It has its own syntax
It has its own rules

C++ -> OOP

OOSD -> Object Oriented Software Development
1. OOA -> Object Oriented Analysis
2. OOD -> Object Orineted Design
3. OOP -> Object Oriented Programming

eAttendance

OOA -> Student
Employee
Attendance

OOD ->

```
Student{          Employee{        Attendance{      Date{
int rollno        empid            id               day
string name       name             type             month
mobile            salary           in_time          year
course            dept             out_time         }
email             .                Date date
}                 .                }
                  }
```

OOP -> Object Oriented Programming

CPP
- Bjarne Stroustrup
- Added the concepts of classes in C
- C With Classes
- C++ / CPP

OOP ->
1. Major Pillars
   - Abstarction
   - Encapsulation
   - Modularity
   - Hirerachy
2. Minor Pillars
   - Typing/Polymorphism
   - Concurrency
   - Persistance

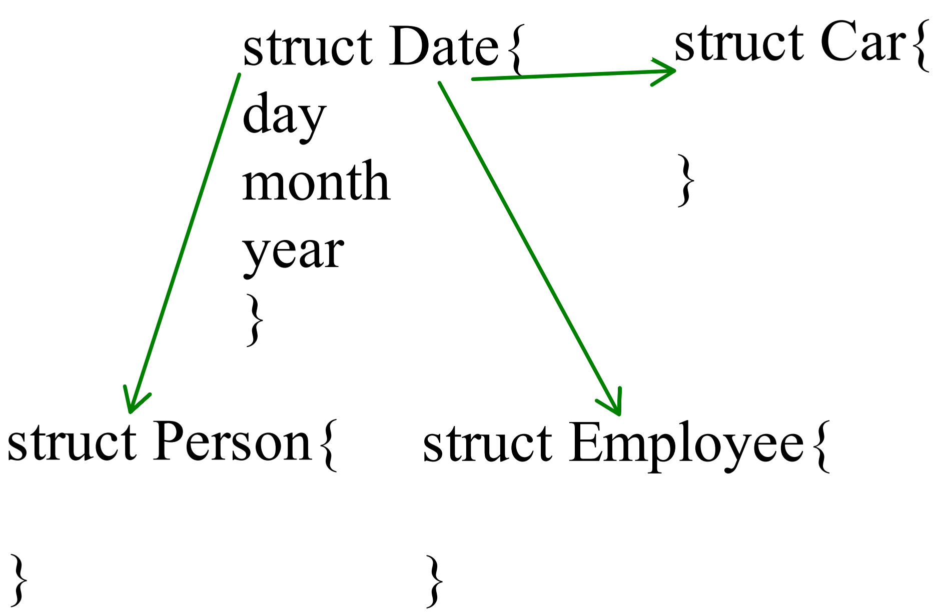
```
printf(){

}
```

Java
C++

Modularity

Encapasulation
- binding the data and code Together

Hirerachy
- Reusability
- has-a (Association)
- is-a (inheritance)



```
struct Date{        struct Car{
day
month               }
year
}

struct Person{      struct Employee{

}                   }
```

Minor Pillars

Typing/Polymorphism

Concurrency

Persistance

```
printf("Hello");
printf("%d",num);
printf("%d%d",num1,num2);
```

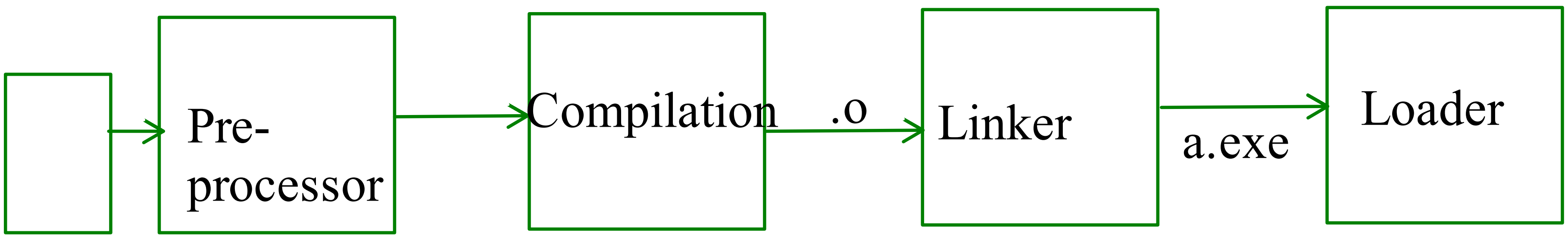
vs-code ?

- CPP
- Html/css
- Javascript
- Express

Editor

IDE

Flow of Execution



Pre-processor

Compilation

.o

Linker

a.exe

Loader

main.cpp
Emp.cpp
Stu.cpp

main.o
emp.o
stu.o

8-10 -> Session -1
10:25 - 12 -> Session-2
12:10 - 1:30 -. Session-3

2:15 - 7 PM -> Lab

60 hrs Theory
60 hrs Lab

DataType

It defines 3 things

1. nature
   - What type of data that can be stored.
2. memory
   - how much memory is required to store that data.
3. operations
   - What all operations we can carry on that stored data

Fundatamental Datatypes

- void, int, char, float, double, bool, wchar_t

Derived Datatypes

- Array, function, Pointer, Structure, Class bool -> boolean

- true
- false
- 1 byte in the memory

A - 65 ->

```
stucture Time{
hr
min
}
 acceptTime();
displayTime();

int main(){
struct Time t1;
}
```

wchar_t

unicode
UTF-8
UTF-16
UTF-32

UTF-16
2 bytes