# Objects & Classes

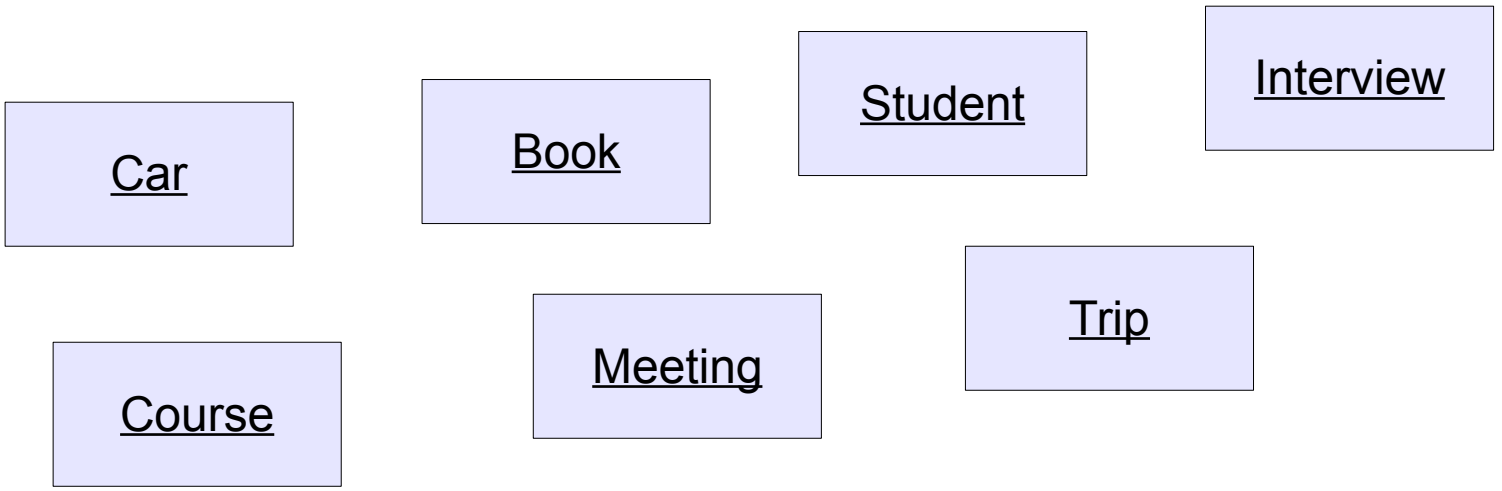
# What is an Object?

- “Object. An abstraction of something in a problem domain, reflecting the capabilities of the system to keep information about it, interact with it, or both.” (Coad and Yourdon, 1990)
- “We define an object as a concept, abstraction, or thing with crisp boundaries and meaning for the problem at hand. Objects serve two purposes: They promote understanding of the real world and provide a practical basis for computer implementation.” (Rumbaugh et al, 1991)

# Our World

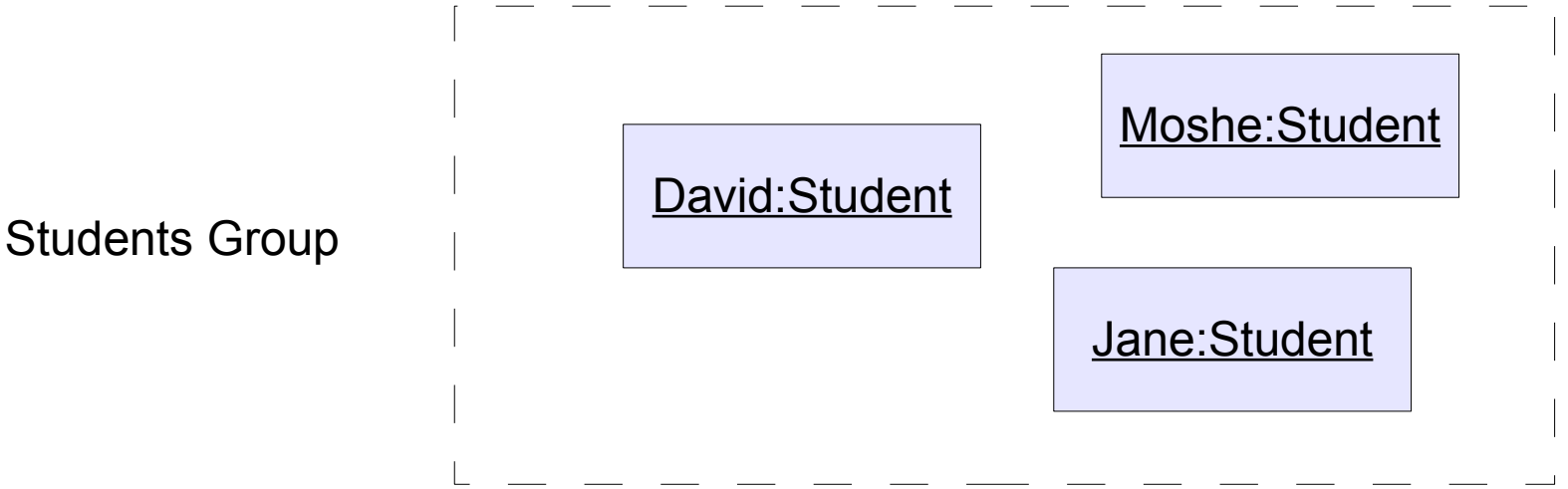
- Looking around us, everything we see, feel or even think of can be represented as an object.

  Objects can represent anything we can think of. Whether it is something concrete or abstract.

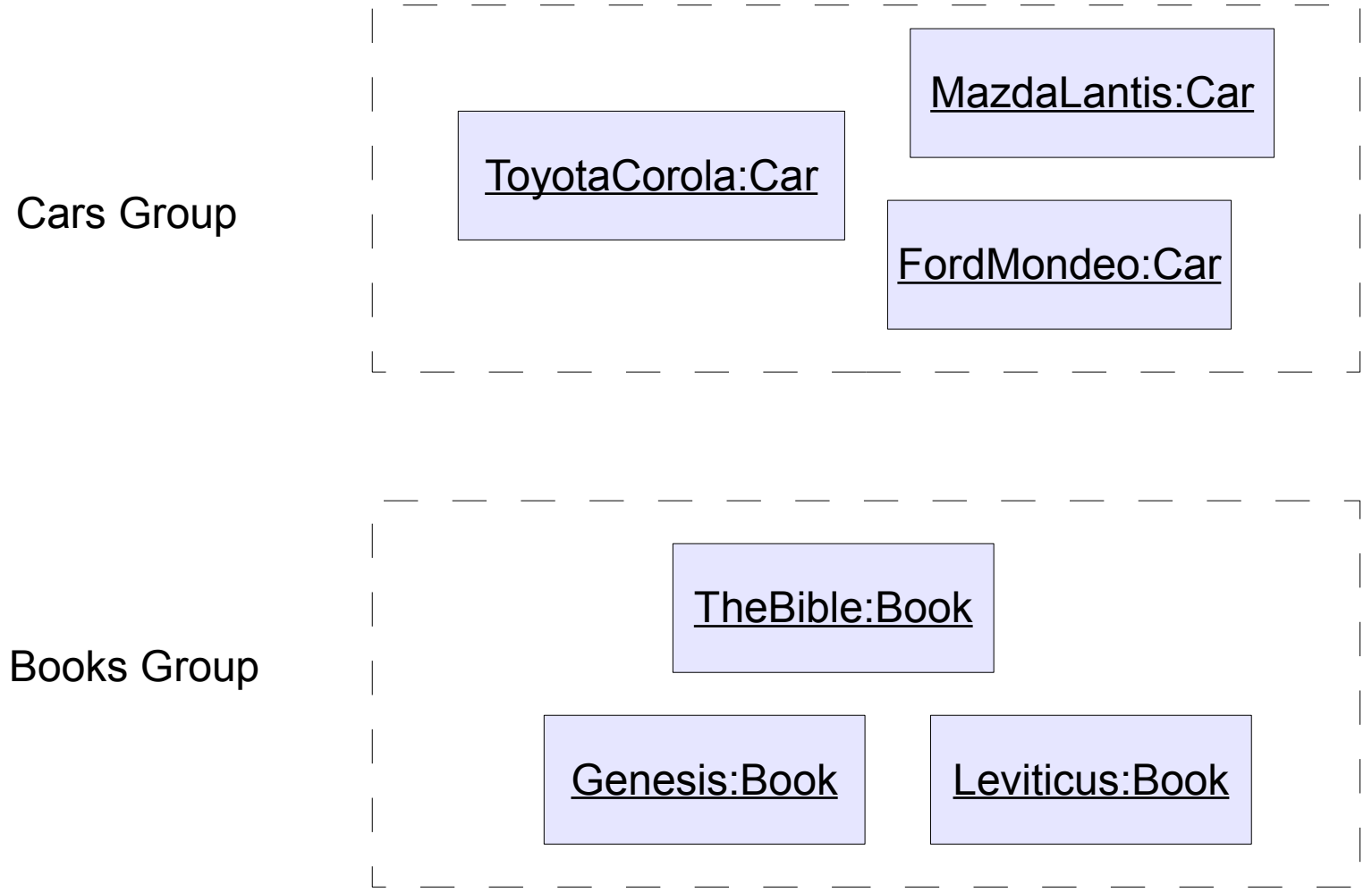


# Our World

- Thinking about the objects that represent everything we can think of we can put them into groups (classes) we can create based on the attributes and the operations the objects share.



# Our World



# What is a Class?

- “A class is the general template we use to define and create specific objects. Every object is associated with a class. An object is instantiation of a class.” (Alan Dennis, 2006)
- “Class is a concept that describes a set of objects that are specified in the same way. All objects of a given class share a common specification for their features, their semantics and the constraints upon them.” (OMG, 2004)

# What is a Class?

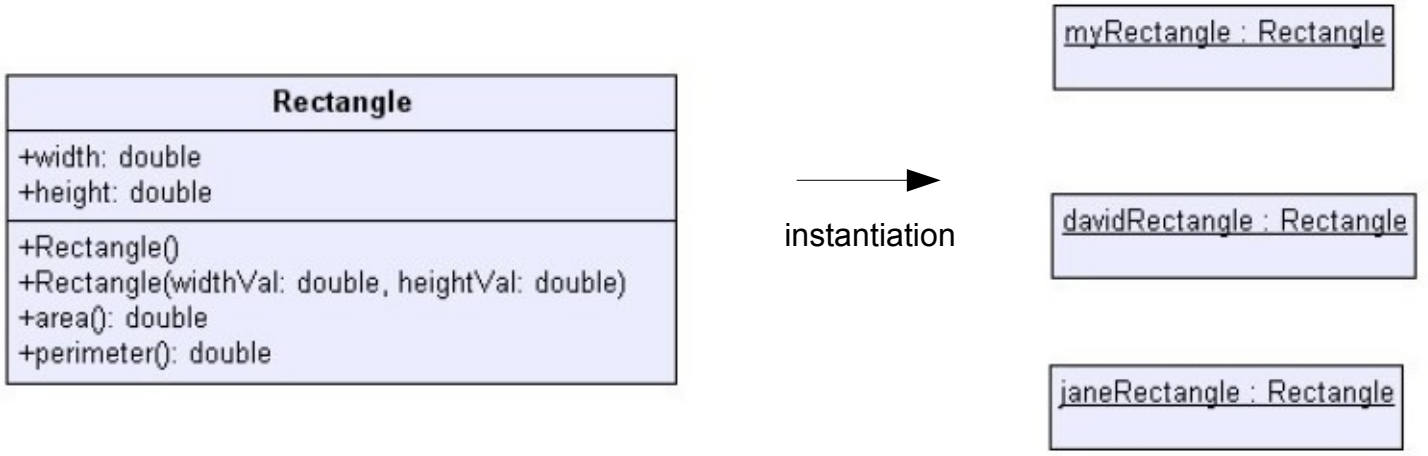
- Each attribute that was defined in our class exists in each one of the objects that were instantiated from it. Each object has the same attributes with possible different values.

  The information can be anything we can think of, such as a student name, birthday, address and phone number. The attributes

- Each operation that was defined in our class can be called on each one of the objects that were instantiated from it.

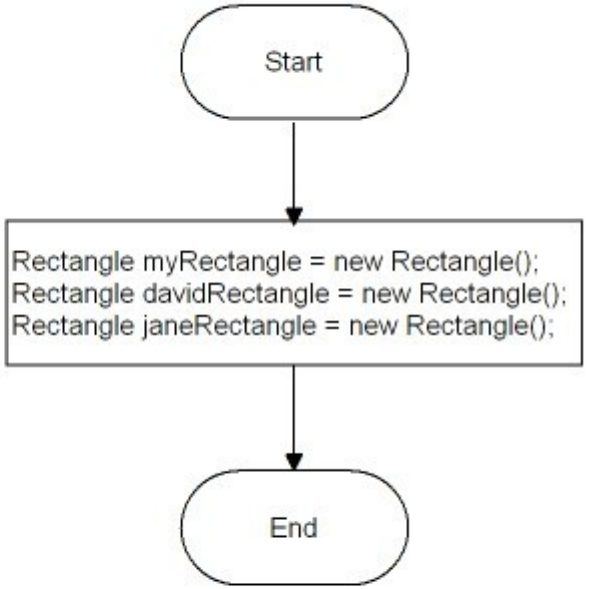
  When calling a method on a specific object the method uses the values that object holds in its attributes.

# Rectangle Class Instantiation Sample



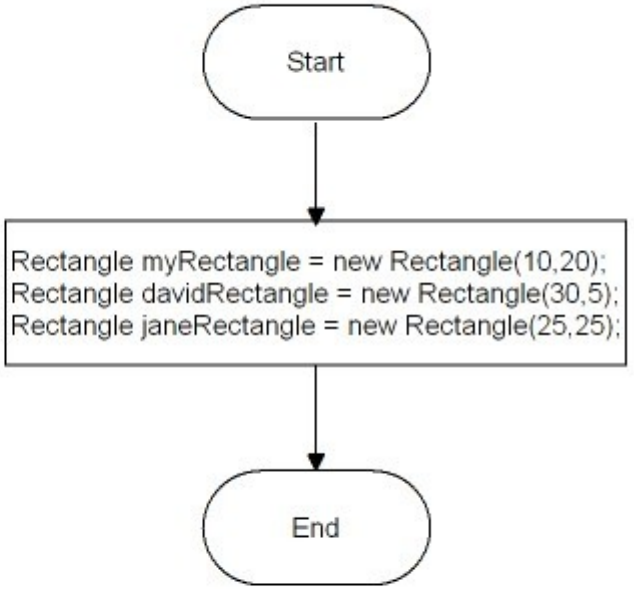
# Rectangle Class Instantiation Sample

```
class Rectangle
{
   double width;
   douoble height;
   Rectangle()  { }
   Rectangle(widthVal double, heightVal double)
   {
      width = widthVal;
      height = heightVal;
   }
   double area()
   {
      return width * height;
   }
   double perimeter()
   {
      return 2*(width+height);
   }
}
```

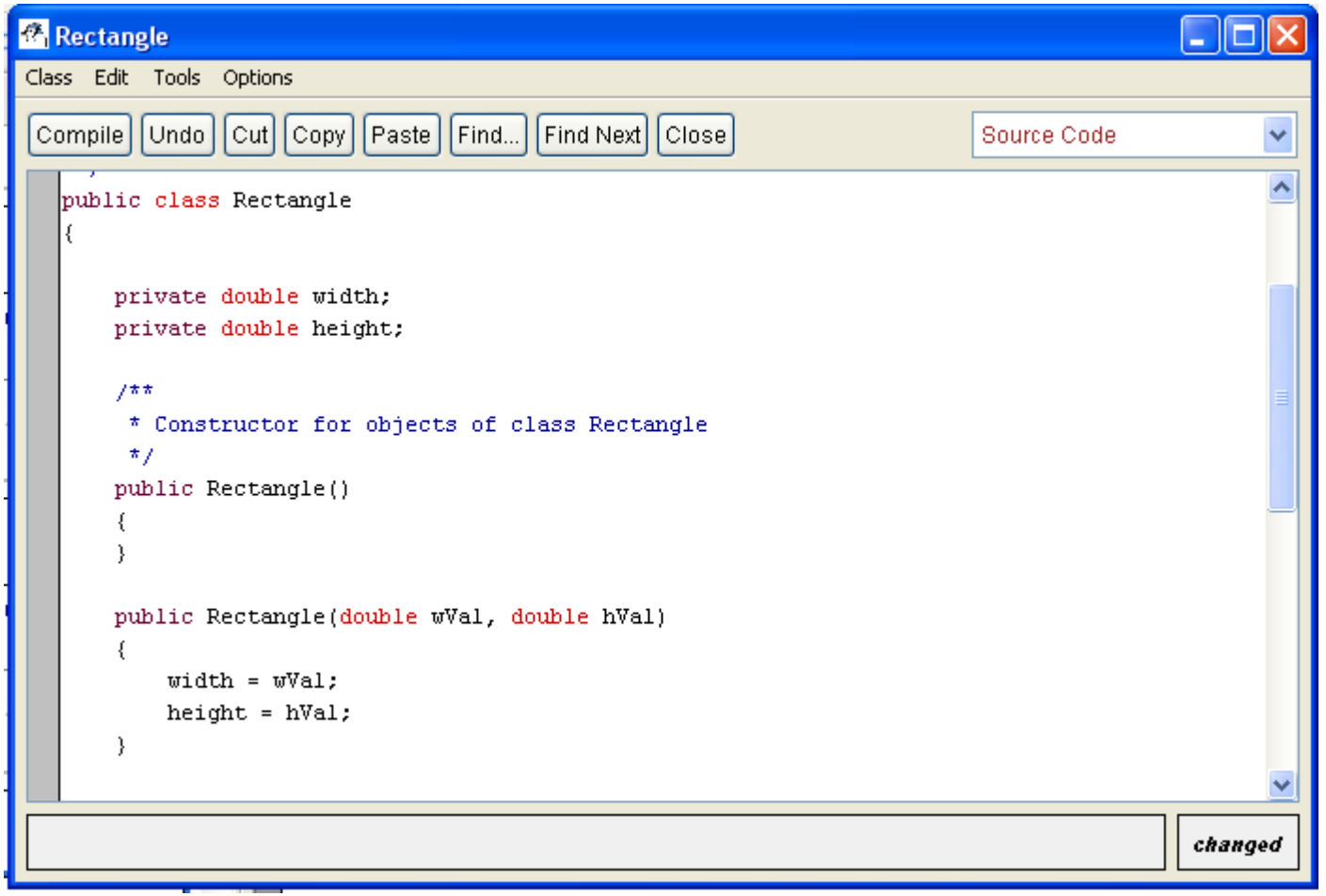


# Rectangle Class Instantiation Sample

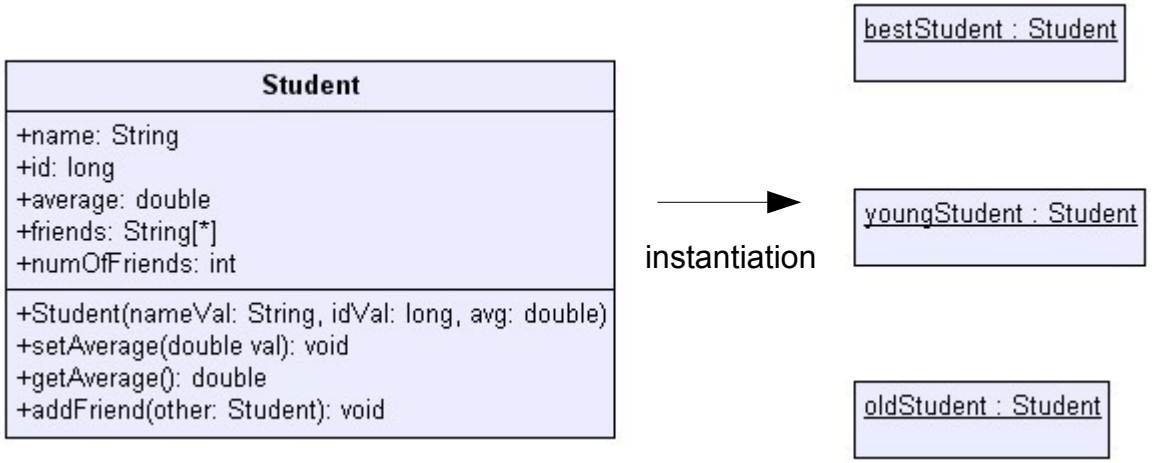
```
class Rectangle
{
   double width;
   douoble height;
   Rectangle()  { }
   Rectangle(widthVal double, heightVal double)
   {
      width = widthVal;
      height = heightVal;
   }
   double area()
   {
      return width * height;
   }
   double perimeter()
   {
      return 2*(width+height);
   }
}
```



# Rectangle Class Instantiation Sample

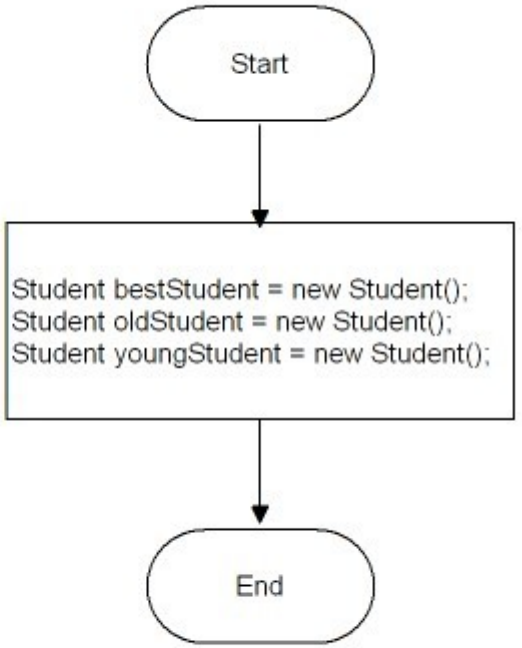


# Student Class Instantiation Sample

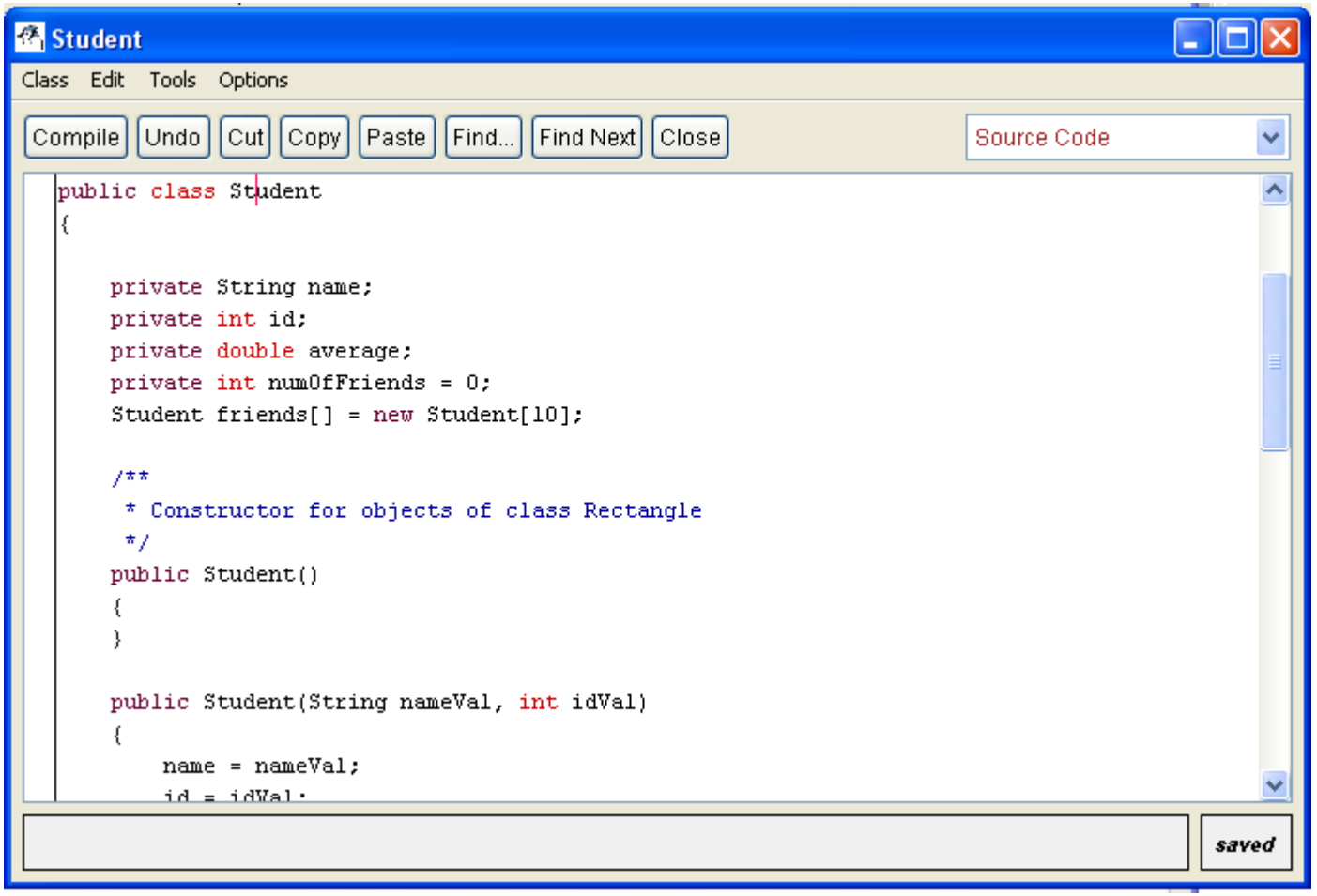


# Student Class Instantiation Sample

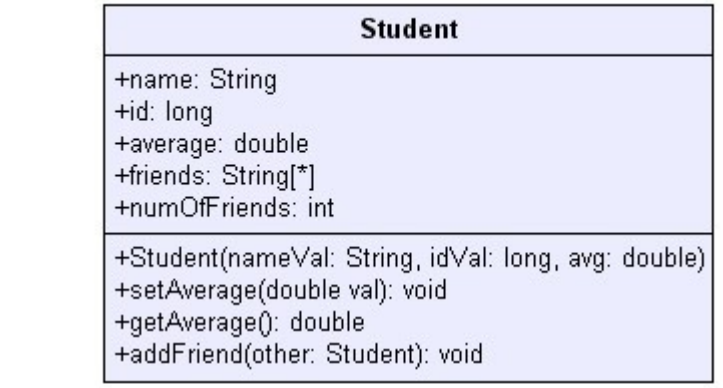
```
class Student
{
  String name;
  int id;
  double average;
  Stdudent friends[] = new Student[10];
  int numOfFriends = 0;
  Student(String str, int value)
  {
    name = str;
    id = value;
  }
  void addFriend(Student friend)
  {
    numOfFriends = numOfFriends + 1;
    friends[numOfFriends] = friend;
  }
  double getAverage()
  {
    return average;
  }
  double setAverage(double avg)
  {
    average = avg;
  }
}
```

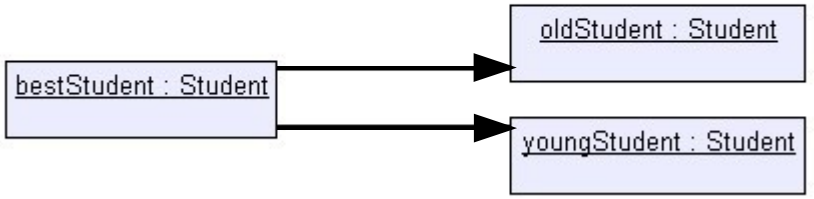


# Student Class Instantiation Sample



# Students Objects Association Sample



# Students Objects Association Sample

```
class Student
{
   String name;
   int id;
   double average;
   Stdudent friends[] = new Student[10];
   int numOfFriends = 0;
   Student(String str, int value)
   {
      name = str;
      id = value;
   }
   void addFriend(Student friend)
   {
       numOfFriends = numOfFriends + 1;
       friends[numOfFriends] = friend;
   }
   double getAverage()
   {
       return average;
   }
   double setAverage(double avg)
   {
       average = avg;
   }
}
```