C programing language pt.6

Arrays:

Collection of elements of the same data type.

Elements are stored at the contiguous memory location.

Easily, you can access the elements with index.

Syntax:

Declaration: `data_type name[size];`

Definition: `data_type name[size]={elements};`

`data_type name[]={elements};`

Access (read or write): `name[index]`

To enter string: `gets(array_name);`

To clear buffer: `fflush(stdin);`

you don't have to initialize all elements.

Size of array is static configuration.

Invalid index returns undefined.

Array name holds the address of array. [pointer to first element]

To do operations in all elements use for loop.

You can define string using character array. "%s"

Add one more element to add null terminator to avoid errors.

You should clear buffer when you scan character specifier.

2-D array:

```
data_type name[row][column];
```

```
data_type name[row][column]={elemnts};
```

```
data_type name[row][column]={{elemnts},{elements},{elments}.........};
```

You can initialize with number of columns only.

3-D array:

The same as 2-D array.

Passing array to function:

```
void function_name(data_type array_name[])
{
    //code
}
```

```
void function_name(data_type array_name[10],data_type length)
{
    //code
}
```

recommended