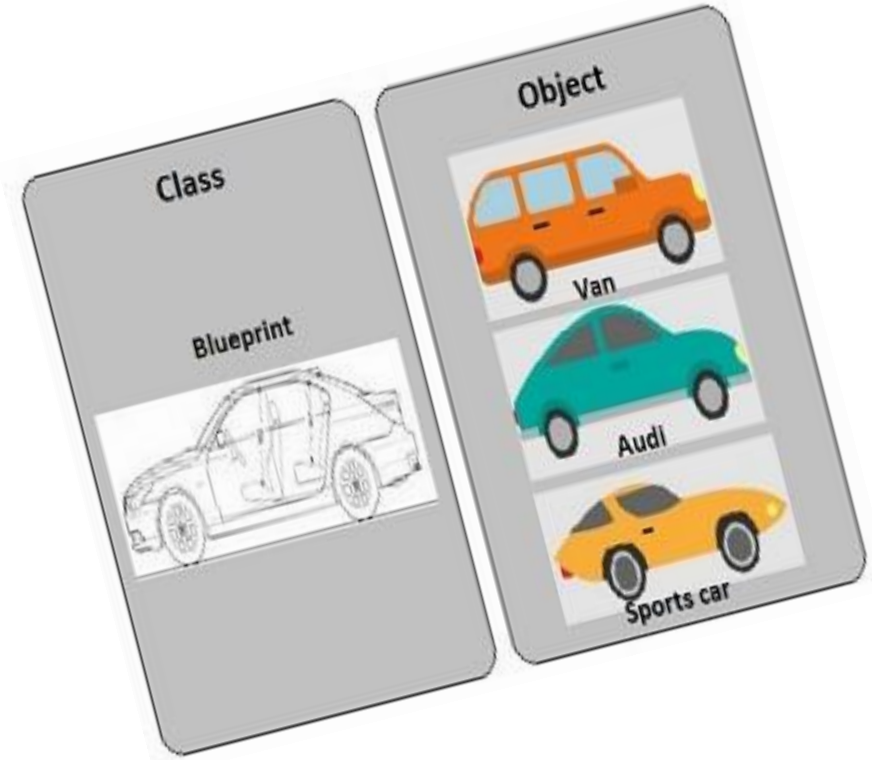


Writing our own classes to build *custom* data types

Computer Science OOD
Boston University

Christine Papadakis-Kanaris

Recall: Classes As Blueprints

- A *class* is a blueprint – a *custom* definition of a data type.
  - specifies the data values and methods of that type
- Objects are built according to the blueprint provided by their class.
  - they are "values" / *instances* of that type
  - they are the *physical* entities created from the class blueprint.

Why the need for classes?

- Let's say I want to keep track of data on my students:
  - **Name {first, last, middle}**
  - **Date of birth {month, day, year}**
  - **Student ID**

- Let's assume all data is stored as strings.

- How can I maintain all this data for each student?

- We wouldn't create a variable for each piece of data multiplied by all students?

- We could use arrays. So let's start there!

Why the need for classes?

```
String [] studentName = new String[N];
String [] dob = new String[N];
String [] sid = new String[N];
```

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for classes?

```
String [] studentName = ne
String [] dob = new Strin
String [] sid = new Str
```

Note each element is really a reference to a string!

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

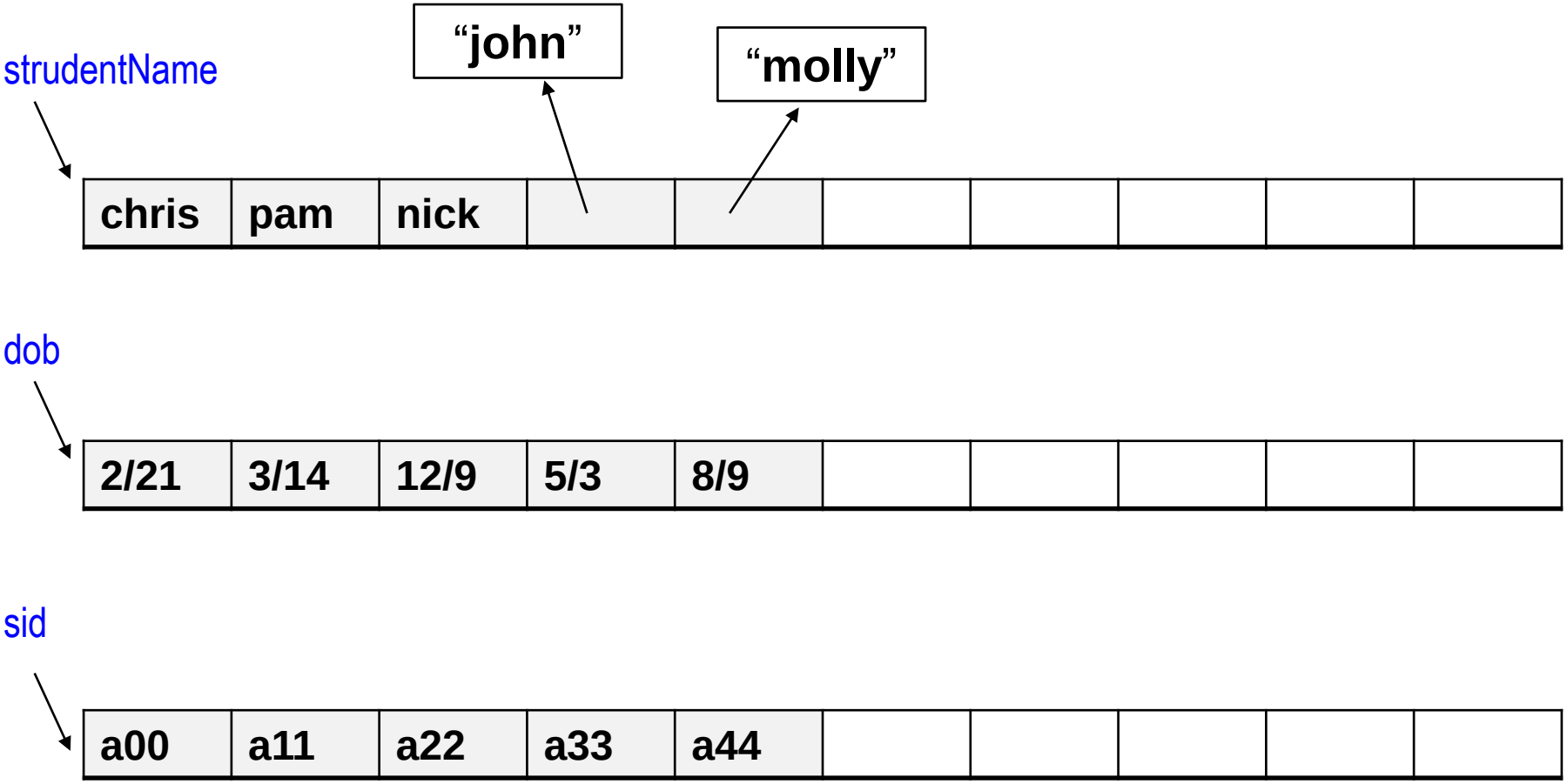
2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

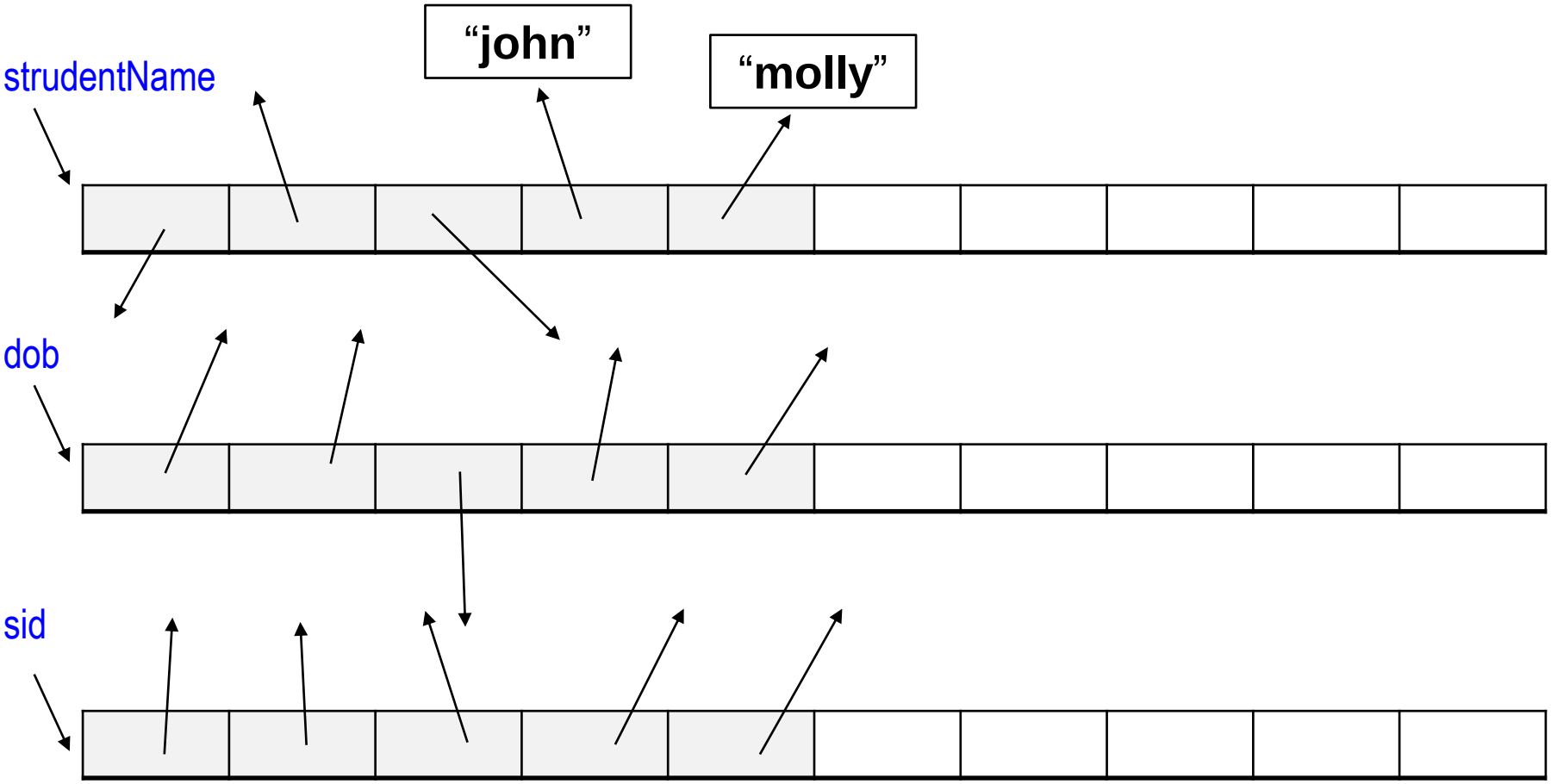
Why the need for classes?

```
String [] studentName = new String[N];
String [] dob = new String[N];
String [] sid = new String[N];
```



Why the need for classes?

```
String [] studentName = new String[N];
String [] dob = new String[N];
String [] sid = new String[N];
```



Why the need for classes?

```
String [] studentName = ne
String [] dob = new Strin
String [] sid = new Str
```

… for simplicity the strings are shown within the box and not as references…

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for classes?

```
String [] studentName = ne
String [] dob = new Strin
String [] sid = new Str
```

What are the implications of making all data items of type String, specifically date of birth?

strudentName

chris	**pam**	**nick**	**john**	**moll**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for classes?

```
String [] studentName = ne
String [] dob = new Strin
String [] sid = new Str
```

Cannot perform any mathematical computations (i.e. age)!

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for classes?

```
String [] studentName = new String[N];
int [] month = new int[N];
int [] day = new int[N];
int [] year = new int[N];
```

… three *integer* arrays to properly represent the date of birth!

strudentName

chris	**pam**	**nick**	**john**	**molly**					

month

2	**3**	**12**	**5**	**8**					

day

21	**14**	**9**	**3**	**9**					

year

1989	**2001**	**1996**	**2012**	**1999**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

… for simplicity…

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

Recall the physical memory layout of an array is …

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

contiguous!

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

Implies the physical structure of our data is horizontal.

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

How do we want to process or view this data?

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for class?

```
String [] studentName = ne
String [] dob = new String[
String [] sid = new String[
```

By a specific array
or by the
logical definition of
a student?

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for class

```
String [] studentName = ne
String [] dob = new String[
String [] sid = new String[
```

By a specific array
or by the
logical definition of
a student?

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for class

```
String [] studentName = new String
String [] dob = new String[N];
String [] sid = new String[N];
```

We would usually be interested in all the data for a specific student

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for class

```
String [] studentName = new String
String [] dob = new String[N];
String [] sid = new String[N];
```

Therefore we rely on the index:
studentName[**4**]
dob[**4**]
sid[**4**]

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for clas

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

But what if I wanted to add a new student and maintain the sid order?

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

Say a34?

strudentName

chris	**pam**	**nick**	**john**	**molly**					

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

…shift down to open the spot to add the new student…

strudentName

chris	**pam**	**nick**	**john**		**molly**				

dob

2/21	**3/14**	**12/9**	**5/3**	**8/9**					

sid

a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

and again…

strudentName

dob

sid

chris	**pam**	**nick**	**john**		**molly**				
2/21	**3/14**	**12/9**	**5/3**		**8/9**				
a00	**a11**	**a22**	**a33**	**a44**					

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

and again…

strudentName

dob

sid

chris	**pam**	**nick**	**john**		**molly**				
2/21	**3/14**	**12/9**	**5/3**		**8/9**				
a00	**a11**	**a22**	**a33**		**a44**				

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

…and then insert the data for the new student!

strudentName

chris	**pam**	**nick**	**john**	**joy**	**molly**				

dob

2/21	**3/14**	**12/9**	**5/3**	**9/1**	**8/9**				

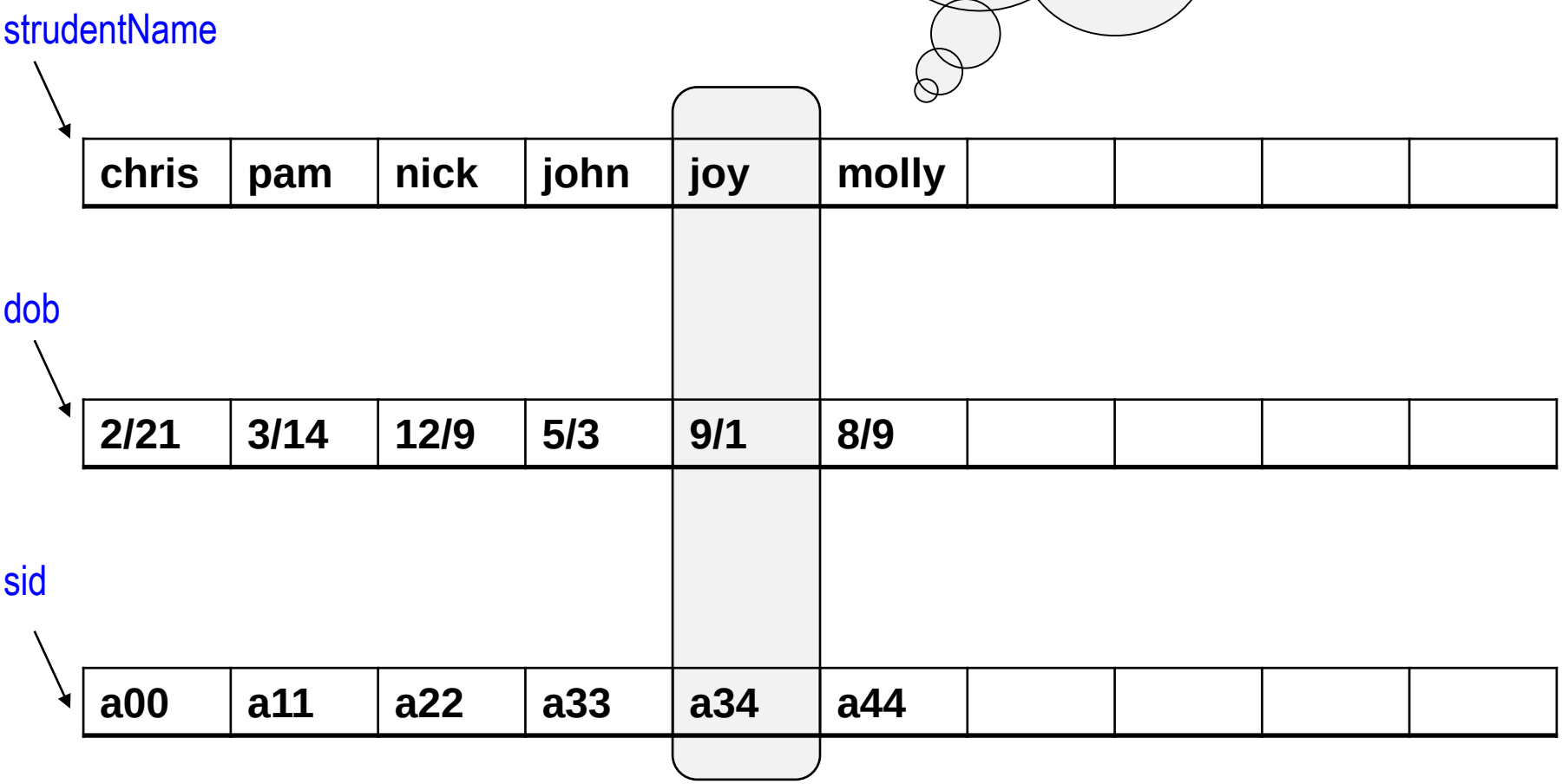
sid

a00	**a11**	**a22**	**a33**	**a34**	**a44**				

Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

Consider the *inefficiency* of the memory use in this scenario?



Why the need for cla

```
String [] studentName = ne
String [] dob = new String
String [] sid = new String
```

The problem is that our logical view of the data…

strudentName

chris	**pam**	**nick**	**john**	**joy**	**molly**				

dob

2/21	**3/14**	**12/9**	**5/3**	**9/1**	**8/9**				

sid

a00	**a11**	**a22**	**a33**	**a34**	**a44**				

Why the need for class?

```
String [] studentName = ne
String [] dob = new String[
String [] sid = new String[
```

does not match the physical memory *layout* of our data!

strudentName

chris	**pam**	**nick**	**john**	**joy**	**molly**				

dob

2/21	**3/14**	**12/9**	**5/3**	**9/1**	**8/9**				

sid

a00	**a11**	**a22**	**a33**	**a34**	**a44**				

Why the need for class?

```
String [] studentName = ne
String [] dob = new String[
String [] sid = new String[
```

We need a physical representation that matches our logical view!

strudentName

chris	**pam**	**nick**	**john**	**joy**	**molly**				

dob

2/21	**3/14**	**12/9**	**5/3**	**9/1**	**8/9**				

sid

a00	**a11**	**a22**	**a33**	**a34**	**a44**				

Why the need for class ...?

```
String [] studentName = ne
String [] dob = new String[
String [] sid = new String[
```

This is what *class definitions* allow us to do!

strudentName

chris	**pam**	**nick**	**john**	**joy**	**molly**				

dob

2/21	**3/14**	**12/9**	**5/3**	**9/1**	**8/9**				

sid

a00	**a11**	**a22**	**a33**	**a34**	**a44**				

Why the need for classes?

```
class Student {
   String name;
   String dob;
   String sid;
}
```

Student is a custom datatype that we define and can use to create an ***instance*** of the Student class: the physical object created from this class definition.

Why the need for classes?

```
class Student {
    String name;
    String dob;
    String sid;
}
```

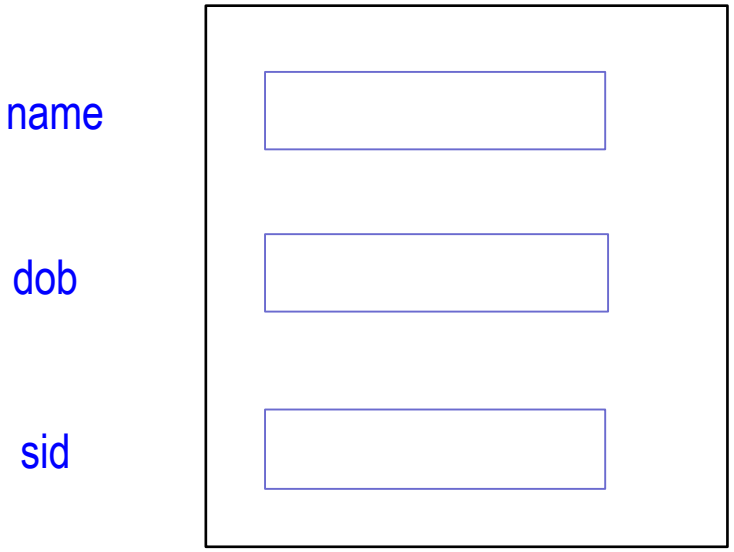
These are the fields or attributes of this class!

Why the need for classes?

```
class Student {
   String name;
   String dob;
   String sid;
}
```

Create an instance of the Student class:

```
Student s = new Student();
```

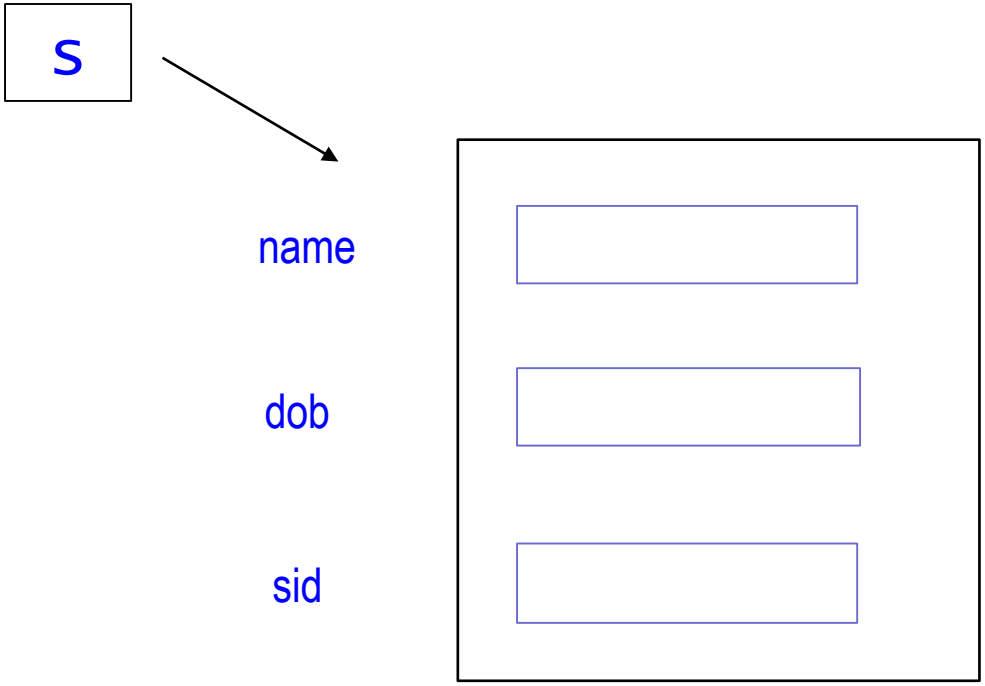


Why the need for classes?

```
class Student {
   String name;
   String dob;
   String sid;
}
```

Create an instance of the Student class:

```
Student s = new Student();
```

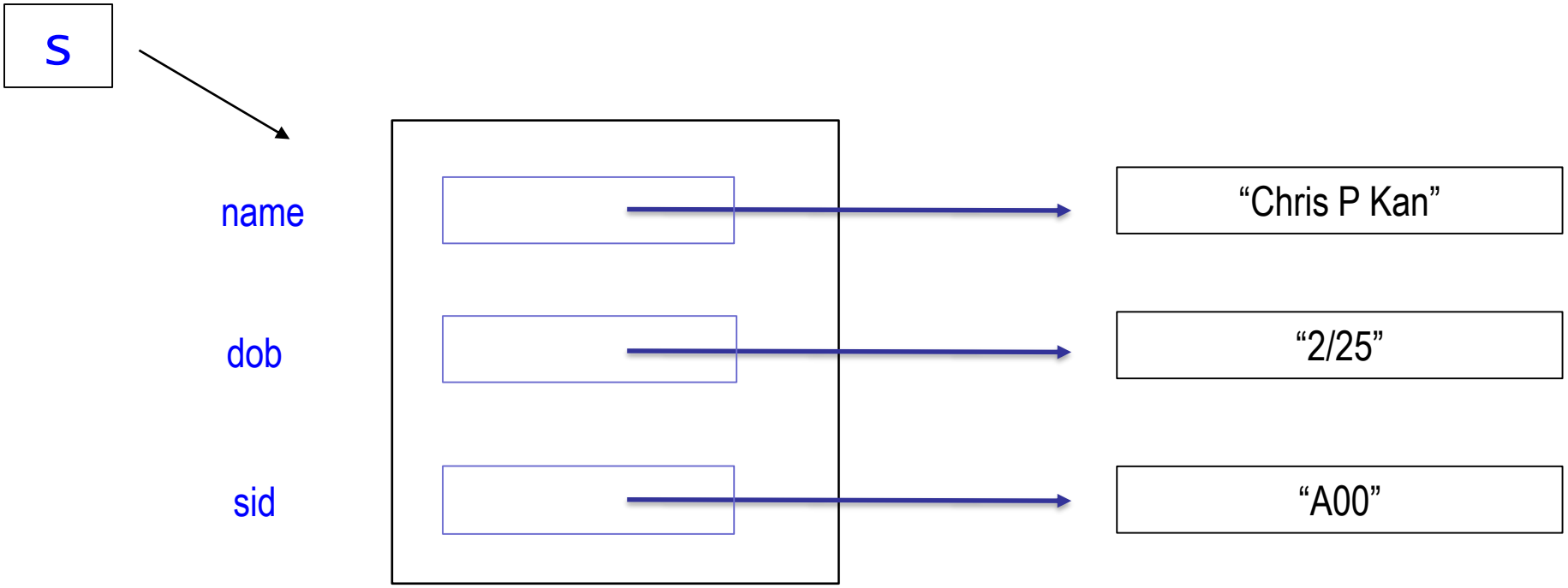


Why the need for classes?

```
class Student {
   String name;
   String dob;
   String sid;
}
```

Create an instance of the Student class:

```
Student s = new Student("Chris P Kan",
"2/25", "A00");
```

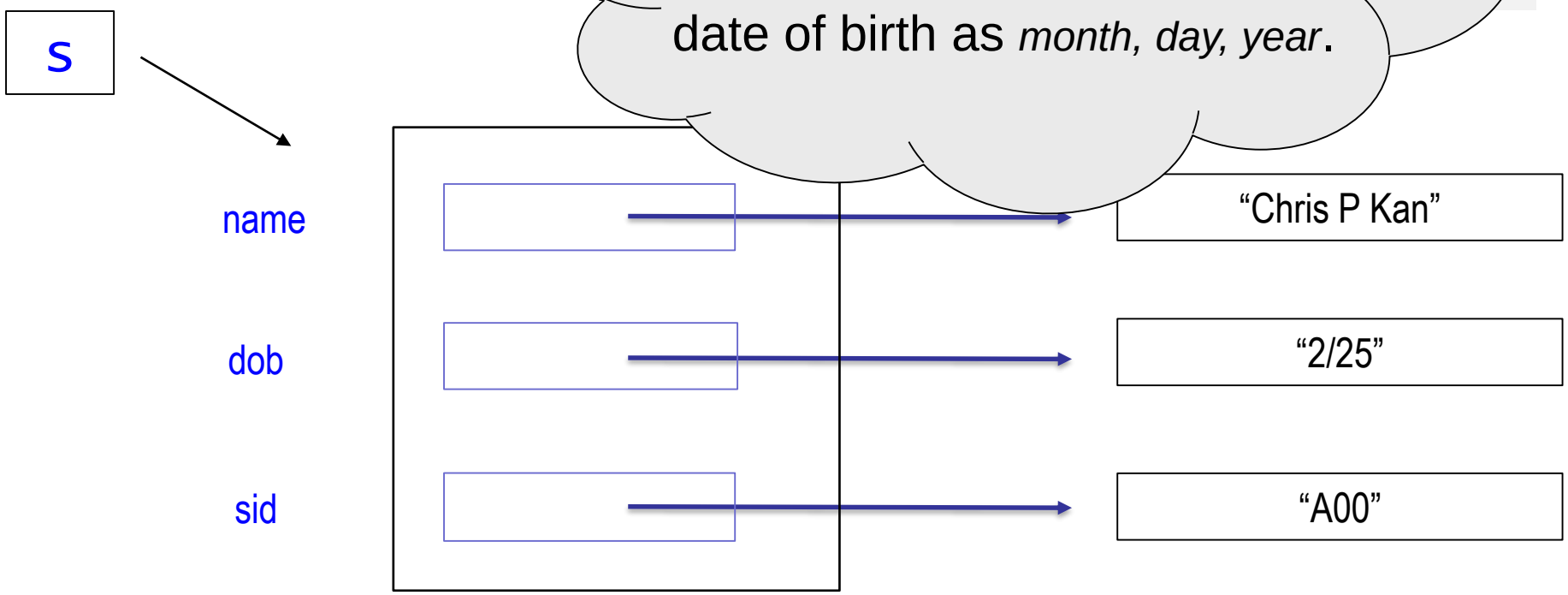


Why the need for classes?

```
class Student {
    String name;
    String dob;
    String sid;
}
```

Create an instance of the Student class!

We can even define
new classes,
that logically represent a
person's name as *first, middle, last* and
date of birth as *month, day, year*.

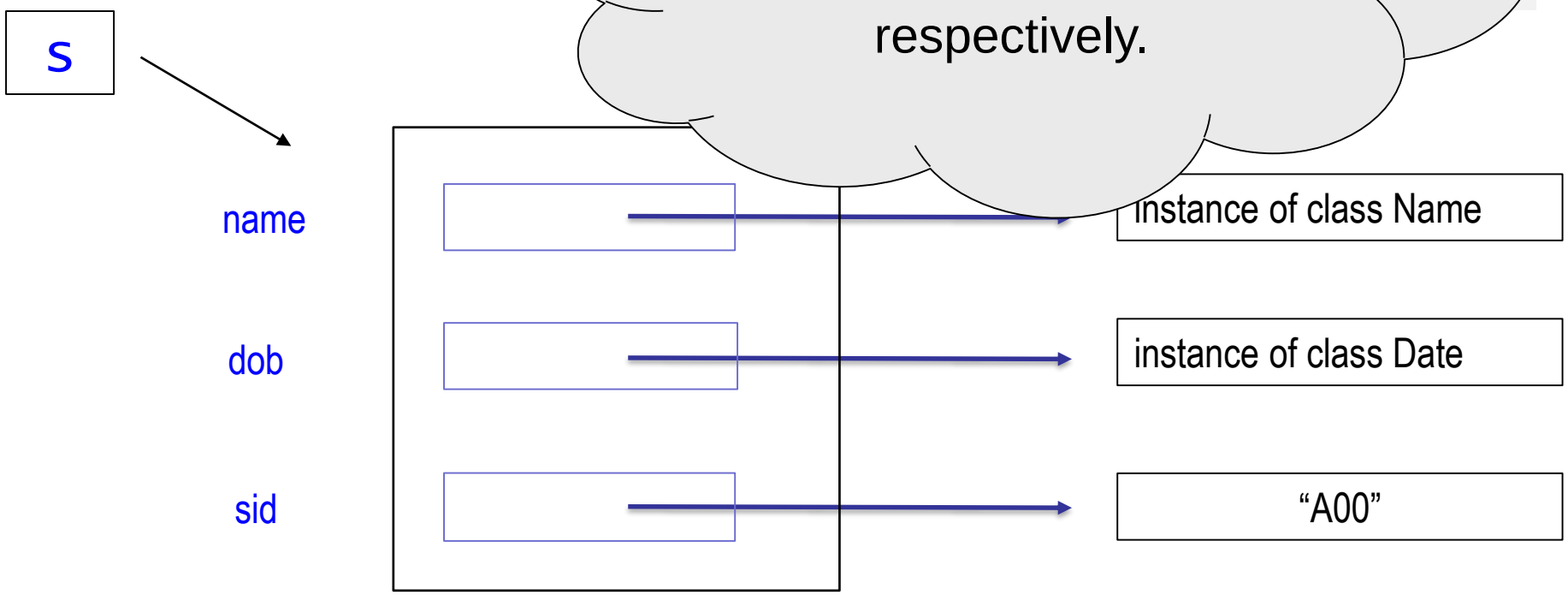


Why the need for classes?

```
class Student {
    Name name;
    Date dob;
    String sid;
}
```

Create an instance of the Student class!

The data members: *name* and *dob* are *references* to instances of the Name and Date class respectively.

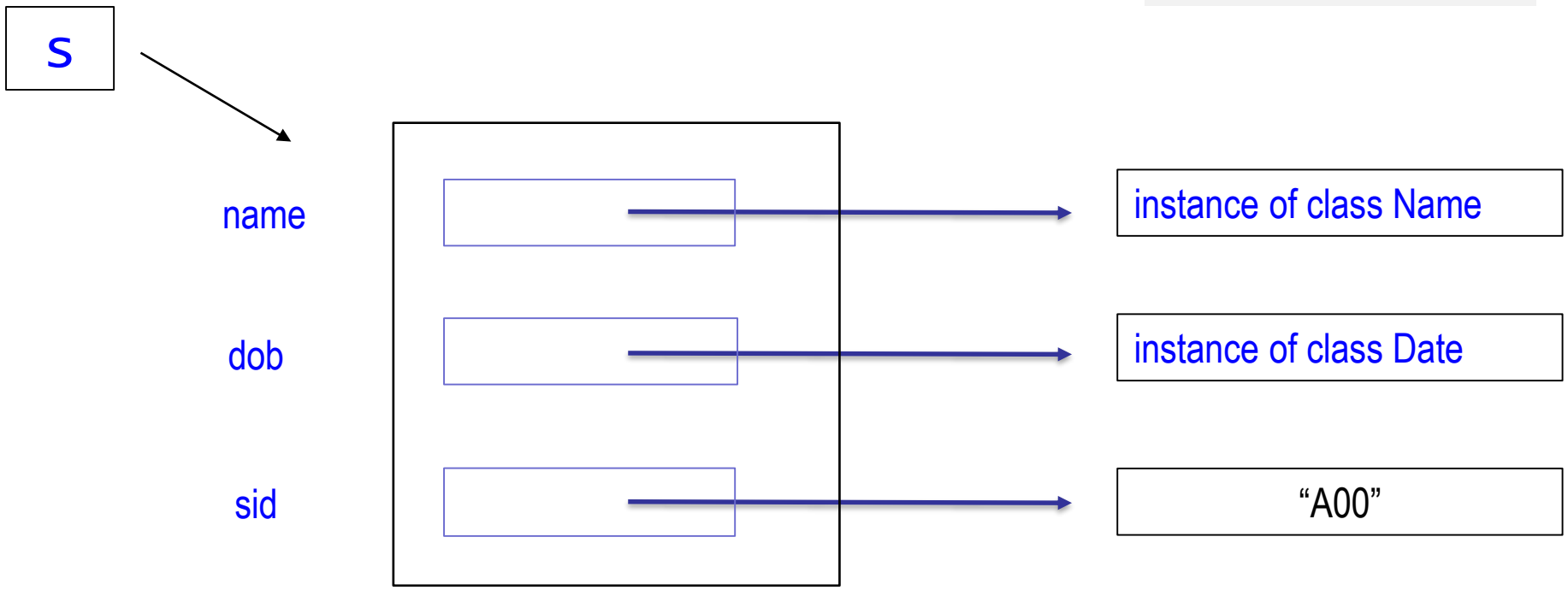


Why the need for classes?

```
class Student {
    Name name;
    Date dob;
    String sid;
}
```

Create an instance of the Student class!

A more accurate visualization…

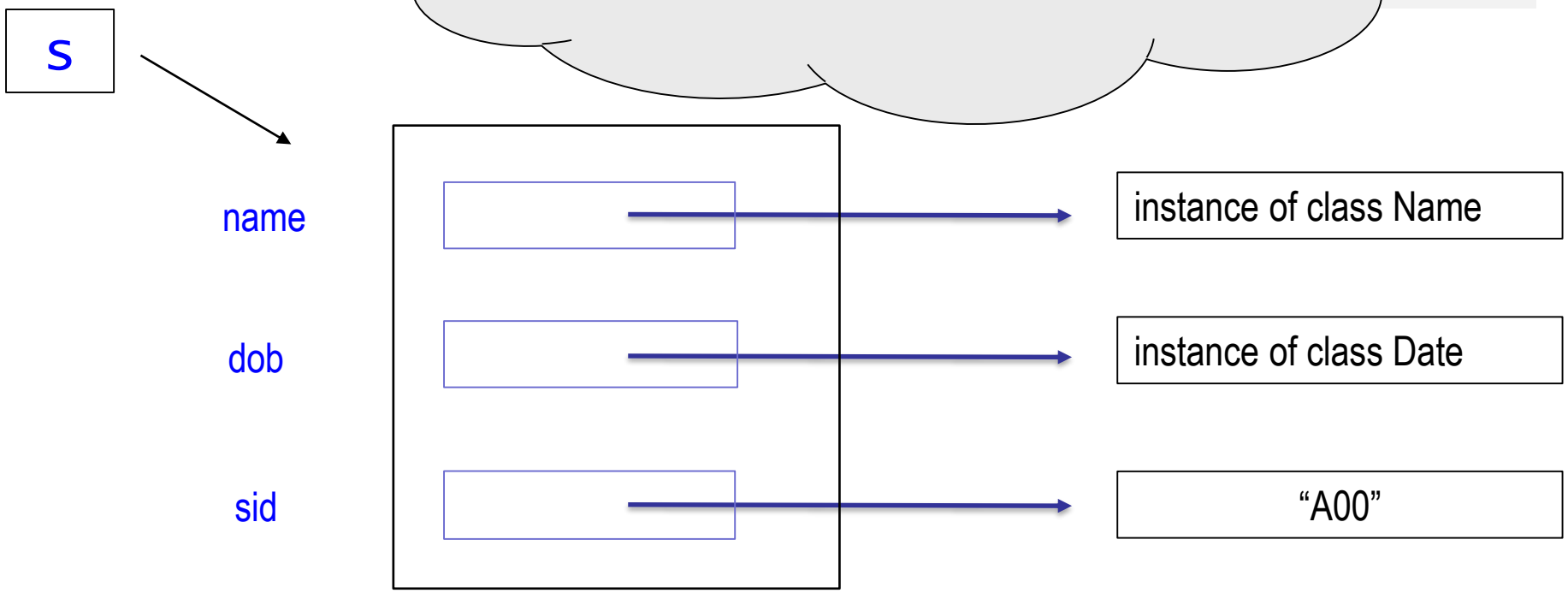


Why the need for classes?

```
class Student {
    Name name;
    Date dob;
    String sid;
}
```

Create an instance of the Student class!

This is referred to as *object composition*. When one class is composed of objects of other classes!



Why the need for classes?

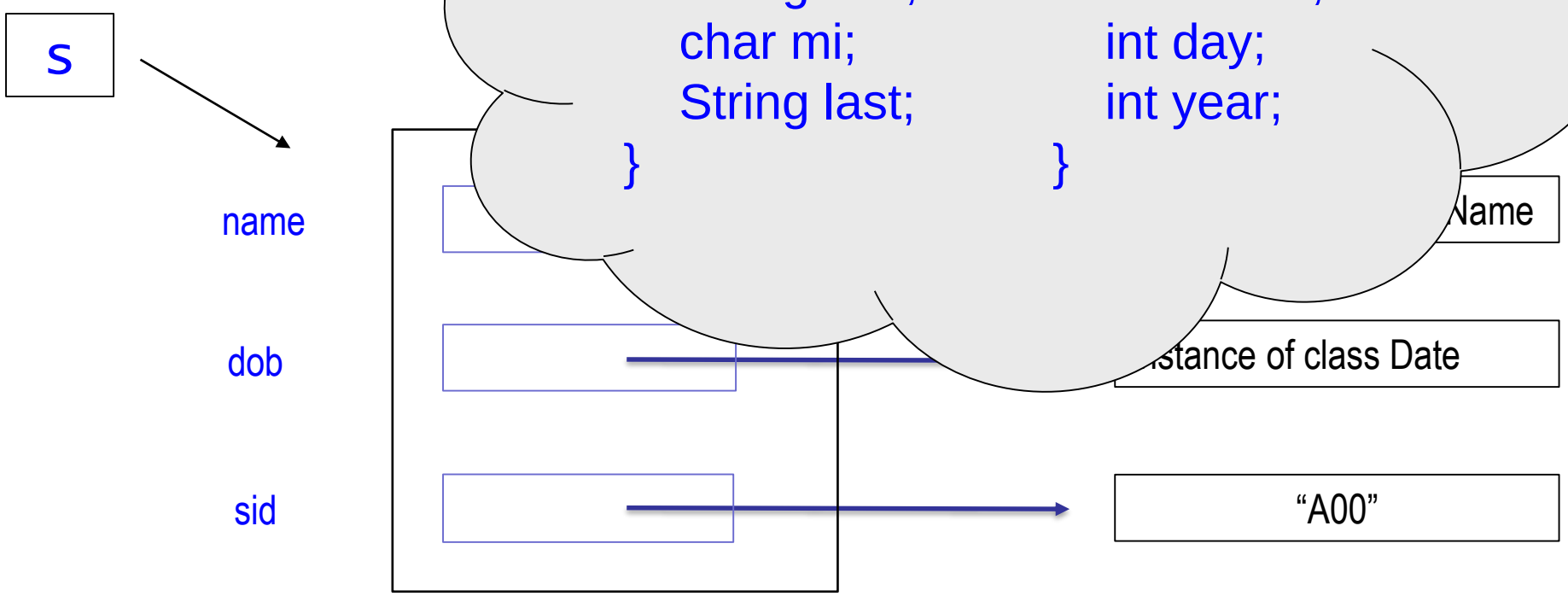
```
class Student {
    Name name;
    Date dob;
    String sid;
}
```

Create an instance of the Student class!

Assuming the following defintions:

```
class Name {
    String first;
    char mi;
    String last;
}
```

```
class Date {
    int month;
    int day;
    int year;
}
```

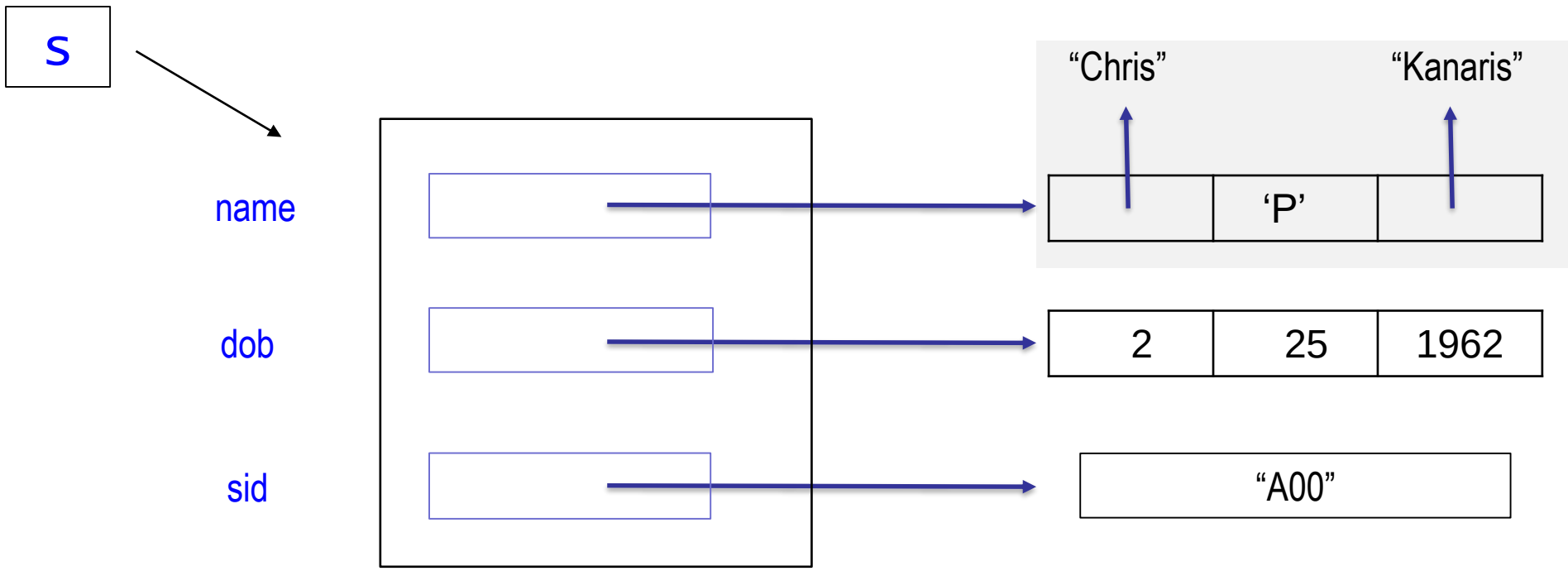


Why the need for classes?

```
class Student {
    Name name;
    Date dob;
    String sid;
}
```

Create an instance of the Student class!

A more accurate visualization…

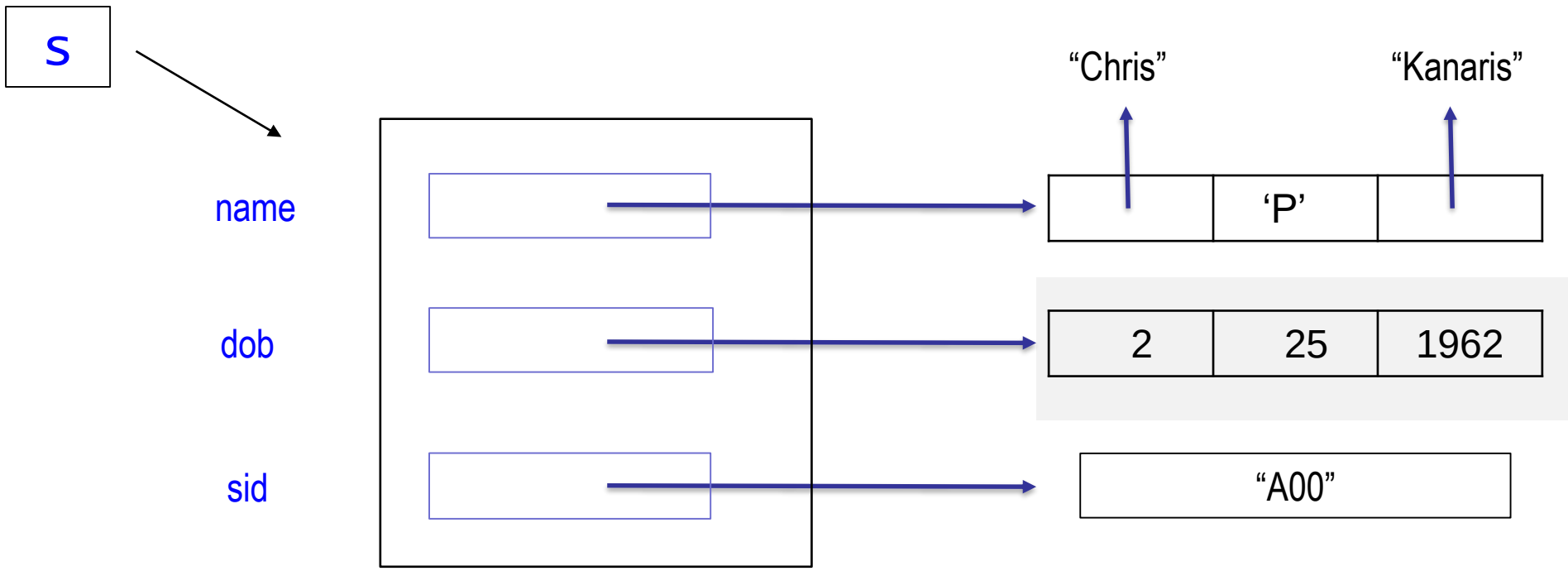


Why the need for classes?

```
class Student {
    Name name;
    Date dob;
    String sid;
}
```

Create an instance of the Student class!

A more accurate visualization…

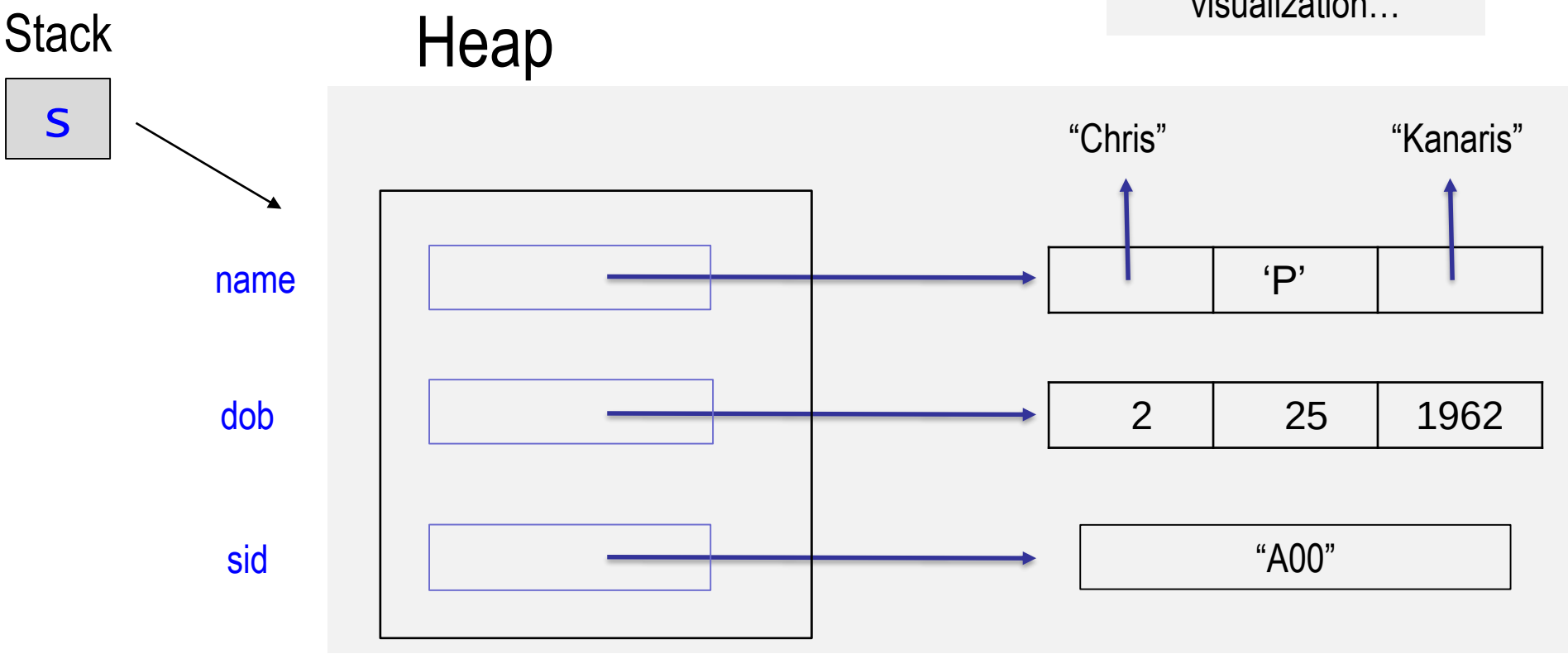


Why the need for classes?

```
class Student {
    Name name;
    Date dob;
    String sid;
}
```

Create an instance of the Student class!

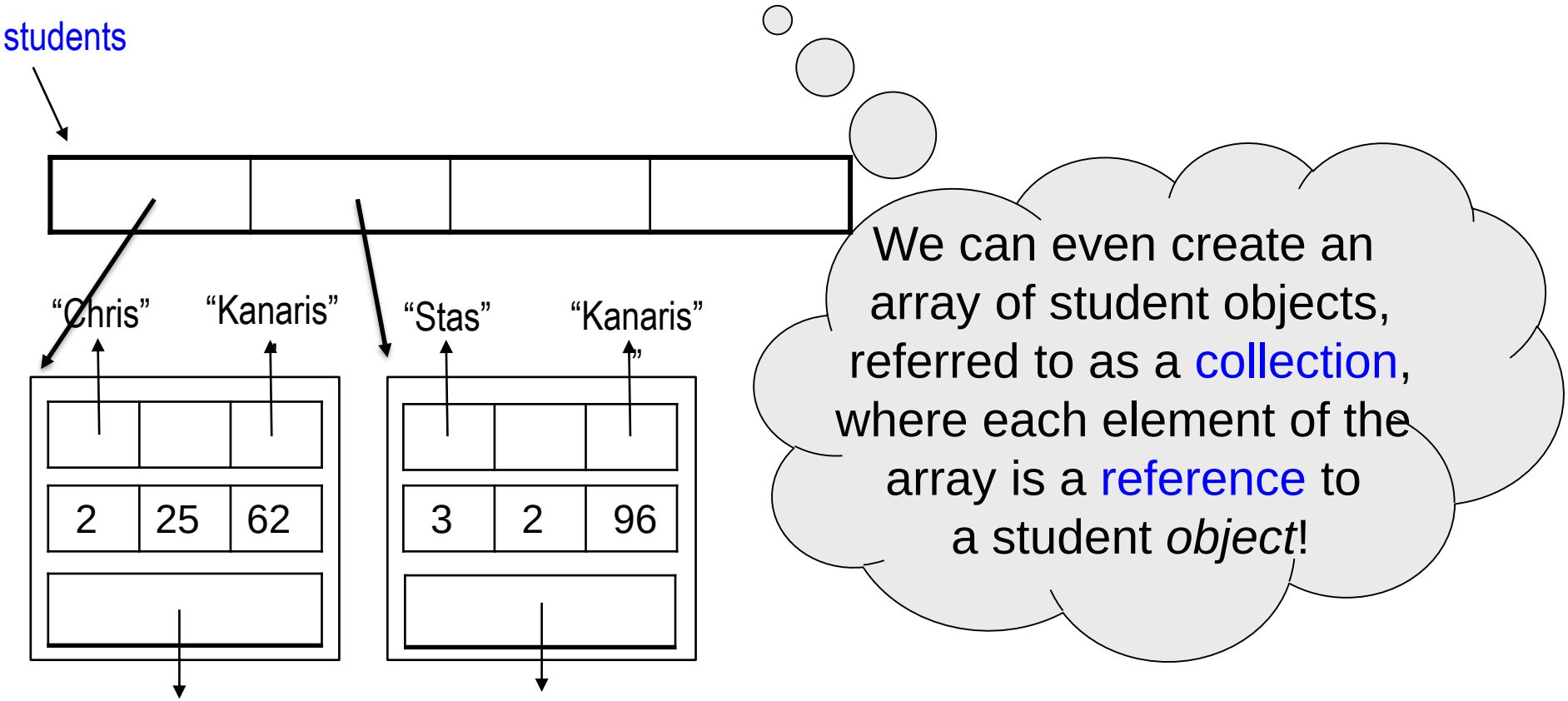
A more accurate visualization…



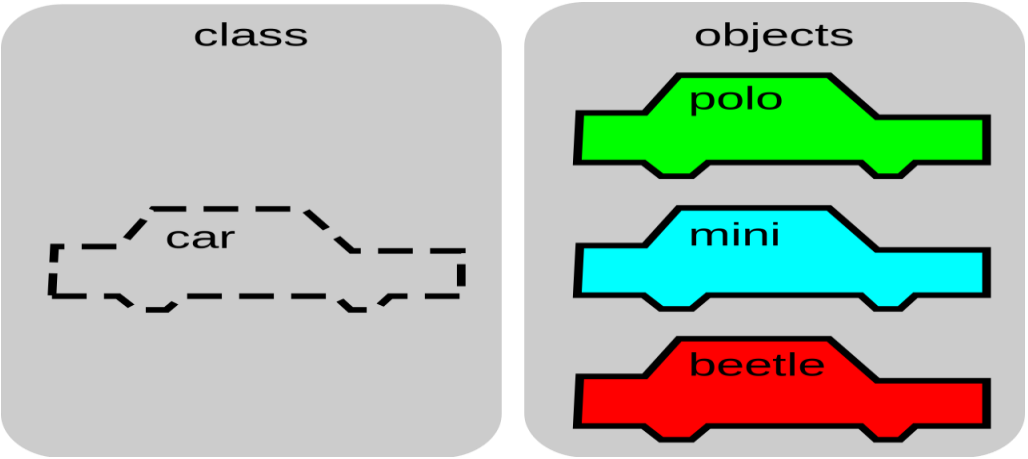
Why the need for classes?

```
Student students = new Student[4];

students[0] = // create an instance of Student
students[1] = // create an instance of Student
```



Why classes?



Classes allow us to specify a blueprint that can be used to create a *physical structure of data* that *models* the logical entity it is representing!

Each physical structure (or object) created is an ***instance*** of the class!

Why classes?

The purpose of adding functions or *methods* to a class is to provide a class with all the functionality that its' objects need to perform.