

Python Data Model

Any data in python is represented by an object or by relationships between objects.

Every object has:

- an **identity**
- a type
- a value

Python Data Model Every object has an identity, a type and a value id() Identity unique integer **IDENTITY** address in print()** never changes **OBJECT** VALUE TYPE Type Value An object's type determines A value is a basic unit of data the **operations** that the object and belongs to a type. A value supports, the **mutability** and represents specific instance (or also defines and the an example) of a given type and type() possible values for can be referenced by a variable objects of that type. ** or inspect

Python Data Model

Any data in python is represented by an **object** or by relationships between objects.

Every object has:

- an identity id()
- 2. a **type type ()**
- 3. a value print() / inspection

Python Data Model Every object has an identity, a type and a value id(x)Identity unique integer **IDENTITY** address in print(x) ** never changes VALUE TYPE Type Value An object's type determines A value is a basic unit of data the **operations** that the object and belongs to a type. A value supports, the **mutability** and represents specific instance (or also defines and the an example) of a given type and type(x) possible values for can be referenced by a variable objects of that type. ** or inspect

Python Data Model

Example:

```
x = [1, 2, 3, 1000]
```

- print(id(x)) # prints
 the unique identity of
 the object
- print(type(x)) # prints
 the type of the object
- print(x) # prints thevalue of the object

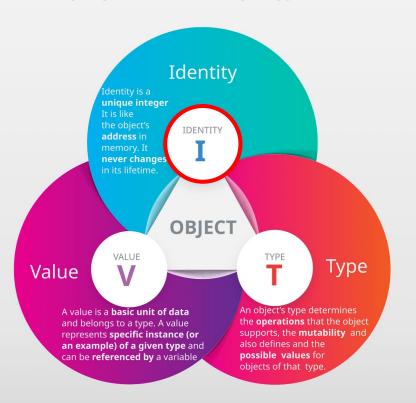
Python Data Model Every object has an identity, a type and a value id(x)2170951156608 Identity unique integer **IDENTITY** address in print(x) ** never changes [1,2,3,1000] VALUE TYPE Type Value An object's type determines A value is a basic unit of data the operations that the object and belongs to a type. A value supports, the **mutability** and represents specific instance (or also defines and the an example) of a given type and type(x) possible values for can be referenced by a variable objects of that type. <class 'list'> ** or inspect x in REPL

Python Data Model

Example:

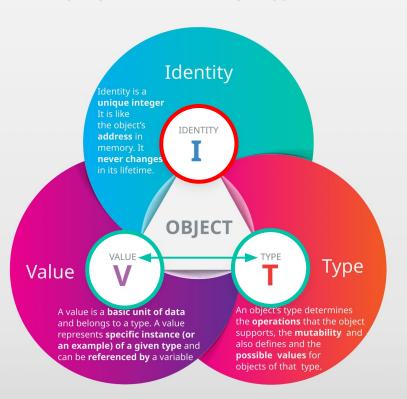
```
x = [1, 2, 3, 1000]
```

- print(id(x)) # prints
 the unique identity of
 the object
- print(type(x)) # prints
 the type of the object
- print(x) # prints thevalue of the object



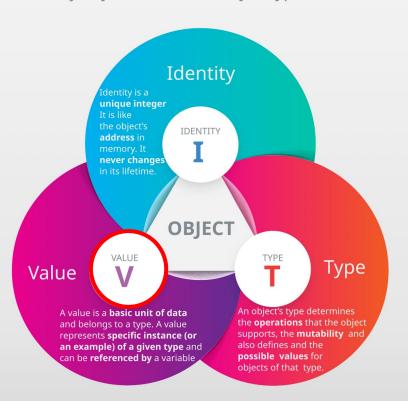
Python Data Model

Objects	having	same	identity	will
nave <mark>sa</mark> r	ne			



Python Data Model

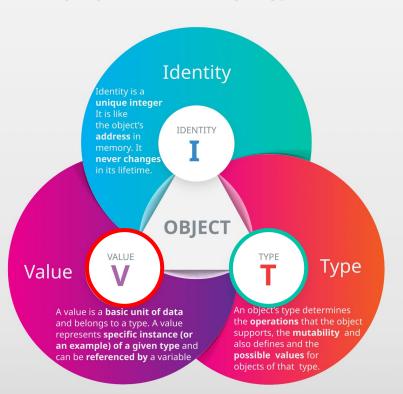
Objects having same identity will have same value and type.



Python Data Model

Objects having same identity will have same value and type.

Objects having same value will have same ____.



Python Data Model

Objects having same identity will have same value and type.

Objects having same value will have same type.

Acknowledgement

Infographic: "Designed by SilviaNatalia / Freepik"

https://docs.python.org/3/reference/datamodel.html