the library called ggplot2 which is a system for declaratively creating graphics. I have specified the X label as crime type and colors blue and black.

```
# D)
# Plotting a bar graph of frequecy of crime type and setting the axe labels and bar colours
library(ggplot2)
ggplot(AllNICrimeData, aes(x=Crime.type)) + geom_bar(fill = "blue", color ="black")
```

Fig.10:

Output: By observing the below bar plot we can say that anti-social behavior (ASBO) has the highest frequency of crimes comparing with the other crimes. While Theft from a person(THPR) and Robbery(ROBY) record lowest frequency in the crime rate. Violence and sexual offence(VISO) is the second highest in the crime type.

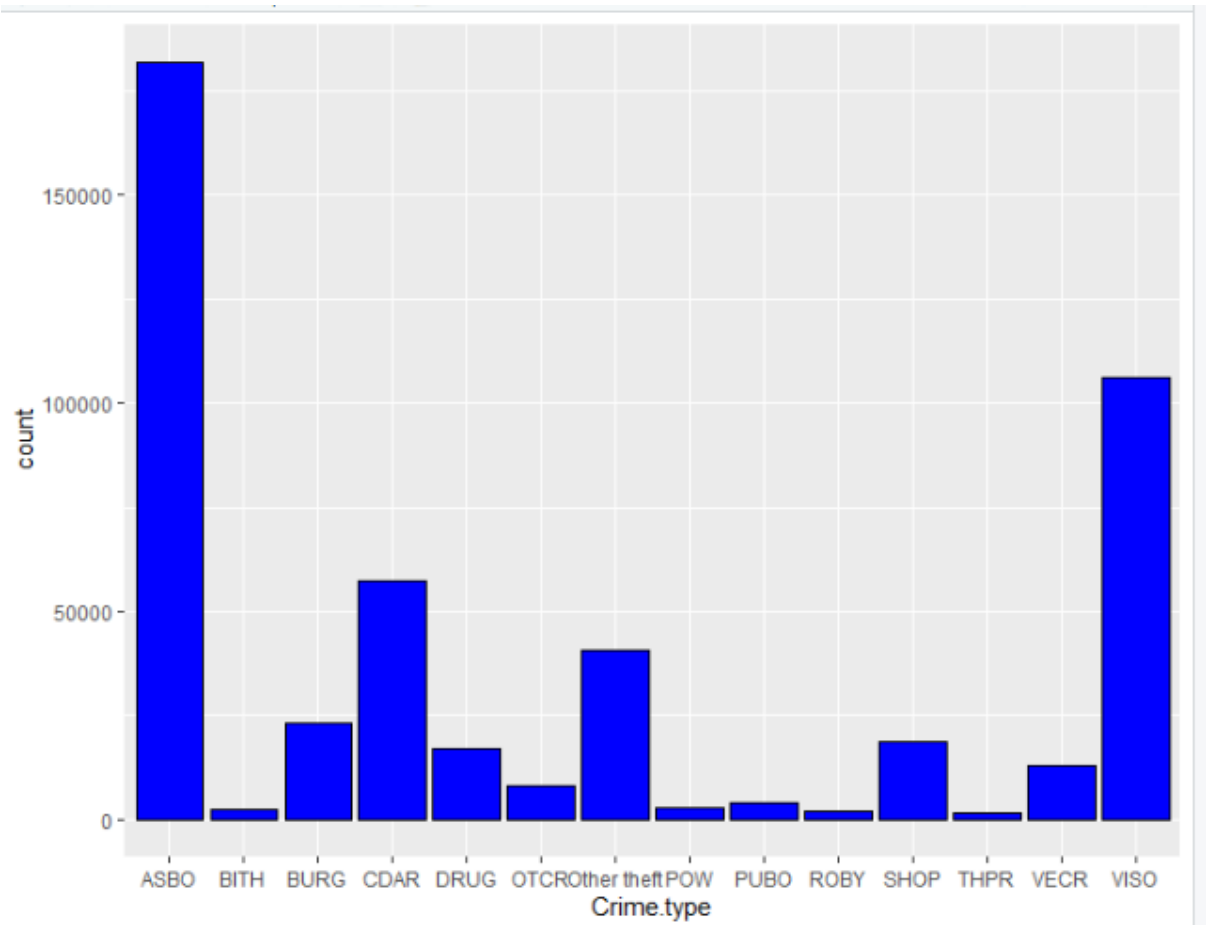


Fig.11: Bar plot for the crime data

E) In this task we need to modify the AllNICrimeData such that the location attribute should contain only street name. we have observed that the attribute values are like on or near west rock square we need to remove the on or near and only assign street name to the attribute. And some attribute values are like no location.

Functions.R × | CA2.R* × | NIPOSTCODE × | section2(CA2).R* × | AllNICrimeData ×

Filter

	Month	Longitude	Latitude	Location	Crime.type
1	2015-01	-6.003289	54.55165	On or near Salisbury Place	ASBO
2	2015-01	-5.707979	54.59231	On or near	ASBO
3	2015-01	-5.815976	54.73161	On or near Milebush Park	ASBO
4	2015-01	-6.393411	54.19788	On or near College Square North	ASBO
5	2015-01	-6.251798	54.85970	On or near Staffa Drive	ASBO
6	2015-01	-7.206893	54.62265	On or near Killyclogher Road	ASBO
7	2015-01	-5.915793	54.59242	On or near Ravenhill Reach	ASBO
8	2015-01	-5.535389	54.48792	On or near	ASBO
9	2015-01	-7.322812	54.99940	On or near Great James Street	ASBO
10	2015-01	-5.954670	54.61568	On or near Jamaica Road	ASBO
11	2015-01	-6.707998	55.18703	On or near Station Road	ASBO
12	2015-01	-6.049408	54.51354	On or near Antrim Street	ASBO
13	2015-01	-5.709197	54.59706	On or near Westland Drive	ASBO

Showing 1 to 14 of 477,696 entries, 5 total columns

Fig.12: Data frame before changing the data

Code: we have used str_remove_all function by importing the library stringr and to remove the on or near in the location attribute and saved AllNICrimeData. For attribute value with no location we have assigned NA values as in the upcoming tasks we need to remove the NA values.

```
# E)
#install.packages("stringr")
library(stringr)
# Removing the extra strings in the location attribute and only keeping the street names
AllNICrimeData$Location <- str_remove_all(AllNICrimeData$Location, 'On or near ')
# Assigning NA values for no location attribute value
AllNICrimeData$Location[AllNICrimeData$Location == "no location" | AllNICrimeData$Location == ""] <- NA
```

Fig.13: Removing the NA values

Output:

Functions.R × | CA2.R* × | NIPOSTCODE × | section2(CA2).R* × | AllNICrimeData × | final_crime_

Filter

	Month	Longitude	Latitude	Location	Crime.type
1	2015-01	-6.003289	54.55165	Salisbury Place	ASBO
2	2015-01	-5.707979	54.59231	NA	ASBO
3	2015-01	-5.815976	54.73161	Milebush Park	ASBO
4	2015-01	-6.393411	54.19788	College Square North	ASBO
5	2015-01	-6.251798	54.85970	Staffa Drive	ASBO
6	2015-01	-7.206893	54.62265	Killyclogher Road	ASBO
7	2015-01	-5.915793	54.59242	Ravenhill Reach	ASBO
8	2015-01	-5.535389	54.48792	NA	ASBO
9	2015-01	-7.322812	54.99940	Great James Street	ASBO
10	2015-01	-5.954670	54.61568	Jamaica Road	ASBO
11	2015-01	-6.707998	55.18703	Station Road	ASBO
12	2015-01	-6.049408	54.51354	Antrim Street	ASBO
13	2015-01	-5.709197	54.59706	Westland Drive	ASBO

Showing 1 to 14 of 477,696 entries, 5 total columns

Fig.14: Output after removing unwanted strings and assigning NA for no location

F) This task assigns us to choose 5000 random samples from the crime data of AllNICrimeData dataset where location attribute should contain location name and there should be no NA values. We need to set seed value to 100 and create a new data frame called random_crime_sample. Now create a function called find _a_town which uses CleanNIPostCodeData dataset to find the correct town information for each location variable within the random_crime_sample dataset. Save each matched town into the random_crime_sample.

Code: In the first line of the code we have set the seed value to 100 to generate random number. After that by using subset function we have extracted the 5000 random samples and used !is.na function to ignore the NA values and saved in the variable random. Now save the extracted data into new dataset and name it as random_crime_sample.

```
# F)
# Choosing 5000 random samples of crime data from AllNICrimeData
# setting the seed value to 100
set.seed(100)
# Where location attribute should not contain any NA values
random <- subset(AllNICrimeData[, ], !is.na(AllNICrimeData$Location))
# saving the data into the new file randowm_crime_data
random_crime_sample <- random[sample(1:nrow(random), 5000, replace = FALSE), ]
```

Fig.15: code to extract 5000 random samples

Output:

Functions.R × | CA2.R* × | NIPOSTCODE × | section2(CA2).R* × | random_crime_s

Filter

	Month	Longitude	Latitude	Location	Crime.type
95810	2015-08	-5.933066	54.59109	Dublin Road	ASBO
384531	2017-06	-7.476470	54.78748	Daisy Park	ASBO
189080	2016-03	-5.935054	54.59857	King Street	ASBO
104404	2015-08	-5.963882	54.63147	Tyndale Gardens	VECR
241344	2016-07	-6.223332	54.48017	Meeting Street	ASBO
410624	2017-07	-5.839762	54.63121	Ardnagreena Gardens	VISO
103416	2015-08	-5.896897	54.21220	Shimna Mile	PUBO
206367	2016-04	-6.431276	54.43006	Grantham Park	CDAR
133178	2015-10	-5.930067	54.60018	Rosemary Street	SHOP
241430	2016-07	-6.739900	54.63720	Stewart Avenue	ASBO
219424	2016-05	-7.281825	54.60935	Knockshee Park	CDAR
97330	2015-08	-6.278889	54.86499	George Street	ASBO
146854	2015-11	-6.457149	54.43173	Ballyoran Heights	VISO
300797	2016-11	-6.338295	54.45736	Sloan Street	BURG
448064	2017-10	-5.970893	54.62690	Silverstream Parade	Other theft
314169	2016-12	-6.281757	54.85361	Montague Park	DRUG
265748	2016-08	-6.275444	54.87672	Kew Gardens	VISO
447649	2017-10	-6.213340	54.59704	Gobrana Road	DRUG
327654	2017-01	-6.068101	54.52843	The Oaks	Other theft
395384	2017-06	-5.948576	54.68843	Beverley Road	VISO

Showing 1 to 21 of 5,000 entries, 5 total columns

Fig.16: Output of 5000 random samples

Now change the string to lower case by using str_to_lower of both the data sets NIPOSTCODE and random_crime_sample in the location and primary thorfare attribute respectively to match the attribute values.

```
# changing the string to lower case to match the attribute
NIPOSTCODE$`Primary Thorfare` <- str_to_lower(NIPOSTCODE$`Primary Thorfare`)
random_crime_sample$Location <- str_to_lower(random_crime_sample$Location)
```

Fig.17: code for changing to lower case

After changing to lower now create a function find_a_town and return the match function to compare the location of two datasets. Store it into the dataset random_crime_sample and check the structure of the data frame

```
# creating a function to match to find correct town/city information for each location variable within th
find_a_town <- function(x,y){

  return(NIPOSTCODE$Town[match(x,y)])
}
# Saving the match attributes into the town column of the random_crime_sample
random_crime_sample$Town <- find_a_town(random_crime_sample$Location, NIPOSTCODE$`Primary Thorfare`)
str(random_crime_sample)
```

Fig.18: creating a function

Output:

Functions.R | CA2.R* | NIPOSTCODE | limavady_data | section2(CA2).R* | random_crime_sample

Filter

	Month	Longitude	Latitude	Location	Crime.type	Town
95810	2015-08	-5.933066	54.59109	dublin road	ASBO	OMAGH
384531	2017-06	-7.476470	54.78748	daisy park	ASBO	SION MILLS
189080	2016-03	-5.935054	54.59857	king street	ASBO	BELFAST
104404	2015-08	-5.963882	54.63147	tyndale gardens	VECR	BELFAST
241344	2016-07	-6.223332	54.48017	meeting street	ASBO	DROMORE
410624	2017-07	-5.839762	54.63121	ardnagreena gardens	VISO	HOLYWOOD
103416	2015-08	-5.896897	54.21220	shimna mile	PUBO	NEWCASTLE
206367	2016-04	-6.431276	54.43006	grantham park	CDAR	PORTADOWN
133178	2015-10	-5.930067	54.60018	rosemary street	SHOP	BELFAST
241430	2016-07	-6.739900	54.63720	stewart avenue	ASBO	COOKSTOWN
219424	2016-05	-7.281825	54.60935	knockshee park	CDAR	OMAGH
97330	2015-08	-6.278889	54.86499	george street	ASBO	PORTADOWN
146854	2015-11	-6.457149	54.43173	ballyoran heights	VISO	PORTADOWN
300797	2016-11	-6.338295	54.45736	sloan street	BURG	LISBURN
448064	2017-10	-5.970893	54.62690	silverstream parade	Other theft	BELFAST
314169	2016-12	-6.281757	54.85361	montague park	DRUG	BALLYMENA

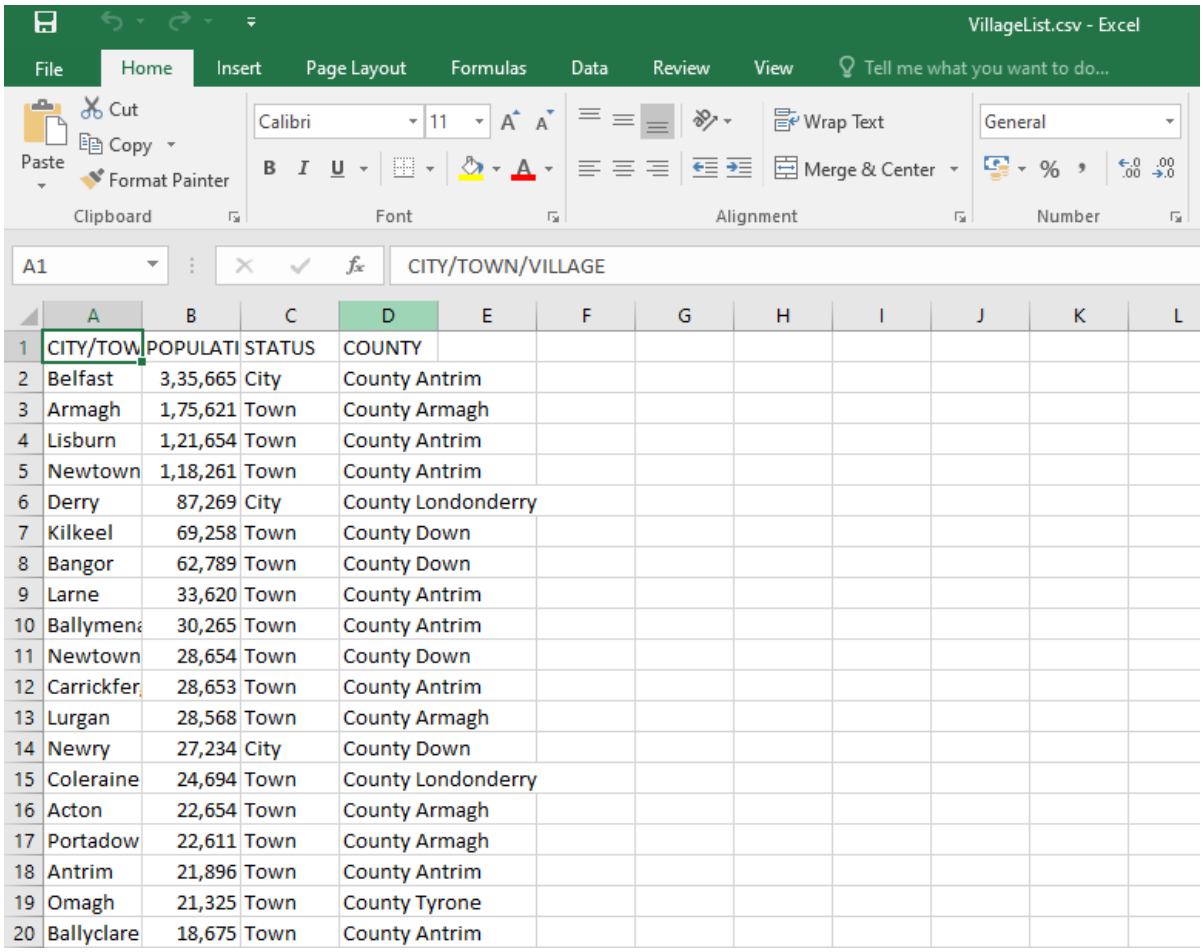
Showing 1 to 18 of 5,000 entries, 6 total columns

Fig.19: Output after the match and adding the town

```
> str(random_crime_sample)
'data.frame':   5000 obs. of  6 variables:
 $ Month     : Factor w/ 36 levels "2015-01","2015-02",..: 8 30 15 8 19 31 8 16 10 19 ...
 $ Longitude : num  -5.93 -7.48 -5.94 -5.96 -6.22 ...
 $ Latitude  : num  54.6 54.8 54.6 54.6 54.5 ...
 $ Location  : chr  "dublin road" "daisy park" "king street" "tyndale gardens" ...
 $ Crime.type: chr  "ASBO" "ASBO" "ASBO" "VECR" ...
 $ Town      : Factor w/ 314 levels "","AGHAGALLON",..: 260 287 46 46 127 177 251 265 46 94 ...
```

Fig.20: structure of the random_crime_sample

G) Now we need to create a function called add_town_data which examines each file of crime data in the random_crime_sample and matches the relevant information in the village_list dataset.



	A	B	C	D
1	CITY/TOW	POPULATI	STATUS	COUNTY
2	Belfast	3,35,665	City	County Antrim
3	Armagh	1,75,621	Town	County Armagh
4	Lisburn	1,21,654	Town	County Antrim
5	Newtown	1,18,261	Town	County Antrim
6	Derry	87,269	City	County Londonderry
7	Kilkeel	69,258	Town	County Down
8	Bangor	62,789	Town	County Down
9	Larne	33,620	Town	County Antrim
10	Ballymena	30,265	Town	County Antrim
11	Newtown	28,654	Town	County Down
12	Carrickfer	28,653	Town	County Antrim
13	Lurgan	28,568	Town	County Armagh
14	Newry	27,234	City	County Down
15	Coleraine	24,694	Town	County Londonderry
16	Acton	22,654	Town	County Armagh
17	Portadow	22,611	Town	County Armagh
18	Antrim	21,896	Town	County Antrim
19	Omagh	21,325	Town	County Tyrone
20	Ballyclare	18,675	Town	County Antrim

Fig.21: Village_list.csv file

Code:

In the first line of the code we have read the village_list.csv file. Now we have changes the city/town information to upper case by using str_to_upper. From the above figure we can see that the city Londonderry is names as Derry as it will be unmatched with the random_crime_sample. So we have replaced Derry with Londonderry by using the function str_replace_all.

```
# Reading village data
village_data  <- read.csv("VillageList.csv")
# Changing the string into upper case
village_data$i..CITY.TOWN.VILLAGE  <- str_to_upper(village_data$i..CITY.TOWN.VILLAGE)
village_data$i..CITY.TOWN.VILLAGE  <- str_replace_all(village_data$i..CITY.TOWN.VILLAGE, "DERRY", "LONDONDERRY")
```

Fig.22: changing the strings

Output:

Functions.R × | CA2.R* × | NIPOSTCODE × | limavady_data × | section2(

Filter

	ï..CITY.TOWN.VILLAGE	POPULATION	STATUS	COUNTY
1	BELFAST	335,665	City	County Antrim
2	ARMAGH	175,621	Town	County Armagh
3	LISBURN	121,654	Town	County Antrim
4	NEWTOWNABBEY	118,261	Town	County Antrim
5	LONDONDERRY	87,269	City	County Londonderry
6	KILKEEL	69,258	Town	County Down
7	BANGOR	62,789	Town	County Down
8	LARNE	33,620	Town	County Antrim
9	BALLYMENA	30,265	Town	County Antrim
10	NEWTOWNARDS	28,654	Town	County Down
11	CARRICKFERGUS	28,653	Town	County Antrim
12	LURGAN	28,568	Town	County Armagh
13	NEWRY	27,234	City	County Down
14	COLERAINE	24,694	Town	County Londonderry
15	ACTON	22,654	Town	County Armagh
16	PORTADOWN	22,611	Town	County Armagh

Showing 1 to 18 of 510 entries, 4 total columns

Fig.23: change in string values to upper case and change Derry to Londonderry

Now we need to create a function add_town_data and match the town attribute within the random_crime_sample to the city/town/village in the village dataset. And store the matched population data from the two datasets into the population attribute in the random_crime_sample.

```
# Creating a function to match the each crime data to relevant information in village list
add_town_data <- function(x,y){

  return(village_data$POPULATION[match(x,y)])
}
# storing it into the random_crime_sample
random_crime_sample$Population <- add_town_data(random_crime_sample$Town, village_data$ï..CITY.TOWN.VILLAGE)
```

Fig.24: creating a function add_town_data

Output:

Functions.R × | CA2.R* × | NIPOSTCODE × | limavady_data × | section2(CA2).R* × | village_data × | rand

Filter

	Month	Longitude	Latitude	Location	Crime.type	Town	Population
95810	2015-08	-5.933066	54.59109	dublin road	ASBO	OMAGH	21,325
384531	2017-06	-7.476470	54.78748	daisy park	ASBO	SION MILLS	2,064
189080	2016-03	-5.935054	54.59857	king street	ASBO	BELFAST	335,665
104404	2015-08	-5.963882	54.63147	tyndale gardens	VECR	BELFAST	335,665
241344	2016-07	-6.223332	54.48017	meeting street	ASBO	DROMORE	5,254
410624	2017-07	-5.839762	54.63121	ardnagreena gardens	VISO	HOLYWOOD	12,657
103416	2015-08	-5.896897	54.21220	shimna mile	PUBO	NEWCASTLE	7,524
206367	2016-04	-6.431276	54.43006	grantham park	CDAR	PORTADOWN	22,611
133178	2015-10	-5.930067	54.60018	rosemary street	SHOP	BELFAST	335,665
241430	2016-07	-6.739900	54.63720	stewart avenue	ASBO	COOKSTOWN	10,718
219424	2016-05	-7.281825	54.60935	knockshee park	CDAR	OMAGH	21,325
97330	2015-08	-6.278889	54.86499	george street	ASBO	PORTADOWN	22,611
146854	2015-11	-6.457149	54.43173	ballyoran heights	VISO	PORTADOWN	22,611
300797	2016-11	-6.338295	54.45736	sloan street	BURG	LISBURN	121,654
448064	2017-10	-5.970893	54.62690	silverstream parade	Other theft	BELFAST	335,665
314169	2016-12	-6.281757	54.85361	montague park	DRUG	BALLYMENA	30,265

Showing 1 to 18 of 5,000 entries, 7 total columns

Fig.25: output after adding the population attribute

```
> str(random_crime_sample)
'data.frame':   5000 obs. of  7 variables:
 $ Month     : Factor w/ 36 levels "2015-01","2015-02",..: 8 30 15 8 19 31 8 16 10 19 ...
 $ Longitude : num  -5.93 -7.48 -5.94 -5.96 -6.22 ...
 $ Latitude  : num  54.6 54.8 54.6 54.6 54.5 ...
 $ Location  : chr  "dublin road" "daisy park" "king street" "tyndale gardens" ...
 $ Crime.type: chr  "ASBO" "ASBO" "ASBO" "VECR" ...
 $ Town      : Factor w/ 314 levels "","AGHAGALLON",..: 260 287 46 46 127 177 251 265 46 94 ...
 $ Population: Factor w/ 387 levels "1,021","1,062",..: 167 127 232 232 275 70 333 174 232 52 ...
```

Fig.26: structure of random_crime_sample

H) Now we need to update the random_crime_sample so that it only contains the attributes Month, Longitude, Latitude, Location, Crime type, City-Town-Village, Population.

```
# changing the column name of town to city-town-village
colnames(random_crime_sample)[6] <- c("City-Town-Village")
```

Fig.27: changing the column name

As we can see in the above structure the town attribute is in the 6th index position we need to change the name to city-town-village by creating the attribute name and change in the random_crime_sample.

Output:

Functions.R × | CA2.R* × | NIPOSTCODE × | limavady_data × | section2(CA2).R* × | village_data × | rand

Filter

	Month	Longitude	Latitude	Location	Crime.type	City-Town-Village	Population
95810	2015-08	-5.933066	54.59109	dublin road	ASBO	OMAGH	21,325
384531	2017-06	-7.476470	54.78748	daisy park	ASBO	SION MILLS	2,064
189080	2016-03	-5.935054	54.59857	king street	ASBO	BELFAST	335,665
104404	2015-08	-5.963882	54.63147	tyndale gardens	VECR	BELFAST	335,665
241344	2016-07	-6.223332	54.48017	meeting street	ASBO	DROMORE	5,254
410624	2017-07	-5.839762	54.63121	ardnagreena gardens	VISO	HOLYWOOD	12,657
103416	2015-08	-5.896897	54.21220	shimna mile	PUBO	NEWCASTLE	7,524
206367	2016-04	-6.431276	54.43006	grantham park	CDAR	PORTADOWN	22,611
133178	2015-10	-5.930067	54.60018	rosemary street	SHOP	BELFAST	335,665
241430	2016-07	-6.739900	54.63720	stewart avenue	ASBO	COOKSTOWN	10,718
219424	2016-05	-7.281825	54.60935	knockshee park	CDAR	OMAGH	21,325
97330	2015-08	-6.278889	54.86499	george street	ASBO	PORTADOWN	22,611
146854	2015-11	-6.457149	54.43173	ballyoran heights	VISO	PORTADOWN	22,611
300797	2016-11	-6.338295	54.45736	sloan street	BURG	LISBURN	121,654
448064	2017-10	-5.970893	54.62690	silverstream parade	Other theft	BELFAST	335,665
314169	2016-12	-6.281757	54.85361	montague park	DRUG	BALLYMENA	30,265

Showing 1 to 17 of 5,000 entries, 7 total columns

Fig.28: Output after changing the attribute name

I) We need to plot the crimes in the cites Belfast and Londonderry. We need to show the plot data side by side.

Code: first we need create to variables and use the subset function to group the crimes in the cities Belfast and Londonderry. And use the libraries dplyr and ggplot for graphics. In the code we can see that we have grouped by crime type and summarise the frequency counts and change the label names.

```
# I)
# Ploting the crimes in cities Belfast and Londonderry
random_crime_belfast <- subset(random_crime_sample, random_crime_sample$`City-Town-Village` == "BELFAST")
random_crime_derry <- subset(random_crime_sample, random_crime_sample$`City-Town-Village` == "LONDONDERRY")


library(dplyr)
library(ggplot2)
# Step 1
plot1 <- random_crime_belfast %>%
  #Step 2
  group_by(Crime.type) %>%
  #Step 3
  summarise(count_fre = n()) %>%
  #Step 4
  ggplot(aes(x = Crime.type, y = sort(count_fre, decreasing = TRUE), fill = Crime.type)) +
  geom_bar(stat = "identity") +
  theme_classic() +
  labs(
    x = "Crime.type",
    y = "Count",
    title = paste(
      "Crimes in Belfast"
    )
  )
```

Fig.29: code for plotting the crimes in Belfast

```r
plot2 <- random_crime_derry %>%
  #Step 2
  group_by(Crime.type) %>%
  #Step 3
  summarise(count_fre = n()) %>%
  #Step 4
  ggplot(aes(x = Crime.type, y = sort(count_fre, decreasing = TRUE), fill = Crime.type)) +
  geom_bar(stat = "identity") +
  theme_classic() +
  labs(
    x = "Crime.type",
    y = "Count",
    title = paste(
      "Crimes in Londonderry"
    )
  )

install.packages("gridExtra")
require(gridExtra)
grid.arrange(plot1,plot2,ncol=2)
```

Fig.30: code for plotting the crimes in Londonderry

Output: From the below output we can see that anti-social behavior is the highest in the both cities. when compared to Bicycle theft Londonderry has slightly highest frequency than Belfast. Violence and sexual offence is the least in both the cities. Burglary and criminal damage and arson are vice versa in both the cities.