LAPORAN
FINAL PROJECT 1
Visualisasi Data Online Retail
(Soal 1)

Humaira Nur Pradani (05211640000011)
Nevada Veterino (05211640000096)
Rizki Ahmad Fauzi (05211640000111)

Analitika BIsnis – D
Sistem Informasi
Institut Teknologi Sepuluh Nopember

TAHAP PREPROCESSING DATA

Data Preparation atau bisa disebut juga dengan data preprocessing adalah suatu proses/langkah yang dilakukan untuk membuat data mentah menjadi data yang berkualitas (input yang baik untuk data mining tools). Pada studi kasus kali ini, sebelumnya kita lakukan data preprocessing sebagai berikut.

SYNTAX PREPOCESSING DATA

```
# Cleaning the data and removing null/missing/ negative values and
creating subset and converting to factors for variables like
InvoiceNo,StockCode, InvoiceDate, CustomerID, Country.

library(stringr)
isUndesirable2 = function(x) {
  str_detect(toupper(x),"WRONG") | str_detect(toupper(x),"LOST") |
    str_detect(toupper(x),"CRUSHED") |
str_detect(toupper(x),"DAMAGE") |
    str_detect(toupper(x),"FOUND") |
str_detect(toupper(x),"THROWN") |
    str_detect(toupper(x),"SMASHED") |
    str_detect(toupper(x),"\\?") |
    str_detect(toupper(x),"AWAY") |
str_detect(toupper(x),"CHARGES") |
    str_detect(toupper(x),"FEE") | str_detect(toupper(x),"FAULT")
  str_detect(toupper(x),"SALES") | str_detect(toupper(x),"ADJUST")
|
    str_detect(toupper(x),"COUNTED") |
    str_detect(toupper(x),"INCORRECT") |
    str_detect(toupper(x),"BROKEN") |
str_detect(toupper(x),"BARCODE") |
    str_detect(toupper(x),"RETURNED") |
    str_detect(toupper(x),"MAILOUT") |
str_detect(toupper(x),"DELIVERY") |
    str_detect(toupper(x),"MIX UP") |
str_detect(toupper(x),"MOULDY") |
    str_detect(toupper(x),"PUT ASIDE") |
str_detect(toupper(x),"ERROR") |
    str_detect(toupper(x),"DESTROYED") |
str_detect(toupper(x),"RUSTY") |
    str_detect(toupper(x),"MANUAL") |
str_detect(toupper(x),"AMAZON") |
    str_detect(toupper(x),"FEE") |
str_detect(toupper(x),"POSTAGE") |
    str_detect(toupper(x),"PADS") | str_detect(x, "Bank")
}

EcommDataClean <- Ecommdata[which(Ecommdata$Quantity>=0 &
!is.na(Ecommdata$Quantity) &  Ecommdata$UnitPrice >=0 &
!is.na(Ecommdata$UnitPrice) & !is.na(Ecommdata$CustomerID)),]

EcommDataClean$InvoiceNo<- factor(EcommDataClean$InvoiceNo)
```

```
EcommDataClean$StockCode<- factor(EcommDataClean$StockCode)
EcommDataClean$CustomerID<- factor(EcommDataClean$CustomerID)
EcommDataClean$Country<- factor(EcommDataClean$Country)
EcommDataClean$InvoiceDate<- as.Date(EcommDataClean$InvoiceDate,
'%m/%d/%Y %H:%M')

# Adding a new column of total purchase price i.e( Unit price *
Quantity)

EcommDataClean$TotalPrice <- EcommDataClean$UnitPrice *
EcommDataClean$Quantity

#Adding three columns Day, month and year by splitting the invoice
date.

EcommDataClean$DateYear <-
as.numeric(format(EcommDataClean$InvoiceDate, format = "%Y"))
EcommDataClean$DateMonth <-
as.numeric(format(EcommDataClean$InvoiceDate, format = "%m"))
EcommDataClean$DateDay <-
as.numeric(format(EcommDataClean$InvoiceDate, format = "%d"))

# Converting them to factors.

EcommDataClean$DateYear<- factor(EcommDataClean$DateYear)
EcommDataClean$DateMonth<- factor(EcommDataClean$DateMonth)
EcommDataClean$DateDay<- factor(EcommDataClean$DateDay)
EcommDataClean$Quarter<- factor(EcommDataClean$Quarter)

#Menjalankan fungsi isundesirable
EcommDataClean =
EcommDataClean[which(!isUndesirable2(as.character(EcommDataClean$D
escription))),]

#Menghilangkan Carriage
EcommDataClean =
EcommDataClean[which(!startsWith(as.character(EcommDataClean$Stock
Code), "C")),]

#View(EcommDataClean)
```

Hal-hal yang dilakukan pada proses diatas adalah :

- ✓ Menghilangkan data null dan negatif
- ✓ Membuat kolom baru bernama TotalPrice yang merupakan *derived column* dari pengalian kolom UnitPrice dan Quantity
- ✓ Membuat masing-masing variabel menjadi faktor
- ✓ Menghilangkan data-data item yang tidak diperlukan (M-Manual, C2-Carriage, Bank Charges, dll.

SOAL 1a

Temukan dan buatlah visualisasi dari hal-hal berikut ini dari data Online Retail. Jumlah pelanggan unik dan besar total nilai transaksi masing-masing. Buatlah visualisasinya dalam bentuk bar plot.

Jawaban 1a

Untuk menjawab pertanyaan diatas, kita menggunakan hasil data yang telah dilakukan *preprocessing data* pada tahap sebelumnya (EcommDataClean). Tahap selanjutnya adalah melakukan agregat dengan fungsi penjumlahan untuk melihat daftar total nilai transaksi masing-masing pelanggan dan disimpan pada data frame bernama OverallPricePerCustomer. Pada R, agregat dilakukan dengan memanggil fungsi aggregate() sebagai berikut.

AGREGASI DATA

```
OverallPricePerCustomer <-  aggregate(EcommDataClean$TotalPrice,
by=list(CustomerID=dataset5$CustomerID), FUN=sum)
```

Setelah mendapatkan daftar customer dan total nilai transaksi yang dilakukan masing-masing customer, langkah selanjutnya adalah membuat barplot. Untuk membuat barplot, dapat dilakukan dengan beberapa cara, salah satunya dalah memamggil fungsi barplot() tanpa harus memasukkan beberapa pendefinisian yang dilakukan adalah :

- ✓ ***height = OverallPricePerCustomer$x***

  Menentukan ketinggian tiap bar pada barplot dipengaruhi oleh nilai x(total price)
- ✓ ***names.arg = OverallPricePerCustomer$CustomerID***

  Label untuk setiap bar yang dibentuk adalah berdasarkan nilai dari CustomerID
- ✓ ***xlab = "Pelanggan"***

  Label pada sumbu x adalah “Pelanggan”
- ✓ ***ylab = "Total Transaksi"***

  Label pada sumbu y adalah “Total Transaksi”
- ✓ ***main = "Total Transaksi per Pelanggan Unik"***

  Memberikan judul pada barplot yang terbentuk sebaai

MEMBUAT BARPLOT

```
barplot(height = OverallPricePerCustomer$x, names.arg =
OverallPricePerCustomer$CustomerID, xlab = "Pelanggan", ylab =
"Total Transaksi", main = "Total Transaksi per Pelanggan Unik")
```

Setelah menerapkan syntax tersebut, maka berikut hasil yang akan didapat. Pada tabel OverallPricePerCustomer, kolom customerID adalah kolom yang mengidentifikasi pelanggan yang melakukan pembelian, dan x mengidenfikasi jumlah nilai transaksi yang dilakukan oleh pelanggan tersebut.

HASIL TABEL (OverallPricePerCustomer)

	CustomerID	x
1	12346	77183.60
2	12347	4310.00
3	12348	1437.24
4	12349	1457.55
5	12350	294.40
6	12352	1365.94
7	12353	89.00
8	12354	1079.40
9	12355	459.40
10	12356	2487.43
11	12357	6121.27
12	12358	928.06

Showing 1 to 13 of 4,334 entries

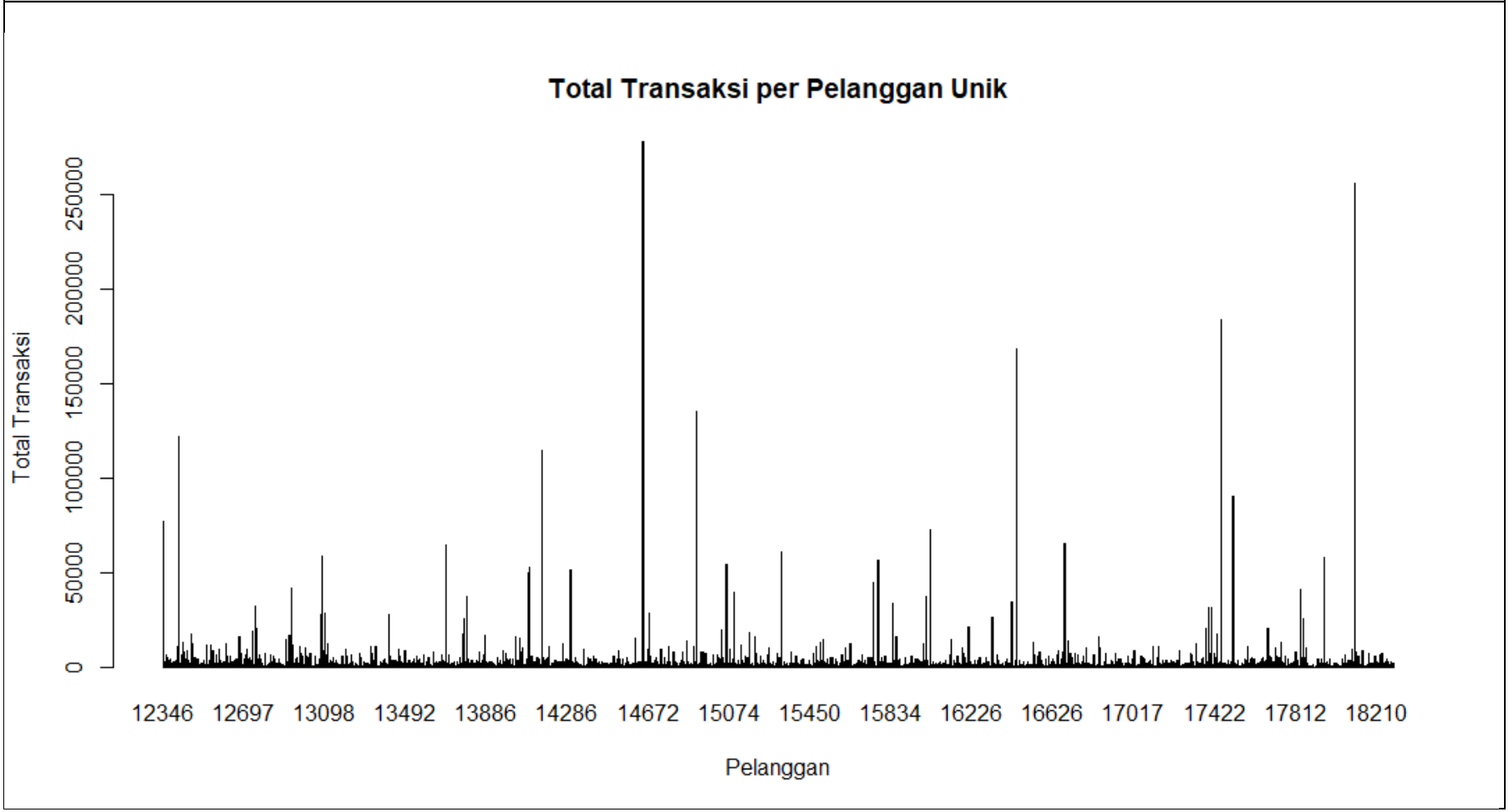
Diurutkan berdasarkan **nilai transaksi paling besar** :

	CustomerID	x
1691	14646	278197.07
4198	18102	256424.50
3726	17450	183984.06
3008	16446	168472.50
1881	14911	135387.18
56	12415	122461.38
1335	14156	114654.98
3769	17511	90372.14

Dari hasil tersebut, dapat dilihat bahwa jumlah customer keseluruhan adalah 4.334. Untuk data customer yang melakukan transaksi paling banyak adalah customer dengan ID 14646 dengan melakukan pembelian sebesar 287.197,07, dan kedua terbanyak adalah customer dengan ID 18102 , kemudian diikuti oleh 17450, 16446, 14911, 12415, 15156, dst.

Hasil visualisasi dari data berbentuk barplot dapat dilihat pada halaman selanjutnya.

HASIL BARPLOT



Pada barplot di atas diperlihatkan IDCustomer yang divisualisasikan dengan sumbu X berjudul Pelanggan yang memiliki banyak transaksi pada sumbu Y yang diberi judul Total Transaksi. Setiap IDCustomer dihubungkan dengan jumlah total transaksi yang dikeluarkan. Dari total 4.334 pelanggan diperlihatkan beberapa pelanggan pada barplot tersebut. Didapati pelanggan paling banyak melakukan transaksi yaitu pelanggan dengan IDCustomer 14646, dengan total transaksi 287.197,07, diikuti dengan IDCustomer 18102, lalu IDCustomer 17450, 16446, 14911, dan seterusnya.

SOAL 1b

Jumlah negara yang terlibat transaksi dan jumlah pelanggan unik dari masing-masing negara. Buatlah visualisasinya dalam bentuk bar plot.

Jawaban 1b

Untuk menjawab pertanyaan nomor 1b maka digunakan data yang sudah melalui tahap *preprocessing* (EcommDataClean). Kemudian langkah selanjutnya digunakan library ggplot2, plyr, dan dplyr. Library ggplot2 digunakan untuk membuat barplot. Sedangkan library plyr serta dplyr pada studi kasus kali ini digunakan dalam melakukan agregasi data.

MENGIMPORT LIBRARY

```
library(ggplot2)
library(plyr)
library(dplyr)
```

Setelah melakukan pemanggilan terhadap library ggplot2, plyr, dan dplyr, dilakukan pembuatan tabel yang akan memunculkan kolom *Country*, *CustomerID*, dan *Frequency* dengan syntax sebagai berikut.

MEMBUAT DATAFRAME PERTAMA

```
CustomerbyCountry <- select(EcommDataClean, c(7,8))
count_CustIDRecord <- ddply(CustomerbyCountry,
.(CustomerbyCountry$Country, CustomerbyCountry$CustomerID), nrow)
names(count_CustIDRecord) <- c("Country", "CustomerID",
"Frequency")
```

Dataframe pertama dibentuk untuk melakukan roll-up pada data sehingga data dikelompokkan per CustomerID berdasarkan negaranya, sebagai berikut. Sebagai contoh, pada baris pertama pada negara Australia *CustomerID* bernomor 12386 melakukan transaksi sebanyak 10 kali.

	Country	CustomerID	Frequency
1	Australia	12386	10
2	Australia	12388	99
3	Australia	12393	64

Gambar 1 Sampel Dataframe count_CustIDRecord

Lalu setelah itu dibuatlah tabel yang akan menampilkan customer unik yang melakukan transaksi pada negara-negara yang terdapat pada data. Pada tabel tersebut ditampilkan 2 kolom yaitu kolom *Country* dan kolom *Number of Unique Customer*.

MEMBUAT DATAFRAME KEDUA

```
CustomerbyCountry <- select(count_CustIDRecord, c(1))
```

```
count_CustIDRecord <- ddply(CustomerbyCountry,
.(CustomerbyCountry$Country), nrow)
names(count_CustIDRecord) <- c("Country", "Number of Unique
Customer")
CustomerbyCountry <- count_CustIDRecord
```

MENAMPILKAN DATA FRAME (CustomerbyCountry)

```
View(CustomerbyCountry)
```

Jika dilihat dari tabel yang dibuat, pada baris satu pada kolom *Country* terdapat negara Australia dan pada kolom *Number of Unique Customer* tertulis 9, maka dapat diartikan pada negara Australia terdapat 9 Customer yang bertransaksi dengan *CustomerID* yang berbeda.

	Country	Number of Unique Customer
1	Australia	9
2	Austria	11
3	Bahrain	2
4	Belgium	25
5	Brazil	1
6	Canada	4
7	Channel Islands	9
8	Cyprus	8
9	Czech Republic	1

Gambar 2 Sampel Data CustomerbyCountry

Langkah selanjutnya yaitu dengan membuat Barplot dari tabel yang sudah dibuat untuk menampilkan grafik yang menjelaskan hubungan antara negara dengan jumlah pelanggan unik. Pada Barplot yang telah dibuat dapat dilihat bahwa negara dengan jumlah customer unik paling banyak adalah negara United Kingdom.

MEMBUAT BARPLOT

```
ggplot(CustomerbyCountry,
aes(x=reorder(CustomerbyCountry$Country,-CustomerbyCountry$`Number
of Unique Customer`), y=CustomerbyCountry$`Number of Unique
Customer`)) +
  geom_bar(stat="identity", fill="#009999") + theme_bw() +
  xlab("Negara") +
  ylab("Jumlah Pelanggan Unik") +
  ggtitle("Jumlah Pelanggan Unik dalam Tiap Negara")
```

Untuk membuat barplot tersebut, ada beberapa hal yang harus didefinisikan :

- ✓ **aes(x=reorder(CustomerbyCountry$Country,-CustomerbyCountry$`Number of Unique Customer`), y=CustomerbyCountry$`Number of Unique Customer`))**

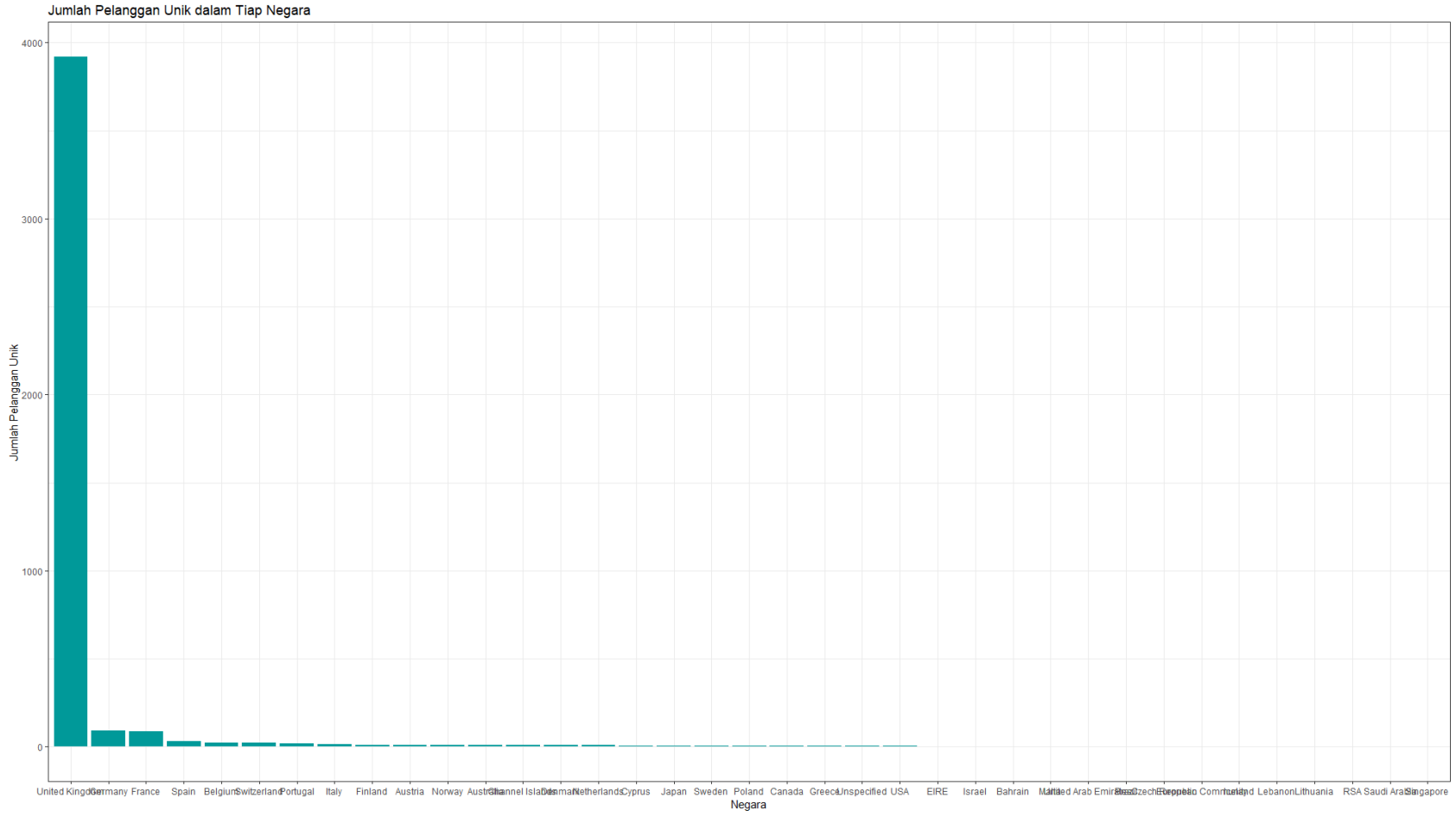
mendefinisikan bahwa pada sumbu x, data yang digunakan adalah data Country atau negara yang dimana sudah di urutkan berdasarkan jumlah customer dari yang terkecil hingga terbanyak

- ✓ **geom_bar(stat="identity", fill="#009999")**
  mendefinisikan bahwa plot yang dibuat adalah berjenis barplot dengan tiap barnya terisi dengan warna biru keungu-unguan
- ✓ **theme_bw()**
  mendefinisikan tema yang digunakan untuk background/grid pada barplot adalah tema black and white
- ✓ **xlab("Negara")**
  mendefinisikan label pada sumbu x adalah “Negara”
- ✓ **ylab("Jumlah Pelanggan Unik")**
  mendefinisikan label pada sumbu y adalah “Jumlah Pelanggan Unik”
- ✓ **ggtitle("Jumlah Pelanggan Unik dalam Tiap Negara")**
  mendefinisikan judul dari sebuah barplot yang terbentuk adalah “Jumlah Pelanggan Unik Tiap Negara

Pada barplot yang telah dibuat dari script di atas diperlihatkan Negara yang divisualisasikan dengan sumbu X yang dihubungkan dengan Jumlah Pelanggan Unik yang divisualisasikan dengan sumbu Y. Pada barplot yang ditampilkan, memperlihatkan dimana sejumlah pelanggan unik yaitu pelanggan dengan IDCustomer yang berbeda-beda melakukan transaksi pada tiap-tiap negara yang ada pada data Online Retail.

Hasil dari barplot ada pada halaman selanjutnya. Dari data tersebut, kita dapat melihat bahwa total yang melakukan transaksi pada Online Retail sebanyak 37 negara. Kemudian negara yang terdapat traksaksi terbanyak adalah United Kingdom dapat dilihat letaknya ada pada baplot paling kiri dan memiliki bar yang paling besar. Kemudian transaksi tertinggi selanjutnya diikuti oleh Germany, France, dan negara-negara lainnya.

HASIL BARPLOT



SOAL 1c

Item dengan harga termurah, harga termahal, rata-rata dan standar deviasinya. Buatlah visualisasi distribusi harga item dalam bentuk histogram.

Jawaban 1c

Pertanyaan 1 C ini juga menggunakan data yang sudah di bersihkan EcommDataClean dengan menggunakan library ggplot2 untuk melakukan visualisasi persebaran harga produk

MENGIMPORT LIBRARY
`library(ggplot2)`

Setelah memanggil library ggplot2, selanjutnya membuat tabel ProductbyPrice yang berisikan StockCode, Description, dan UnitPirce yang diambil dari tabel EcommDataClean, setelah itu tabel yang sudah terbuat dibersihkan lagi dari duplikasi data dengan menggunakan *distinct*, lalu menyeleksi harga produk sehingga hanya yang mempunyai harga lebih dari 0 yang akan ditampilkan

MEMBUAT DATAFRAME

`ProductbyPrice <- EcommDataClean[,c(2,3,6)]`
`ProductbyPrice <- distinct(ProductbyPrice)`
`ProductbyPrice <- ProductbyPrice[ProductbyPrice$UnitPrice>0,]`

MENAMPILKAN DATAFRAME
`View(ProductbyPrice)`

	StockCode	Description	UnitPrice
1	23166	MEDIUM CERAMIC TOP STORAGE JAR	1.04
2	85116	BLACK CANDELABRA T-LIGHT HOLDER	2.10
3	22375	AIRLINE BAG VINTAGE JET SET BROWN	4.25
4	71477	COLOUR GLASS. STAR T-LIGHT HOLDER	3.25
5	22492	MINI PAINT SET VINTAGE	0.65
6	22771	CLEAR DRAWER KNOB ACRYLIC EDWARDIAN	1.25
7	22772	PINK DRAWER KNOB ACRYLIC EDWARDIAN	1.25
8	22773	GREEN DRAWER KNOB ACRYLIC EDWARDIAN	1.25
9	22774	RED DRAWER KNOB ACRYLIC EDWARDIAN	1.25
10	22775	PURPLE DRAWERKNOB ACRYLIC EDWARDIAN	1.25
11	22805	BLUE DRAWER KNOB ACRYLIC EDWARDIAN	1.25
12	22725	ALARM CLOCK BAKELIKE CHOCOLATE	3.75
13	22726	ALARM CLOCK BAKELIKE GREEN	3.75
14	22727	ALARM CLOCK BAKELIKE RED	3.75

Gambar 3 Sampel Data ProductByPrice

Lalu untuk menampilkan data statistik seperti rata-rata harga, harga tertinggi dan terendah, dan standar deviasi menggunakan *sd, min, max,* dan *mean* pada kolom UnitPrice pada tabel ProductbyPrice dan menghasilkan angka standar deviasi sebesar 136.2116, nilai minimal sebesar 0.001, nilai maximal 8142.75 dan nilai rata-rata sebesar 10.88233.

MENAMPILKAN DATA STATISTIK

```
sd(ProductbyPrice$UnitPrice)
min(ProductbyPrice$UnitPrice)
max(ProductbyPrice$UnitPrice)
mean(ProductbyPrice$UnitPrice)
```

```
> sd(ProductbyPrice$UnitPrice)
[1] 136.2116
> min(ProductbyPrice$UnitPrice)
[1] 0.001
> max(ProductbyPrice$UnitPrice)
[1] 8142.75
> mean(ProductbyPrice$UnitPrice)
[1] 10.88233
```

Gambar 4 Hasil Statistik

Dari tabel ProductbyPrice yang sudah dibuat sebelumnya, kita akan melakukan visualisasi persebaran harga produk berdasarkan banyaknya produk dengan menggunakan gplot dan bentuk histogram, dan binwidth = 20.

MENGIMPORT LIBRARY

```
library(ggplot2)
```

MEMBUAT HISTOGRAM

```
qplot(ProductbyPrice$UnitPrice, geom="histogram",binwidth = 20,
main = "Persebaran Harga Produk", xlab = "Unit Price", ylab =
"Jumlah", fill=I("blue"), alpha=I(.2))
```

Pada histogram yang dibuat dari script di atas, diperlihatkan Unit Price yang divisualisasikan dengan sumbu X yang dihubungkan dengan Jumlah yang divisualisasikan dengan sumbu Y. Pada histogram ini akan diperlihatkan persebaran harga yang terdapat pada data Online Retail. Pada setiap harga barang yang ada dihitung terdapat berapa barang yang memiliki harga tersebut dari harga barang yang paling murah sampai harga barang yang paling mahal.

Pada histogram ini akan terlihat aneh karena persebaran harga produk sangat tidak seimbang. Hal ini dikarenakan terdapat banyak sekali produk yang mempunyai harga sangat kecil (sekitar dibawah 10), lalu terdapat sedikit sekali produk yang mempunyai harga mahal (sekitar diatas 50). Dari persebaran harga yang seperti itu akan menghasilkan histogram yang datanya terlihat seperti menumpuk di angka yang kecil dan terlihat kosong (tidak ada data) namun karena perbandingan skala dengan data dengan harga produk murah menyebabkan tidak terlihat.

HASIL HISTOGRAM