

SISTEM INFORMASI GEOGRAFIS

TUGAS 01

Tugas ini disusun untuk melengkapi tugas Mata Kuliah Sistem Informasi Geografis



Disusun oleh:

Aziz Abdul Rahman

[L200140029]

PRODI INFORMATIKA
FAKULTAS KOMUNIKASI DAN INFORMATIKA
UNIVERSITAS MUHAMMADIYAH SURAKARTA

2017

Jenis-jenis Tipe Data/ Data Type

1. Tipe data primitive (Sederhana)

Tipe data primitive adalah Tipe data yang mampu menyimpan satu nilai tiap satu variabel. Tipe data primitive merupakan tipe data dasar yang sering dipakai oleh program. Contoh tipe data primitive adalah tipe numerik (integer dan real), tipe data karakter/char, tipe data boolean.

a. Numeric

Tipe data numeric digunakan pada variabel atau konstanta untuk menyimpan nilai dalam bentuk angka. Tipe data ini terbagi atas integer, dan real.

- Integer

Integer Merupakan tipe data berupa bilangan bulat, terbagi atas beberapa kategori seperti table dibawah ini

<i>Tipe Data</i>	<i>Ukuran Tempat</i>	<i>Rentang Nilai</i>
<i>Byte</i>	<i>1 byte</i>	<i>0 s/d +255</i>
<i>Shortint</i>	<i>1 byte</i>	<i>-28 s/d +127</i>
<i>Integer</i>	<i>2 bytes</i>	<i>-32768 s/d 32767</i>
<i>Word</i>	<i>2 bytes</i>	<i>0 s/d 65535</i>
<i>Longint</i>	<i>4 bytes</i>	<i>2147483648 s/d 2147483647</i>

- *Real*

Real adalah bilangan yang berisi titik desimal atau jenis bilangan pecahan.

<i>Tipe Data</i>	<i>Ukuran Tempat</i>	<i>Rentang Nilai</i>
<i>real</i>	<i>6 bytes</i>	<i>2.9×10^{-39} s/d 1.7×10^{38}</i>
<i>single</i>	<i>4 bytes</i>	<i>1.5×10^{45} s/d 3.4×10^{38}</i>
<i>double</i>	<i>8 bytes</i>	<i>5.0×10^{-324} s/d 1.7×10^{308}</i>
<i>extended</i>	<i>10 bytes</i>	<i>3.4×10^{-4932} s/d 1.1×10^{4932}</i>
<i>comp</i>	<i>8 bytes</i>	<i>-9.2×10^{18} s/d 9.2×10^{18}</i>

b. *Karakter (char)*

Karakter merupakan tipe data yang hanya mampu menyimpan 1 digit karakter. Ukuran untuk tipe data karakter adalah 1 byte (1 byte = 8 bit). Adapun macam karakter yang ada sejumlah 256 macam karakter yaitu dari kode karakter (ASCII), 0 sampai dengan 255. Untuk penulisan karakter menggunakan tanda petik tunggal (') di depan dan belakang karakter yang ditulis. Contoh : 'a', 'A', '&' dll.

Nilai-nilai yang termasuk karakter adalah :

✓ *Karakter huruf : 'a'...'z', 'A'...'Z'*

✓ *Karakter angka : '0'...'9'*

- ✓ Karakter tanda baca : titik, koma, titik koma, titik dua dan sebagainya
- ✓ Karakter khusus : \$, %, #, @ dan sebagainya.

c. Boolean

Boolean merupakan tipe data logika, yang berisi dua kemungkinan nilai: TRUE (benar) atau FALSE (salah). Tipe data boolean memakai memori paling kecil.

2. Tipe data Composite

Tipe Data Komposit merupakan tipe data yang dapat menampung banyak nilai, antara lain sebagai berikut.

a. Array

Array atau sering disebut sebagai larik, adalah tipe data yang sudah terstruktur dengan baik, meskipun masih sederhana. Array mampu menyimpan sejumlah data dengan tipe yang sama (homogen) dalam sebuah variabel. Sebagai ilustrasi, array mampu menampung banyak data namun dengan satu tipe data yang sama, misalnya integer saja. Setiap lokasi data array diberi nomor indeks yang berfungsi sebagai alamat dari data tersebut.

b. *Record* atau *struck*

Seperti halnya *Array*, *Record* atau *Struct* juga termasuk tipe data komposit. *Record* dikenal dalam bahasa *Pascal/Delphi* sedangkan *Struct* dikenal dalam bahasa *C++*. Berbeda dengan *array*, tipe data *record* mampu menampung banyak data dengan tipe data berbeda-beda (*heterogen*). Misalnya, satu bagian *integer*, satu bagian lagi *character*, dan bagian lainnya *Boolean*. Biasanya *record* digunakan untuk menampung data suatu obyek. Misalnya, siswa memiliki nama, alamat, usia, tempat lahir, dan tanggal lahir. Nama akan menggunakan tipe data *string*, alamat bertipe data *string*, usia bertipe data *single (numeric)*, tempat lahir bertipe data *string*, dan tanggal lahir bertipe data *date*. Berikut ini contoh penggunaan *record* dalam *Delphi*.

c. *Image*

Image, atau gambar, atau citra, merupakan tipe data grafik. Misalnya grafik lain-lain. Pada bahasa-bahasa pemrograman modern terutama yang berbasis visual, tipe data ini telah didukung dengan sangat baik.

d. *Date Time*

Nilai data untuk tanggal (*date*) dan waktu (*time*) secara internal disimpan dalam format yang spesifik. Variabel atau konstanta yang dideklarasikan dengan tipe data *Date* dapat digunakan untuk menyimpan,

baik tanggal maupun jam. Tipe data ini masuk dalam kelompok tipe data composite, karena merupakan bentukan dari beberapa tipe data.

e. Object

Tipe data object digunakan untuk menyimpan nilai yang berhubungan dengan obyek-obyek yang disediakan oleh Visual Basic, Delphi, dan bahasa pemrograman lain yang berbasis GUI. Sebagai contoh, apabila mempunyai form yang memiliki control Command button, yang kita beri nama Command1.

f. Subrange

Tipe data subrange merupakan tipe data bilangan yang mempunyai jangkauan nilai tertentu sesuai dengan yang ditetapkan programmer. Biasanya, tipe data ini mempunyai nilai batas minimum dan nilai batas maksimum. Tipe data ini didukung dengan sangat baik dalam Delphi.

g. Enumerasi

Tipe data ini merupakan tipe data yang mempunyai elemen-elemen yang harus disebut satu persatu, dan bernilai konstanta integer sesuai dengan urutannya. Nilai konstanta integer elemen ini diwakili oleh suatu nama variable yang ditulis di dalam kurung. Tipe data ini juga dijumpai pada Delphi, dan bahasa pemrograman deklaratif seperti SQL.

Tipe Data Lainnya

1. Tipe Data Terstruktur

a. Tipe Data String

Merupakan suatu data yang menyimpan array (larik), sebagai contoh 'ABCDEF' merupakan sebuah konstanta string yang berisikan 6 byte karakter. Ukuran Tempat untuk tipe data ini adalah 2 s/d 256 byte, dengan jumlah elemen 1 s/d 255.

b. Tipe Data Set

Sebuah set merupakan suatu himpunan yang berisi nilai (anggota). Set merupakan Tipe data yang khusus untuk Pascal. Set dalam pemrograman sangat mirip dengan himpunan dalam ilmu matematika.

2. Tipe Data Pointer

Pointer merupakan variabel khusus yang berisi suatu address (alamat) di lokasi lain didalam memori. Suatu variabel yang points (menunjuk) ke sesuatu sehingga disebut pointer. Ada dua macam pointer:

- a. Typed (tertentu) : merupakan pointer yang menunjuk pada tipe data tertentu pada variabel.*
- b. Generic (umum) : merupakan pointer yang tidak menunjuk pada tipe data tertentu pada variabel.*

Daftar nama fungsi dan kelompoknya

1. Function Declaration

merupakan cara penulisan fungsi yang umumnya kita gunakan pada bahasa pemrograman imperatif dan berorientasi objek.

2. Function Expression

cara pembuatan fungsi yang memperbolehkan kita melewati nama fungsi. Fungsi yang dibuat tanpa nama dikenal dengan sebutan fungsi anonim atau fungsi lambda.

3. Function Parameters

pendefinisian function yang menggunakan parameter. Nilai parameter akan dimasukkan ketika function dipanggil.

4. Function Invocation

Meminjam fungsi sebagai fungsi global, menyebabkan nilai ini menjadi obyek global.

Menggunakan objek jendela sebagai variabel dapat dengan mudah crash program Anda.

5. Function Return

mengembalikan fungsi kapanpun, dan fungsi akan segera berhenti ketika kata kunci return ditemukan.

Bagaimana cara membuat sebuah fungsi dalam javascript

Terdapat empat komponen yang membangun fungsi yang baru kita definisikan di atas, yaitu:

- 1. Kata kunci function, yang memberitahu Javascript bahwa kita akan membuat fungsi.*
- 2. Nama fungsi, dalam contoh di atas adalah tambah. Dengan memberikan sebuah fungsi nama maka kita dapat merujuk ke fungsi tersebut dengan nama yang diberikan. Harus diingat bawa nama fungsi bersifat opsional, yang berarti fungsi pada Javascript tidak harus diberi nama. Kita akan membahas tentang hal ini lebih dalam nanti.*
- 3. Daftar parameter fungsi, yaitu a, b pada contoh di atas. Daftar parameter ini selalu dikelilingi oleh tanda kurung (). Parameter boleh kosong, tetapi tanda kurung wajib tetap dituliskan. Parameter fungsi akan secara otomatis didefinisikan menjadi variabel yang hanya bisa dipakai di dalam fungsi. Variabel pada parameter ini diisi dengan nilai yang dikirimkan kepada fungsi secara otomatis.*
- 4. Sekumpulan perintah yang ada di dalam kurung kurawal {}. Perintah-perintah ini dikenal dengan nama badan fungsi. Badan fungsi dieksekusi secara berurut ketika fungsi dijalankan.*

Contoh Pembuatan Fungsi Javascript

```
function sayHello()  
  
    {  
  
        document.write ("Hello there!");  
  
    }
```

```
function tambah(a, b) {  
  
    hasil = a + b;  
  
    return hasil;  
  
}
```