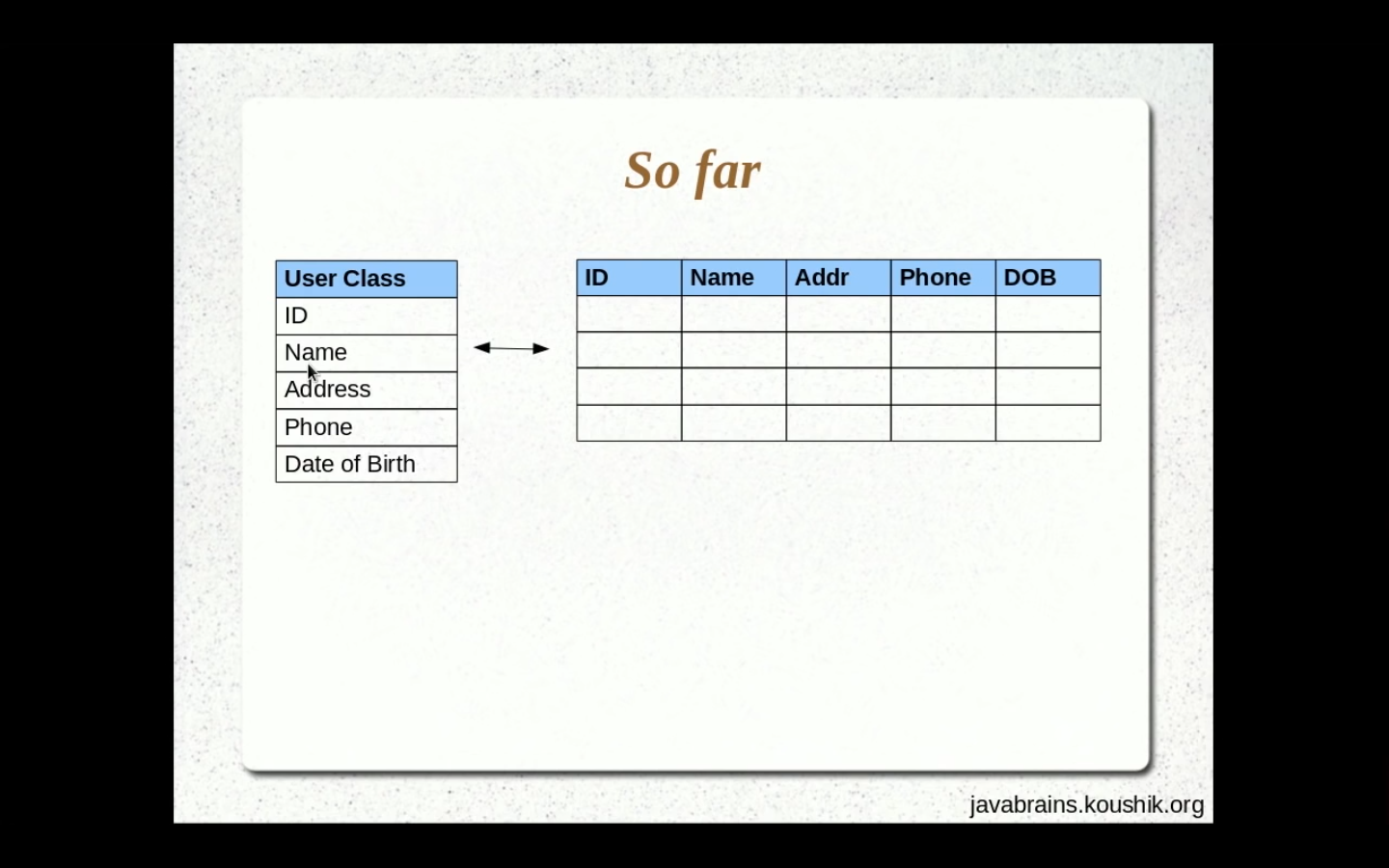
**Embedding Value Object as filed to the Entity**

I the below picture we have the user table in following structure.

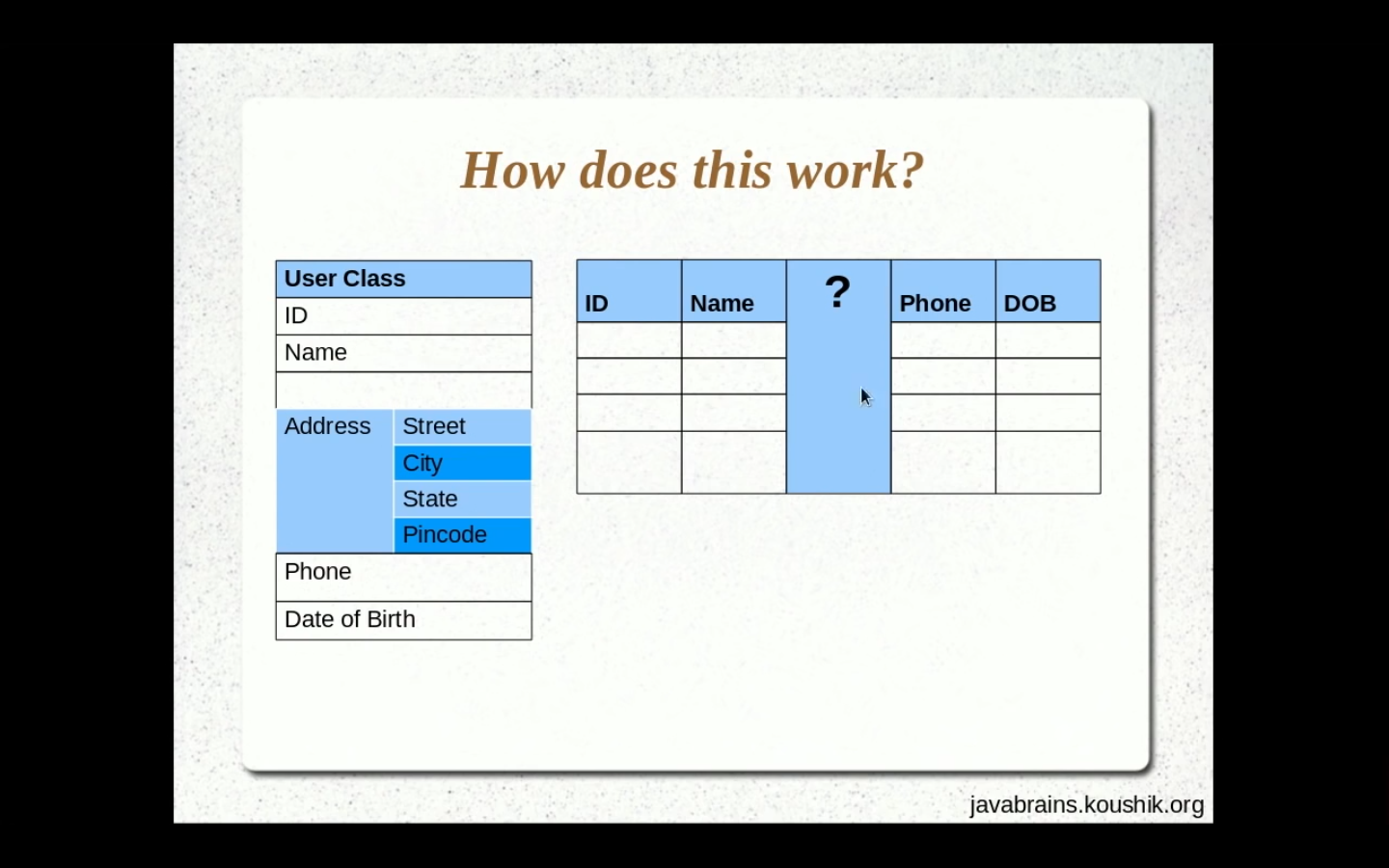
Consider the user has field user\_id, name, address, phone no and DOB.



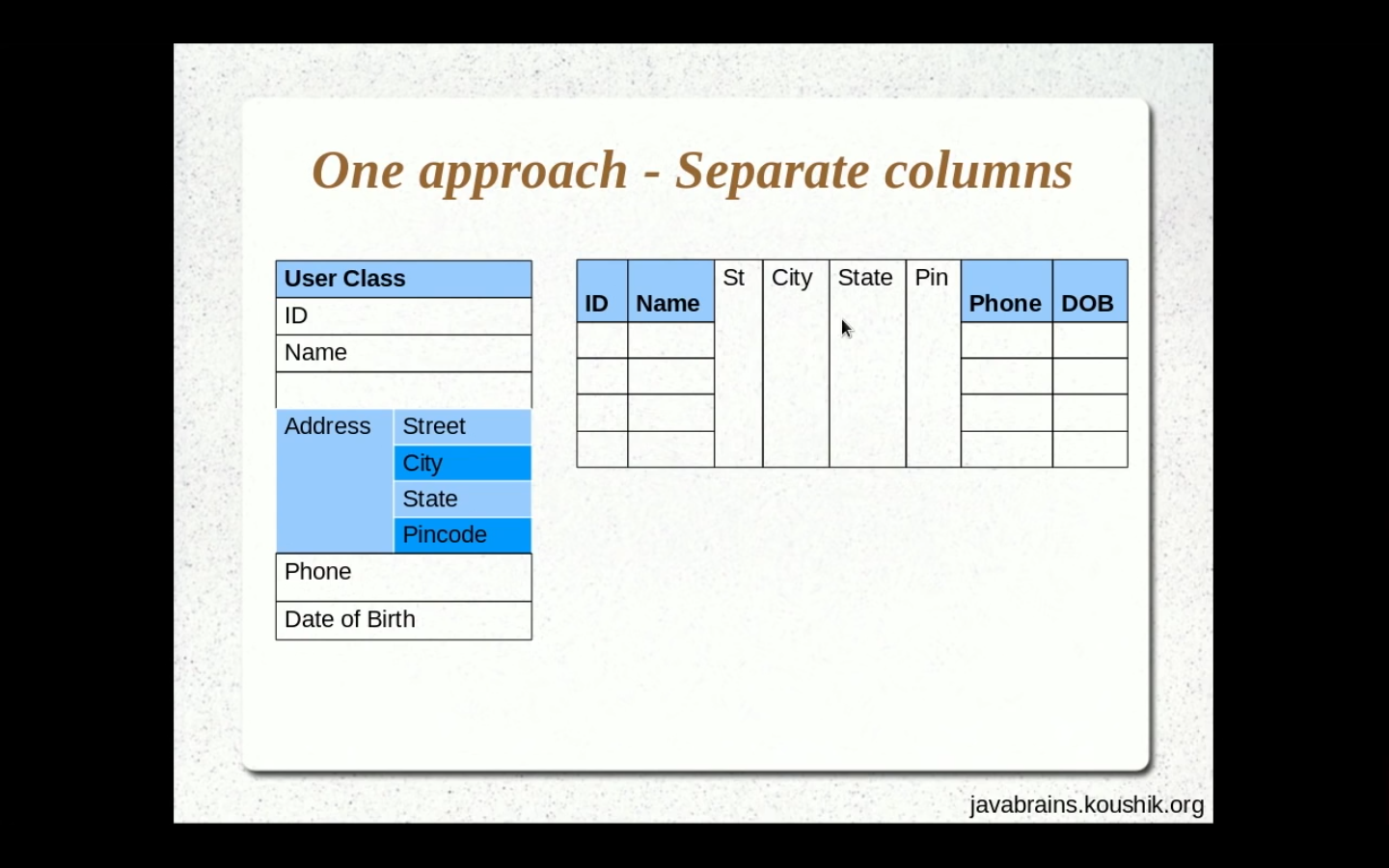
But address can have Street, city, state and Pincode, so the table continue to grow.

Idea here is we can the address class separately and embedded it as a VALUE OBJECT.

And we are planning to inject(Embed) the address value object to User table.



Once we are done injecting it into the user table the below is how the user table looks like.



What is the difference between Entity and Value object?

* Entity has its own meaning but value object (ADDRESS) does not have its own meaning without the parent (USER).
* Entity is created as new table where as value object (ADDRESS) is not created as a new table it added to the same parent (USER) itself.
* Value object can use all the annotation same as entity. Eg VO can have @Column, @Temporal, @Transient.

@Embeddable

**public** **class** Address {

* So @Embeddable annotation denotes the that address class is the value object(VO).

@Entity(name="UserInfo")

**public** **class** User

{

@Embedded

**private** Address address;

* So @Embedded annotation helps is embedding the VO(ADDRESS) to entity(USER).

@Entity(name="UserInfo")

**public** **class** User

{

@EmbeddedId

**private** SocialSecurity socialSecurity;

Consider U have VO that has fields social\_security, PAN\_card, email\_address. So in this case all those fields are unique to each user.

Consider If U want to have VO(SocialSecurity) as a primary key column in user entity table. Then @EmbeddedId annotation helps U have the VO as embedded as primary key to UR entity.