**Database Design for eCommerce website:**

**Product Table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Product Id | Product Name | Product  Description | Price | Ratings | No of Reviews | No of the questions answered |
| 1 | Logitech Mouse | Logitech B170 Wireless Mouse, 2.4 GHz with USB Nano Receiver, Optical Tracking | 799 | 4.5 | 1000 | 453 |

**Field Table**

|  |  |
| --- | --- |
| Field id | Field Name |
| 1 | Color |
| 2 | Height |
| 3 | Width |
| 4 | Toxic |
| 5 | Vegan |

**Product Field Mapping**

|  |  |  |
| --- | --- | --- |
| Product Id | Field Id | Field Value |
| 1 | 1 | Black |
| 1 | 2 | 10 mm |
| 1 | 3 | 20 mm |

**Metaprogramming:**

Metaprogramming is the code that manipulates code.

**Code:**

num = 23

print("Type of num is:", type(num))

lst = [1, 2, 4]

print("Type of lst is:", type(lst))

**Output:**

Type of num is: <class 'int'>

Type of lst is: <class 'list'>

In Python, they are objects of int class or str class. So we can make a new type by creating a class of that type. For example, we can create a new type of *Student* by creating a *Student* class.

**Code:**

class Student:

pass

stu\_obj = Student()

# Print type of object of Student class

print("Type of stu\_obj is:", type(stu\_obj))

**Output:**

Type of stu\_obj is: <class '\_\_main\_\_.Student'>

A Class is also an object, and just like any other object, it’s an instance of something called Metaclass. A special class type creates these Class object. The type class is default metaclass which is responsible for making classes. For example in the above example, if we try to find out the type of Student class, it comes out to be a type.

**Code:**

class Student:

pass

# Print type of Student class

print("Type of Student class is:", type(Student))

**Output:**

Type of Student class is: <class 'type'>

This whole meta thing can be summarized as – Metaclass create Classes and Classes creates objects

**Reference:**

1. <https://www.geeksforgeeks.org/metaprogramming-metaclasses-python/>
2. https://medium.com/fintechexplained/advanced-python-metaprogramming-980da1be0c7d