constructors - a special type of member function

A constructor is a member function that executes when an object is declared

String lname = new String("Sakas");

Object declaration. A constructor function is called. If the function body were:

```
{
    cout << "What a beautiful day!" ;
}
```

Then *What a beautiful day!* would appear on your monitor screen.

String lname = new String( );

Doesn't look like a function is called, but it is! A **DEFAULT constructor**

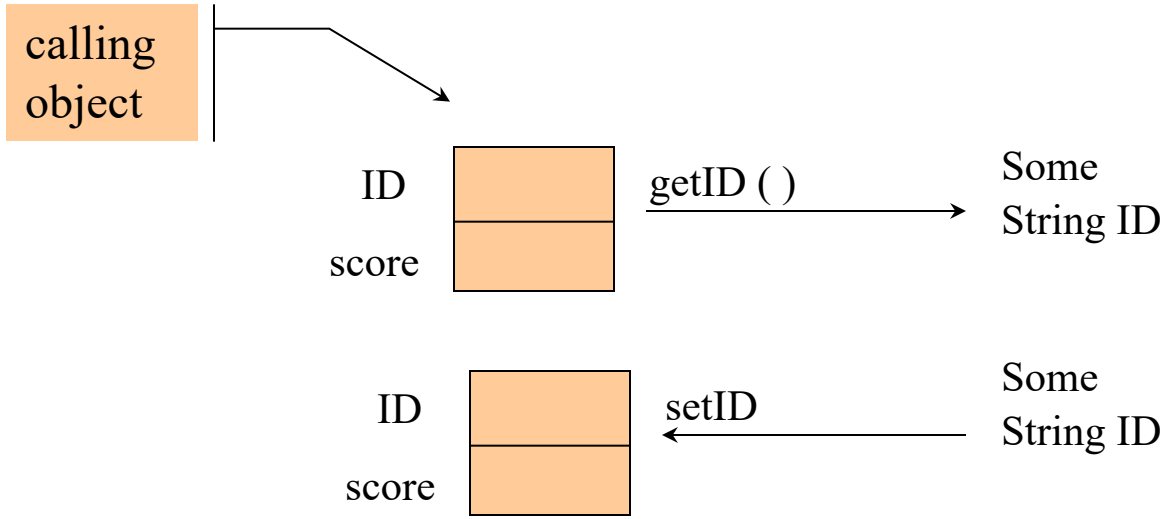
constructors - a special type of member function

A constructor is a member function that executes when an object is declared

```
class  Student
{
     private String ID;
     private int score;

      public Student ( String i, int s) {
      ID = i ;
      score = s ;
   }
}
```

```
String aStudent = new Student("347" , 99);

String anotherStudent = new Student( );
```

Accessor functions



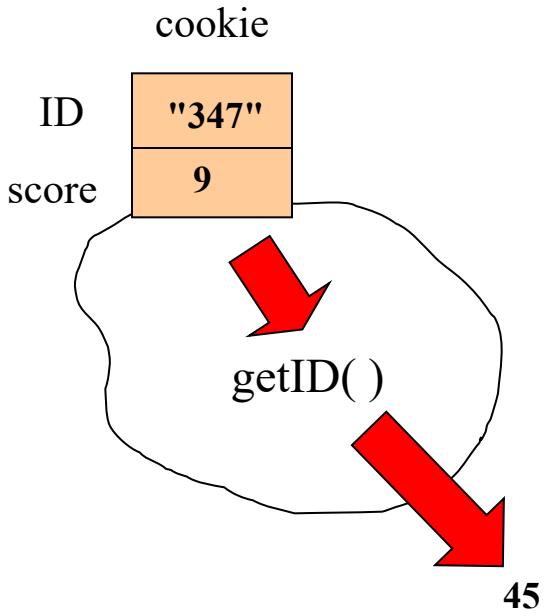

```
class  Student
{
   private String  ID:
   private int score;

   public int  getID(  ) {


    }
    public  void setID ( int s ) {

    }


} ;
```

Accessor functions:
set and *get* are commonly used identifiers.

get - "gets" information from a calling object's member slots. (*Get* values OUT of the member space)

set - "sets" a calling object's slots to some value or values. (*Set* values IN the member space)

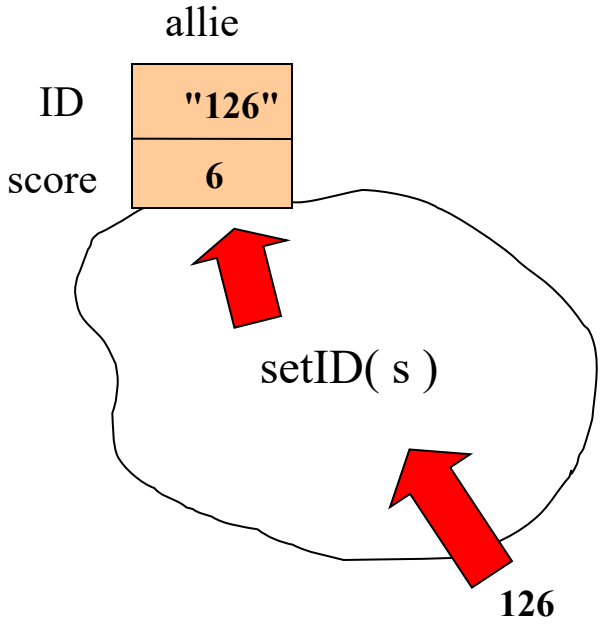
A good picture of what's going on in memory (RAM) when an accessor is executed.



In some block somewhere:
1) ***cookie . getID( )*** is executed.

2) the data members *ID* and *score* are global to the function which means they are accessible inside the function definition block.

3) the function returns the result of the calculation.



In some block somewhere:
1) ***allie . setID (126)*** is executed.

2) the **argument –** 126 is passed to the **parameter placeholder *s.*** – The value is now accessible inside the function definition block.

3) the function converts the value so that the data members: *ID* ***and*** *score* are filled correctly.