

**MODUL PELATIHAN PENGABDIAN KEPADA MASYARAKAT**

*PERENCANAAN DAN ANALISIS DATA SURVEI WILAYAH PENDUKUNG (SWP)  
DALAM MENENTUKAN TYPOLOGY PEMBANGUNAN DESA  
DI KABUPATEN BANDUNG*

**MODUL 4:**

***ANALISIS DATA SWP MENGGUNAKAN R***



Disusun oleh:

Budhi Handoko, S.Si, M.Si.

**DEPARTEMEN STATISTIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS PADJADJARAN**

**2022**

## Daftar Isi

1. Menginstall R dan R Studio .....	3
2. Lingkungan RStudio.....	5
3. Mengimpor data ke R .....	6
4. Fungsi yang diperlukan.....	7
4.1 Fungsi <code>attach()</code> .....	7
4.2 Fungsi <code>return()</code> .....	8
4.3 Fungsi <code>apply()</code> .....	8
4.4 Fungsi <code>data.frame()</code> .....	8
5. Membuat fungsi.....	9
6. Kode R untuk menghitung IPD.....	10

## 1. Menginstall R dan R Studio

Sebelum memulai analisis, terlebih dahulu kita harus menginstall software *R* kedalam laptop atau komputer. Software *R* yang akan diinstall ada dua macam, yaitu *R* dan *R Studio*. *R Studio* dapat diinstall jika *R* sudah terinstall lebih dahulu.

Software *R* dapat diunduh dari website CRAN (*The Comprehensive R Archive Network*) yaitu

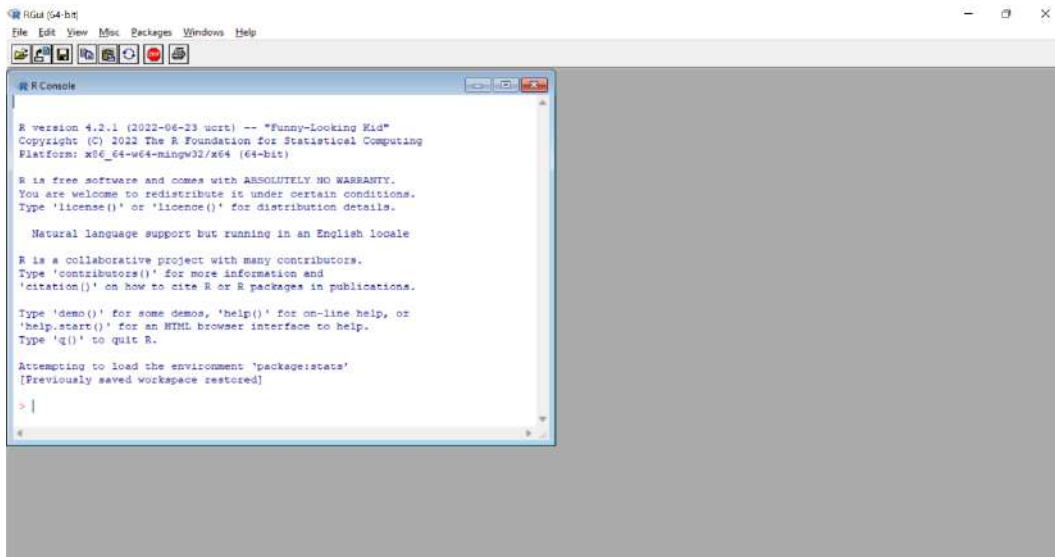
<https://cran.r-project.org>

atau bisa dicari melalui Google dengan kata kunci: *CRAN*.

Apabila laptop atau komputer kita menggunakan sistem operasi Windows, maka software *R* yang dapat diunduh melalui link *Download R for Windows*. Jika menggunakan sistem operasi lain seperti Linux atau macOS bisa mengklik link download yang sesuai pada pilihan yang tersedia pada jendela *Download and Install R* di CRAN. Pada modul ini, instalasi *R* akan difokuskan untuk sistem operasi Windows.

Klik link *Download R for Windows*, kemudian klik *install R for the first time* pada subdirektori *base*. Selanjutnya klik pada link *Download R-4.2.1 for Windows*. Versi *R* terbaru saat ini per Agustus 2022 adalah *R* versi 4.2.1. Versi tersebut dapat berubah jika ada pembaruan menjadi versi yang lebih baru.

Setelah diunduh, install software *R* tersebut dengan cara mengklik dua kali dan ikuti langkah-langkah instalasinya sesuai petunjuk pada jendela yang muncul. Setelah selesai instalasi, jika dibuka maka tampilan software *R* nya adalah seperti berikut ini:



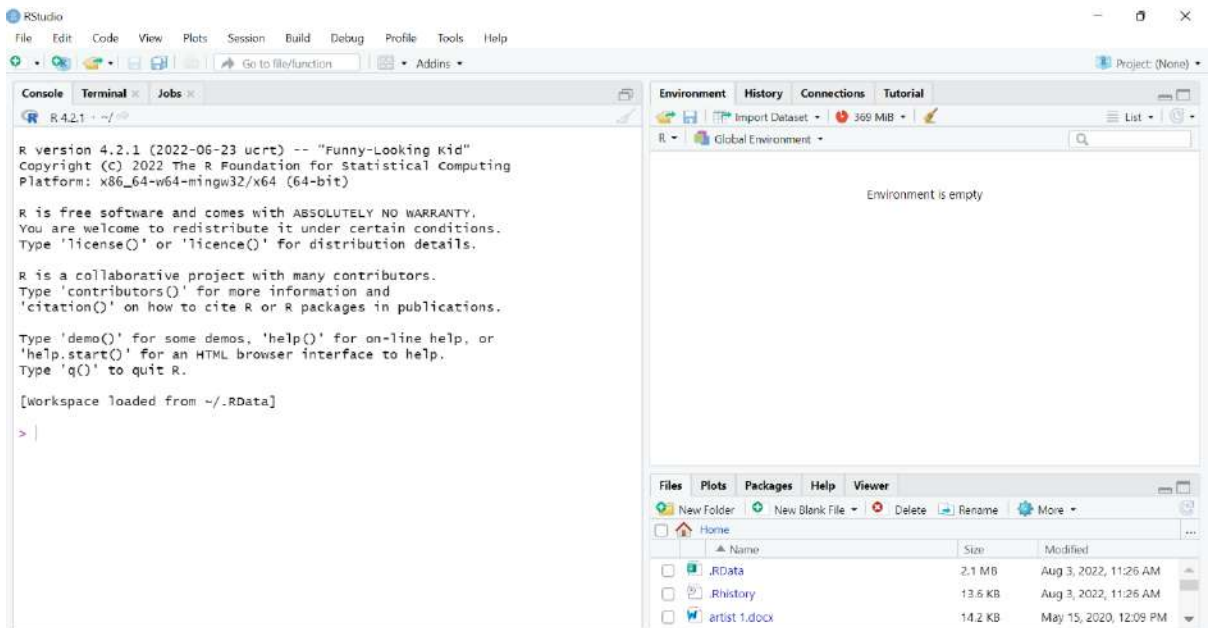
Setelah itu, kita melakukan instalasi R Studio. Software R studio dapat diunduh melalui website berikut ini:

<https://www.rstudio.com>

atau bisa dicari menggunakan Google dengan kata kunci: R Studio.

Setelah masuk ke website R Studio, pilih menu *products* -> *Open Source (R Studio)*. Selanjutnya pilih *RStudio Desktop* dan klik pada tombol **Download RStudio Desktop**. Pilih versi *RStudio Desktop free* dan klik *Download*. Selanjutnya akan ada informasi mengenai versi R Studio yang akan diunduh. R Studio versi terbaru per Agustus 2022 adalah *RStudio Desktop 2022.07.1+554*. Versi ini dapat berubah seiring waktu. Selanjutnya klik pada tombol **Download RStudio for Windows**.

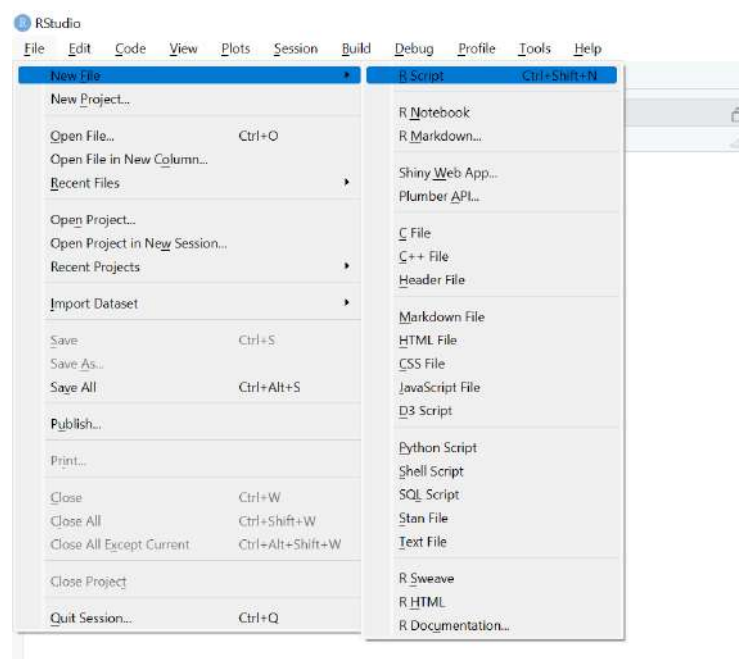
Setelah terunduh, install R Studio tersebut sesuai dengan langkah-langkah instalasi pada jendela yang muncul pada proses instalasi. Apabila sudah sukses terinstall, maka tampilan RStudio adalah sebagai berikut:



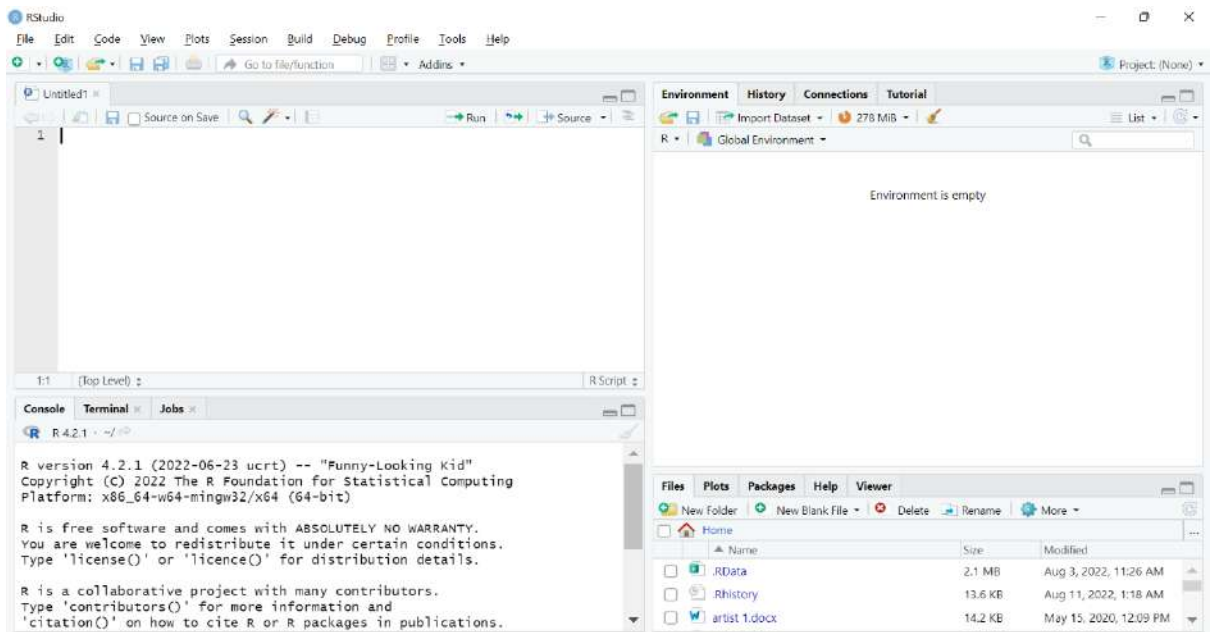
Dengan demikian proses instalasi telah selesai dan kita telah siap dengan RStudio yang akan digunakan untuk keperluan analisis data.

## 2. Lingkungan RStudio

Sebelum kita menggunakan RStudio, kita perlu mengetahui lingkungan pada RStudio yang sudah kita install. Untuk keperluan mengetikkan kode/sintaks kita akan menggunakan *RScript*. Selanjutnya pada RStudio klik File -> New File -> R Script, seperti terlihat pada gambar berikut ini:



Selanjutnya akan muncul tampilan lingkungan RStudio seperti berikut ini:



RStudio tersebut memiliki empat bagian, yaitu *RScript editor* (kiri atas), *Console* (kiri bawah), *Environment* (kanan atas), *Files* (kanan bawah). Secara singkat penjelasan mengenai keempat jendela tersebut yaitu: *RScript editor* digunakan untuk mengetikkan kode, *Console* akan menampilkan *output* dari kode yang kita *run* (jalankan) meskipun bisa digunakan untuk menuliskan kode singkat, *Environment* digunakan untuk menampilkan variabel, nilai variabel, keterangan variabel yang kita gunakan dalam kode; dan *Files* menampilkan file-file yang ada pada direktori yang kita sedang gunakan. Tab *Files* akan berubah menjadi tab yang lain misalnya *Help* jika kita mencari keterangan dari suatu fungsi yang ada di R, atau berubah jadi tab *Plots* untuk menampilkan grafik atau diagram.

Sebelum kita menggunakan *Console* untuk menampilkan *output*, akan lebih baik kita bersihkan terlebih dahulu jendela tersebut dengan cara, mengklik pada jendela *Console* dan tekan tombol **Ctrl + L**.

### 3. Mengimpor data ke R

Untuk keperluan analisis data SWP, data akan kita simpan dalam format *\*.csv* (*comma separated values*). Untuk membuat data dalam format ini, kita bisa menggunakan Microsoft Excel dan data dientry kedalam spreadsheet kemudian disimpan dengan jenis file *\*.csv*.

Selanjutnya untuk analisis, file dalam format csv tersebut akan dibaca kedalam RStudio. Sebagai catatan, data dalam format csv tersebut **harus disimpan dalam folder yang sama** dengan file RScript yang kita gunakan untuk melakukan koding. Hal ini dilakukan untuk menghindari *error* dalam pembacaan data di RStudio.

Untuk membaca data dari file csv kita gunakan kode sebagai berikut:

```
read.csv("nama_file.csv", header = TRUE)
```

dengan `header = TRUE`, jika data csv memiliki header. Jika data csv tidak memiliki header bisa diganti `FALSE`. Selanjutnya data bisa ditampung kedalam suatu variabel untuk memudahkan akses. Misalkan variabel untuk menyimpan datanya kita namakan **data\_skor**, maka pembacaan data bisa kita lengkapi yaitu

```
data_skor <- read.csv("nama_file.csv", header = TRUE)
```

## 4. Fungsi yang diperlukan

Beberapa fungsi bawaan dari R yang diperlukan untuk analisis data SWP diantaranya adalah sebagai berikut:

### 4.1 Fungsi `attach()`

Fungsi `attach()` digunakan untuk mengakses variabel/nama kolom yang ada pada struktur data tanpa harus memanggil nama dari data tersebut.

Sebagai contoh, suatu file data bernama **desa.csv**, yang terdiri atas kolom: nomor, nama\_desa, skor\_desa. Jika data sudah dibaca oleh R dan ditampung kedalam suatu variabel bernama **data\_desa**. Selanjutnya kita bisa menggunakan fungsi `attach()` sbb.:

```
attach(data_desa) .
```

Kemudian untuk melakukan analisis dari setiap kolom pada `data_desa`, kita bisa langsung menggunakan nama kolom tersebut sebagai variabel yaitu: nomor, nama\_desa, dan skor\_desa. Apabila kita tidak menggunakan fungsi `attach()`, maka untuk mengakses kolom tersebut, harus menggunakan format berikut:

```
data_desa$nomor, data_desa$nama_desa, dan  
data_desa$skor_desa
```

Sehingga penggunaan `attach()` lebih meringkas nama variabel.

#### 4.2 Fungsi `return()`

Fungsi `return()` digunakan untuk memunculkan nilai dari kode pada suatu fungsi yang dibuat sendiri (bukan bawaan R). Hal ini karena kita akan membuat fungsi untuk menghitung Indeks Pembangunan Desa (IPD) dan kategorinya agar mempercepat proses komputasi.

Misalkan dalam suatu fungsi yang dibuat kita menggunakan operasi sebagai berikut:

```
a <- b + c
```

maka nilai `a` tersebut bisa ditampilkan dengan cara

```
return(a)
```

#### 4.3 Fungsi `apply()`

Fungsi `apply()` digunakan untuk menerapkan fungsi yang sudah kita buat kepada suatu nilai atau sekumpulan nilai. Fungsi tersebut akan bekerja pada nilai-nilai yang kita terapkan. Struktur fungsi `apply()` secara sederhana adalah sebagai berikut:

```
apply(X, MARGIN, FUN)
```

dengan `X` adalah sebuah array atau matriks, `MARGIN` bisa bernilai 1, 2, atau `c(1,2)`. Nilai 1 menunjukkan baris, nilai 2 menunjukkan kolom, dan `c(1,2)` menunjukkan baris dan kolom. Jika fungsi akan diterapkan per kolom kita gunakan nilai 2.

#### 4.4 Fungsi `data.frame()`

Fungsi `data.frame()` digunakan untuk membentuk suatu struktur data yang terdiri atas baris dan kolom dengan cara menggabungkan beberapa kolom/vector. Kolom yang digabungkan bisa berupa numerik, kategori, atau kombinasi dari keduanya. Sebagai contoh, misalkan kita memiliki dua variabel yaitu `nama_desa` dan `jumlah_kk` yaitu:

```
nama_desa <- c("Desa Cibiru Wetan", "Desa Buah Batu", "Desa Ciapus", "Desa Sukaluyu")
```

```
jumlah_kk <- c(50, 70, 55, 85)
```

maka dengan menggunakan fungsi `data.frame()` kita bisa menggabungkan kedua variabel tersebut, yaitu:

```
data.frame(nama_desa, jumlah_kk)
```

Sehingga akan berbentuk sebagai berikut

	nama_desa	jumlah_kk
1	Desa Cibiru Wetan	50
2	Desa Buah Batu	70
3	Desa Ciapus	55
4	Desa Sukaluyu	85

## 5. Membuat fungsi

Seringkali untuk mempermudah dan menyederhanakan proses komputasi, kita perlu membuat fungsi sendiri sesuai dengan keperluan. Fungsi tersebut sangat berguna jika proses perhitungan dilakukan secara berulang dengan cara yang sama. Secara umum bentuk fungsi adalah sebagai berikut:

```
nama_fungsi <- function (parameter fungsi){  
  Kode/sintaks dari fungsi  
}
```

Sebagai contoh, jika kita ingin membuat fungsi untuk menghitung indeks pembangunan manusia (IPM) dengan rumus

$$IPM = \frac{1}{3} (\text{Indeks } X_1 + \text{Indeks } X_2 + \text{Indeks } X_3)$$

bentuk fungsinya yaitu

```
ipm <- function(X1, X2, X3){  
  (1/3) * (X1 + X2 + X3)  
}
```

Fungsi tersebut bisa kita gunakan dengan cara memasukan nilai indeks X1, X2, dan X3 pada `ipm()`. Misalkan kita memiliki indeks-indeks tersebut yaitu  $X1 = 0.3$ ,  $X2 = 0.4$ , dan  $X3 = 0.6$ . Maka IPM dapat dihitung dengan menggunakan fungsi yaitu `ipm(0.3, 0.4, 0.6)`.

## 6. Kode R untuk menghitung IPD

Setelah kita mengetahui mengenai fungsi-fungsi yang diperlukan serta bagaimana cara membuat fungsi mandiri, maka selanjutnya kita akan implementasikan untuk menghitung indeks pembangunan desa (IPD). Untuk menghitung IPD kita gunakan rumus seperti pada Modul 1 sebelumnya yaitu:

$$IPD = 20 \sum_{i=1}^{42} V_i B_i$$

dengan

$V_i$  = Skor indikator ke-i

$B_i$  = Bobot indikator ke-i.

Untuk keperluan tersebut, kita memerlukan dua buah inputan yaitu  $V_i$  dan  $B_i$ .  $V_i$  merupakan skor indikator ke-i dengan jumlah indikator sebanyak 42 seperti yang sudah dijelaskan pada Modul 1. Sedangkan  $B_i$  merupakan bobot dari setiap indikator yang diambil dari bobot tahun sebelumnya, dalam hal ini bobot pada tahun 2020. Bobot ini juga dapat dilihat pada Modul 1. Untuk bobot indikator karena sudah pasti nilainya bisa kita jadikan sebagai sebuah file csv. Sedangkan untuk skor indikator akan sangat bergantung pada penilaian desa pada tahun tertentu dalam hal ini tahun 2021. Jika data skor indikator sudah diperoleh, data dientry di MS Excel dan juga disimpan dengan format \*.csv.

Data untuk bobot indikator disimpan dengan nama `bobot_indikator.csv` dan skor indikator disimpan dengan nama `skor_indikator.csv`.

Data bobot indikator dalam format csv tampak sebagai berikut dalam MS Excel:

no	nama_indikator	bobot
1	Ketersediaan dan Akses ke TK/RA/BA	0.0227852
2	Ketersediaan dan Akses ke SD Sederajat	0.0057761
3	Ketersediaan dan Akses ke SMP Sederajat	0.0160392
4	Ketersediaan dan Akses ke SMA Sederajat	0.0105802
5	Ketersediaan dan Kemudahan Akses ke Rumah Sakit	0.027163
6	Ketersediaan dan Kemudahan Akses ke Rumah Sakit Bersalin	0.0258106
7	Ketersediaan dan Kemudahan Akses ke Puskesmas	0.0103491
8	Ketersediaan dan Kemudahan Akses ke Poliklinik/Balai Pengobatan	0.0308963
9	Ketersediaan dan Kemudahan Akses ke Tempat Praktek Dokter	0.0325841
10	Ketersediaan dan Kemudahan Akses ke Tempat Praktek Bidan/Rumah Besalin	0.0149669
11	Ketersediaan dan Kemudahan Akses ke Poskesdes atau Polindes	0.0126056
12	Ketersediaan dan Kemudahan Akses ke Apotek	0.0253566
13	Ketersediaan ke Pertokoan, Minimarket, atau Toko Kelontong	0.0065388
14	Ketersediaan ke Pasar	0.0179773
15	Ketersediaan ke Restoran, Rumah Makan, atau Warung/Kedai Makan	0.0076069
16	Ketersediaan ke Akomodasi Hotel atau Penginapan	0.0186228
17	Ketersediaan Bank	0.0229853
18	Elektrifikasi	0.0140417
19	Kondisi Penerangan di Jalan Utama	0.0188277

Sedangkan data untuk skor indikator bernilai 0, 1, 2, 3, 4, dan 5 untuk setiap indikator dengan kriteria seperti yang tercantum pada Modul 2 dan Modul 3. Untuk keperluan ini skor indikator kita generate dari nilai 0 s.d 5 dan diambil contoh tiga desa pertama (skor\_desa1, skor\_desa2 dan skor\_desa\_3) dari total keseluruhan 280 desa. Data skor indikator tampak seperti berikut:

no	nama_indikator	skor_desa1	skor_desa2	skor_desa3	...	skor_desa280
1	1 Ketersediaan dan Akses ke TK/RA/BA	4	4	5		3
2	2 Ketersediaan dan Akses ke SD Sederajat	5	4	2		4
3	3 Ketersediaan dan Akses ke SMP Sederajat	4	5	4		4
4	4 Ketersediaan dan Akses ke SMA Sederajat	3	4	2		4
5	5 Ketersediaan dan Kemudahan Akses ke Rumah Sakit	3	4	5		3
6	6 Ketersediaan dan Kemudahan Akses ke Rumah Sakit Bersalin	5	5	2		5
7	7 Ketersediaan dan Kemudahan Akses ke Puskesmas	3	5	2		3
8	8 Ketersediaan dan Kemudahan Akses ke Poliklinik/Balai Pengobatan	5	5	5		3
9	9 Ketersediaan dan Kemudahan Akses ke Tempat Praktek Dokter	3	4	2		3
10	10 Ketersediaan dan Kemudahan Akses ke Tempat Praktek Bidan/Rumah Besarin	5	4	4		5
11	11 Ketersediaan dan Kemudahan Akses ke Poskesdes atau Polindes	5	4	3		5
12	12 Ketersediaan dan Kemudahan Akses ke Apotek	4	5	2		3
13	13 Ketersediaan ke Pertokoan, Minimarket, atau Toko Kelontong	3	5	4		4
14	14 Ketersediaan ke Pasar	5	4	3		5
15	15 Ketersediaan ke Restoran, Rumah Makan, atau Warung/Kedai Makan	5	5	2		5
16	16 Ketersediaan ke Akomodasi Hotel atau Penginapan	4	4	2		5
17	17 Ketersediaan Bank	3	4	4		4
18	18 Elektrifikasi	3	5	5		5
19	19 Kondisi Penerangan di Jalan Utama	5	5	4		5

Dengan menggunakan R kita perlu membuat fungsi untuk menghitung IPD karena terdapat ulangan sebanyak 280 kali sebanyak jumlah desa. Setelah IPD dihitung, kemudian akan diklasifikasikan kedalam tipologi desa seperti tercantum pada Modul 1 yaitu

- 1) Desa Tertinggal ( $IPD \leq 50,00$ )
- 2) Desa Berkembang ( $50 < IPD \leq 75$ )
- 3) Desa Mandiri ( $IPD > 75$ )

Untuk keperluan pengklasifikasian desa kedalam tipologi juga akan dibuat fungsi tersendiri dalam R. Selanjutnya kita akan menuliskan kode R untuk menghitung IPD dan membuat tipologi desa berdasarkan IPD tersebut menggunakan RScript Editor.

Kode R untuk analisis data SWP untuk contoh tiga buah desa adalah sebagai berikut:

```
# Membaca data dari file csv
bobot_indikator <- read.csv("bobot_indikator.csv", header = TRUE)
skor_indikator <- read.csv("skor_indikator.csv", header =TRUE)

# Membuat supaya nama kolom pada setiap variabel dapat diakses langsung
attach(bobot_indikator)
attach(skor_indikator)

# Fungsi untuk menghitung IPD
ipd_fun <- function(v, b = bobot){
  round(20*t(v)%*%b,2)
}

# Fungsi untuk menentukan tipologi desa
kateg_ipd <- function(ipd = ipd){
  if (ipd > 75){kategori <- "Desa Mandiri"}
  else if(ipd <= 75 && ipd > 50){kategori <- "Desa Berkembang"}
  else {kategori <- "Desa Tertinggal"}
  return(kategori)
}

# Menerapkan fungsi kepada data skor indikator setiap desa
skor_desa <- skor_indikator[,3:dim(skor_indikator)[2]]
IPD <- apply(skor_desa, 2, FUN = ipd_fun)
Kategori <- apply(as.matrix(IPD),1,FUN = kateg_ipd)

# Menampilkan hasil analisis
tabel_hasil <- data.frame(IPD, Kategori)
row.names(tabel_hasil) <- c(paste("Desa", 1:3))
tabel_hasil
```

Output dari kode R tersebut adalah:

```
> tabel_hasil
      IPD      Kategori
Desa 1  69.26  Desa Berkembang
Desa 2  78.31   Desa Mandiri
Desa 3  61.61  Desa Berkembang
```