

Creating Objects

Objectives

- ✉ Identify the built-in data types in C++
- ✉ Use variables in C++ programs
- ✉ Write and execute C++ programs
- ✉ Use arrays in C++ programs

Creating Objects

Variables

- ✉ Are Named locations in memory that contain specific values
- ✉ Rules for naming variables in C++
 - ☰ Should not have any embedded spaces or symbols
 - ☰ Should be unique
 - ☰ Can have any number of characters
 - ☰ Must begin with a letter or an underscore, which may be followed by a sequence of letters, digits, or underscores
 - ☰ Keywords cannot be used as variable names

Creating Objects

Data Types

✉ Define the type of data that can be stored in a variable

✉ Built-in data types are:

- ▣ `char` - For characters and strings

- ▣ `int` - For integers

- ▣ `float` - For numbers with decimals

Creating Objects

Data Types (Contd.)

Data Type	Number of Bytes on a 32-bit Computer	Minimum Value	Maximum Value
<code>char</code>	1	-128	127
<code>int</code>	4	-2^{31}	$(2^{31})-1$
<code>float</code>	4	-10^{39}	10^{39}

Creating Objects

Member Variables

- ✉ Are used to implement attributes in C++
- ✉ Are declared inside the class body
- ✉ Example:

```
class Car
{
    float price;
};
```

Creating Objects

Accepting and Storing Values in Member Variables

✉ The `cin` object is used to accept input from the user

Example:

```
class Car
{
    float price;
public:
    void acceptprice()
    {
        cout << "Enter Price :";cin >> price;
    }
};
```

Creating Objects

Writing and Executing a C++ Program

✉ The `iostream` header file

☰ Is called a pre-processor directive

✉ The `main()` function

☰ Is mandatory in C++ programming for program execution

✉ Creating objects

☰ Is required for reserving memory at the time of declaration

Creating Objects

Compiling, Linking, and Executing a Program

✉ Is done by following the listed steps:

1. The C++ program should contain the `#include` statement, the class declaration and member function definition, and the `main()` function.
2. Save the file with a `.cc` extension.
3. Compile the file using the `g++ <file name>` command from the Linux prompt.
4. Execute the file using the `a.out` command from the Linux prompt.

Creating Objects

Executing C++ Programs

a.out executes the initial
startup code



The startup code executes
the main() function



When the main() function
finishes execution, it sends a
status of execution to the
operating system

Creating Objects

Array

✉ Is a collection of elements of a single data type stored in adjacent memory locations

✉ Syntax:

```
<data_type> <variable_name>[<dimension_size>;
```

✉ Example:

```
int arr[5];
```

Creating Objects

Declaring and Initializing an Array

✉ Example:

```
arr[0] = 14;  
arr[1] = 15;  
arr[2] = 17;  
arr[3] = 45;  
arr[2] = 81;
```

✉ Example:

```
int arr[5] = {14, 15, 17, 45, 81};
```

✉ Example:

```
int arr[] = {14, 15, 17, 45, 81};
```

Creating Objects

Declaring and Initializing an Array (Contd.)

- ✉ Size of an array should be specified at the time of its declaration

Example:

```
char err[]; //ERROR!! will not compile
```

- ✉ An array cannot be initialized with another array

Example:

```
xyz = abc;           // ERROR!!
```

Creating Objects

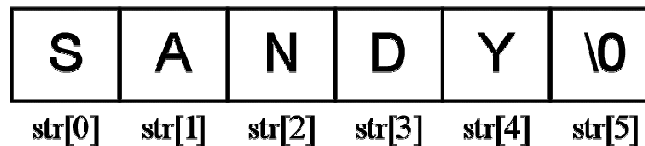
String Constant

✉ Is an array of characters terminated by a
NULL('\0')

Example:

```
char str[] = "SANDY";
```

Can be schematically represented as:



Creating Objects

Problem Statement

As a member of a team that is developing the billing system software for Diaz Telecommunications Inc., you have been assigned the task of creating a software module that accepts the following customer details and displays it.

1. Mobile number, containing a maximum of 12 characters
2. Name, containing a maximum of 25 characters
3. Date of birth, containing a maximum of 10 characters
4. Billing address, containing a maximum of 50 characters

Creating Objects

Problem Statement (Contd.)

5. City, containing a maximum of 25 characters
6. Residence phone number, containing a maximum of 13 characters.
7. Amount outstanding, containing decimal values

Creating Objects

Problem Statement 2.P.1

As a part of the team that is developing the billing system software for Diaz Telecommunications Inc., you have been assigned the task of writing a program that accepts dealer details. The details to be captured are given below:

- ✉ First Name
- ✉ Last Name
- ✉ City
- ✉ Phone number

Creating Objects

Problem Statement 2.P.1 (Contd.)

Write a C++ program to accept and display the dealer details.

The dealer details should be displayed in the following format:

First name:

Last name:

City:

Phone number:

Creating Objects

Summary

In this lesson, you learned that:

- ✉ A variable is a named location in memory that contains a specific value
- ✉ A data type defines the type of data that can be stored in a variable
- ✉ Member variables are declared inside the class body
- ✉ The `cin` object is used to accept input from the keyboard
- ✉ The contents of header files are inserted into a program with the `#include` directive
- ✉ The C++ program execution starts from the first statement of the `main()` function

Creating Objects

Summary (Contd.)

- ✉ Comment entries are notes that a programmer writes anywhere in the code so that any programmer reading the code will understand it better
- ✉ An object is an instance of a class
- ✉ The compiler is a software that translates a program written in a language like C++ into machine language and the file containing the translated program is called the object code of your program
- ✉ *Linking* combines your object code with the object code of the functions you use and adds some standard startup code to produce a run-time version of your program

Creating Objects

Summary (Contd.)

- ✉ The Linux-based GNU compiler for C++ generates the executable code and stores it in a file named `a.out`
- ✉ An *array* is a collection of elements of a single data type stored in adjacent memory locations
- ✉ The array can be initialized when it is defined or later
- ✉ An array must be given a constant dimension size, which should be at least 1
- ✉ Each element of an array can be accessed by its subscript number