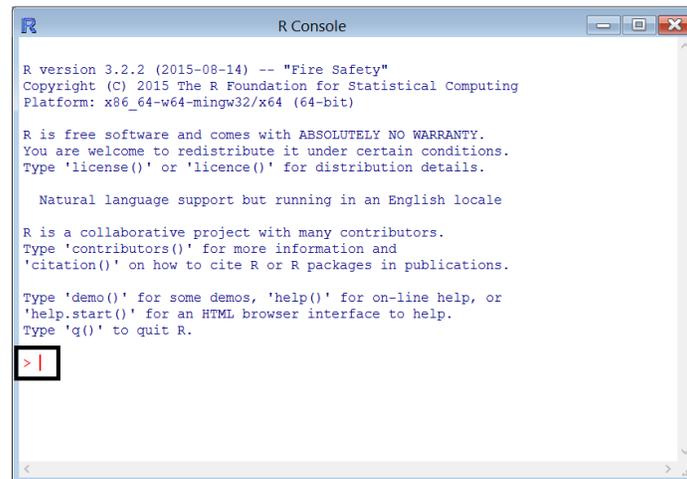


MEMULAI R

- Ketika kita memulai program R, akan muncul tampilan seperti Gambar 1



Gambar 1 R console

- Tanda > merupakan **command prompt** yang menunggu user untuk mengetikkan perintah.

Menulis perintah pada R console

- Misal ingin dihitung $10+20$, maka user dapat mengetikkan secara langsung

```
> 10+20
[1] 30
```

- Berikut adalah beberapa fungsi aritmatika

operation	operator	example input	example output
addition	+	10 + 2	12
subtraction	-	9 - 3	6
multiplication	*	5 * 5	25
division	/	10 / 3	3
power	^	5 ^ 2	25

- R mengenal spasi ,misal

```
> 10+ 20
[1] 30
```

- Contoh operasi aritmatika lain:

```
> 3*3*3*3
[1] 81
> 45^2
[1] 2025
> 1+2*3
[1] 7
> 34-4*2
[1] 26
> sqrt(81)
[1] 9
> abs(-9)
[1] 9
```

Menyimpan data sebagai suatu variabel

- R dapat menyimpan data sebagai suatu variabel. Misal kita ingin mengenakan nilai 350 pada variabel dengan nama sales,

```
> sales<-350
> sales
[1] 350
```

- Misal royalti yang didapat adalah 7 dan ada fungsi `revenue` yang merupakan perkalian sales dan royalti,

```
> royalti<-7
> revenue<-sales*royalti
> revenue
[1] 2450
```

- Fungsi `round` untuk membulatkan angka decimal,

```
> round(3.1415)
[1] 3
> round(3.14165,2)
[1] 3.14
```

Menyimpan data sebagai suatu vektor

- Variabel dalam R yang dapat menyimpan multipel data disebut dengan `vector`

- Misal akan disimpan data 0, 100, 200, 50, 0, 0, 0, 0, 0, 0, 0, 0 maka kita dapat menyimpan data tersebut dengan fungsi `c()`

- Variabel tunggal misal dinamakan `sales.bulan` maka,

```
> sales.bulan<-c(0, 100, 200, 50, 0, 0, 0, 0, 0, 0, 0, 0)
> sales.bulan
[1] 0 100 200 50 0 0 0 0 0 0 0 0
```

- Jika user ingin memanggil data kedua maka,

```
> sales.bulan[2]
[1] 100
```

- Jika 100 tadi ingin dinamakan sebagai variable baru, maka,

```
> datakedua<-sales.bulan[2]
> datakedua
[1] 100
```

- Misalkan data kelima ternyata salah, yang benar adalah 25 maka,

```
> sales.bulan[5]<-25
> sales.bulan
[1] 0 100 200 50 25 0 0 0 0 0 0 0
```

- Fungsi `length()` digunakan untuk mengetahui banyaknya elemen dalam vector

```
> length(x=sales.bulan)
[1] 12
```

- Apabila user ingin mengalikan data pada `sales.bulan` dengan 10. Artinya semua elemen dalam variable dikalikan 10. Hal ini analog dengan penjumlahan, pengurangan, pembagian dan pangkat

```
> sales.bulan*10
[1] 0 1000 2000 500 250 0 0 0 0 0 0 0
```

- Misal akan didefinisikan vector baru,

```
> data.baru<-c(31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31)
```

- Dan didefinisikan variable `untung=sales.bulan*7`,

```
> untung<-sales.bulan*7
> untung
[1] 0 700 1400 350 175 0 0 0 0 0 0 0
```

- Selanjutnya, dicari pembagian `untung` dengan data baru, sehingga diperoleh

```
> untung/data.baru
[1] 0.000000 25.000000 45.161290 11.666667 5.645161 0.000000 0.000000
[8] 0.000000 0.000000 0.000000 0.000000 0.000000
```

Menyimpan data berupa Text

- R juga dapat membuat data text,

```
> salam<-"Assalamualaikum"
> salam
[1] "Assalamualaikum"
```

- Analog dengan data numeric, data text juga dapat menggunakan fungsi c() dan memanggilnya kembali

```
> Hari<-c("Senin", "Selasa", "Rabu", "Kamis", "Jumat", "Sabtu", "Ahad")
> Hari
[1] "Senin" "Selasa" "Rabu" "Kamis" "Jumat" "Sabtu" "Ahad"
> Hari[7]
[1] "Ahad"
```

- Dari contoh diatas, nampak bahwa karakter vector terdiri dari 7 elemen
- Fungsi nchar() digunakan untuk menghitung jumlah karakter dari elemen yang telah didefinisikan oleh user

```
> nchar(x=Hari)
[1] 5 6 4 5 5 5 4
```

Menyimpan data True or False

- Operator == dapat digunakan dalam R untuk menuliskan true or false, contoh

```
> 3+3==6
[1] TRUE
> 3+3==7
[1] FALSE
```

- Beberapa operasi logic

operation	operator	example input	answer
less than	<	2 < 3	TRUE
less than or equal to	<=	2 <= 2	TRUE
greater than	>	2 > 3	FALSE
greater than or equal to	>=	2 >= 2	TRUE
equal to	==	2 == 3	FALSE
not equal to	!=	2 != 3	TRUE

operation	operator	example input	answer
not	!	!(1==1)	FALSE
or		(1==1) (2==3)	TRUE
and	&	(1==1) & (2==3)	FALSE

- Contoh

```
> 100<90
[1] FALSE
> 90<100
[1] TRUE
> sqrt(25)==5
[1] TRUE
> (2+2 == 4) | (2+2 == 5)
[1] TRUE
> (2+2 == 4) & (2+2 == 5)
[1] FALSE
> ! (2+2 == 5)
[1] TRUE
> 2+2 != 5
[1] TRUE
```

Vector dari logika

- Sama seperti penyimpanan data numeric maupun text, fungsi c() juga dapat digunakan

```
> x<-c(TRUE, TRUE, FALSE)
> x
[1] TRUE TRUE FALSE
```

- Misal menggunakan data sales.bulan, user ingin mengetahui sales perbulannya,

```
> sales.bulan>0
[1] FALSE TRUE TRUE TRUE TRUE FALSE FALSE FALSE FALSE FALSE FALSE
```

- Contoh lain

```
> "kucing"=="kucing"
[1] TRUE
> " kucing"=="kucing"
[1] FALSE
```

Mengekstrak multiple elemen

- Contoh

```
> sales.bulan[c(1,2)]
[1] 0 100
> Hari[c(5,7)]
[1] "Jumat" "Ahad"
```

- Atau penjualan dari bulan ke 1 sampai dengan 5,

```
> sales.bulan[1:5]
[1] 0 100 200 50 25
```

- Penjualan yang besarnya >0,

```
> sales.bulan[sales.bulan>0]  
[1] 100 200 50 25
```

Quiting R

- Jika user ingin log out dari R maka dapat menggunakan fungsi q()

