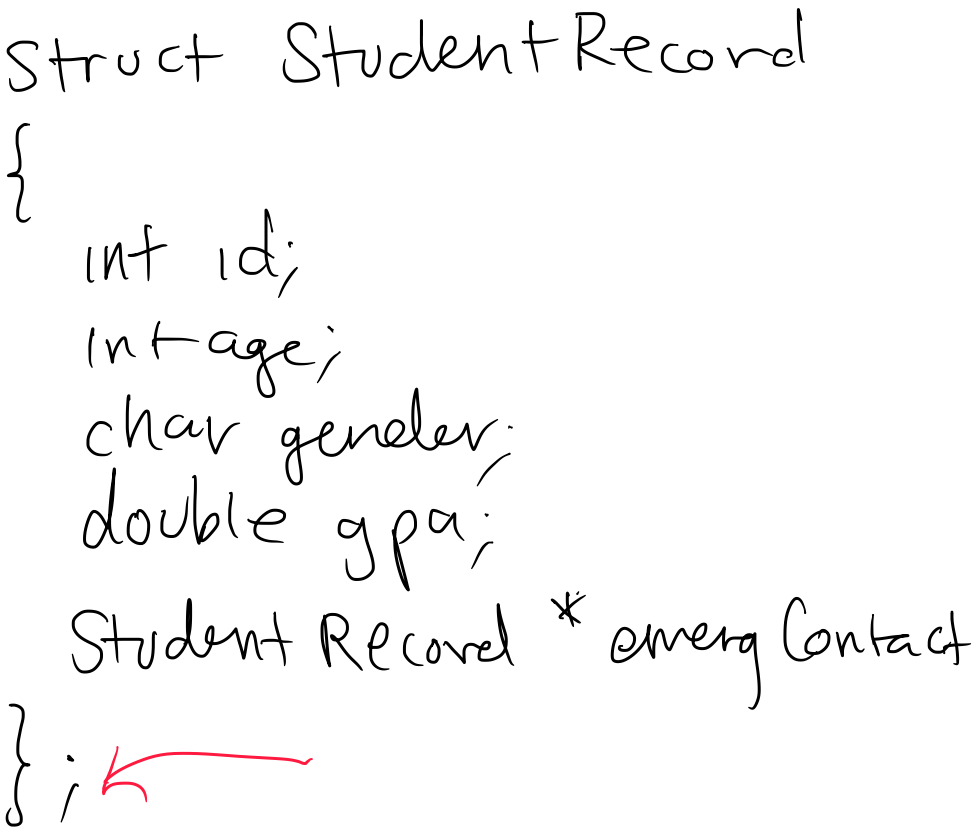
Structs

Structures are used to hold data that **belong** together.

Examples:

- Student record: student ID, name, major, gender, start year
- Address book contact: name, address number

Declaration
Structs are defined outside the main function.



After declaring, variables can be created of the struct type.

- StudentRecord student1 student2;

Member Access

- student1.name

```
strcpy(student1.name, "John Doe");
student1.id = 123;
student1.gender = 'M';

switch(student1.gender) {
  case 'M': cout << "Mr. "; break;
etc etc
```

Can set struct members equal to each other.

Structs are similar to classes, and always public.

Structs Can Have

- Member variables
- Methods (Functions)
- Constructors, Destructors, etc
- public, private, and protected attributes
- virtual functions

Methods / Constructors
No return types.

```
Struct BankAccount {
    int num;
    char name[10];
    BA(int n, char name2[]);
}

BA::BA(int, char [])
{
   num=n;
   strcpy(name, name2);
}
```