"Formal Methods of Describing Syntax"

**ENGLISH**

Alphabets: a, b, c, d, e...z, A, B, C, ... Z

Punctuation: . , ; ! ? " " - /

I eat chocolate

**WORDS**: Combination of Alphabets
Sentence

I eat chocolate. ✓

Eat I. Chocolate ×

He eat chocolate.

He eat chocolate. ×

Here eat should be Eats by Tenses

So

He eats chocolate. ✓
Sentence

- NOUN
  - Pronoun
- Verb
- Subject
  - Word
- Alphabet

---

All these are connected with Grammar which describes the whole English Language Entities

---

Computer Science Lectures By ER. Deepak Garg
Programming Language

1. Alphabets: a, b, c, d, ...... z, A, B, ...... Z

2. DIGITS: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

3. Arithmetics Symbols: * , /, +, -, =

4. Special Symbols: →, ↑, →, $, #, @, !, ☛

Programming Language in More Technical way

→ Headers File: Math.h
→ Statements
→ Expression
→ Identifiers

→ Tokens
→ Lexemes
→ Literals

All needs a grammar which can describe their syntax.

Computer Science Lectures By ER. Deepak Garg
Grammar: Syntax and Structure of a Language.

It is used in Compiler Creation.

Describe the Syntax of a Programming Language.

Natural Language: A Language that has developed naturally through use.

Artificial Language: Language used to communicate with Computers.

Syntax: The rules governing the arrangement of words and phrases to create well-formed sentences.

Meta Language: Language used to model the other language

- Determine whether a series of characters is valid
- Generate well-formed statements
- Break down a statement into constituent parts so it can be converted into a machine language.
Each and every smallest individuals units in a programming language are known as tokens.

C Tokens are of Six Types

1. Keywords  
   (e.g. int, while),
2. Identifiers  
   (e.g. main, total),
3. Constants  
   (e.g. 10, 20),
4. Strings  
   (e.g. "total", "hello"),
5. Special Symbols  
   (e.g. (), {}),
6. Operators  
   (e.g. +, /, -, *)

```c
int main()
{
    int x, y, total;
    x = 10; y = 20;
    total = x + y;
    printf("Total = %d \n", total);
}
```

Where

- main — Identifier
- {, }, (, ) — Delimiters (Special Symbols)
- int — keyword
- x, y, total — identifier
- main, &amp; f, ( ), int, x, y, total — tokens