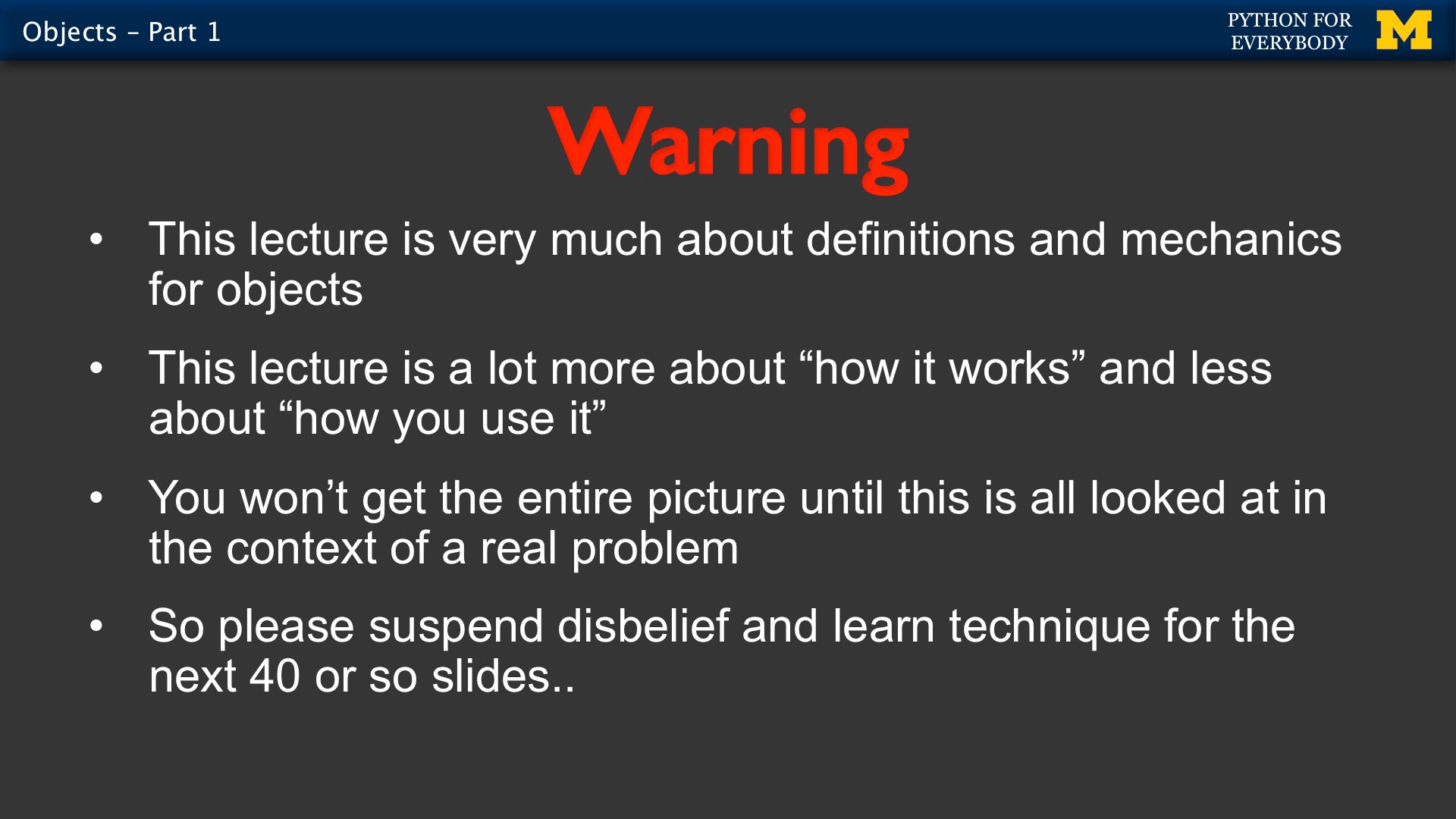


t!;

**UMSI**





1. **. Data Structures**

# Th i s chapter descr·ibes some thing s yo u' ve l earn ed about already ijn more det ail, and adds some new things as well.

* 1. **More on List s**

# The list data type has some more methods. Here are all of the methods of li st o bj ects:

l i s t . **a pp e nd (x)**

Add an item to the end of the li st. Eq u i val ent to a[len(a): J - ( x J .

l is t . **e x t e nd ( L)**

Extend the list by append i ng all the items in the given list. Equiva lent to a [ l e n ( a ) : J - L .

list. **i sert(,;**x)

# Insert an item at a given p osit io n. The first argum.ent is the index of the elemen t before which to insert , so

a . insert( o, x ) inserts at the front of the list, and a . i ns er t ( l e n ( a ) , x) is equivalent to a . appe nd{x ) .

l i s t . **r emove (x)**

# Remove the first item from the list whose value is *x.* It is an error if there is no such item.

l i s t . **pop([;])**

# Remove the item at the given position in the list, and return it. If no index is speci fied, a.pop() removes and returns the last item in the list. (The square bracket s around the ; in the rm ethod sig nat ur e denote that the pa rameter is optional, not that you should type square bracket s at that position. You will see this notation frequ ent ly in the Pytho n Library Reference.)

**https://docs.python.org/3/tutorial/datsatructures.html**

12.6. sqlite3

databases

DB- API 2.0 interface for SQLite



**Sou rce code:** Lib / sqlite 3/

SQLite i•s a C library that provides a lightweight disk- based database that doesn't requ ire a separate server process and allows accessing the database using a nonstandard variant of the SQL query language. Some applications can use SQ Lite for int ernal data storage. It ' s also p o ssible to pr ot otype an application using SQLite and then port the code to a larger database such as PostgreSQL or Oracle.

The sqlite3 module was written by Gerhard Haring. It provides a SQL int erface compliant wit h the DB- API 2.0 specifi cation described by **PEP 2 4 9 .**

To use the modu le, you mu-st first create a connection= = that represents the database. Here the data will be sto red in the example.db file:

**sqlitel**

sqlite3.connect'{example.db')

I

u ••

# •

You can also supply the special name :memor:y to creat e a dat abase1i n **RAM.**

Once you have a connect i on, you can create a cursor ob j ect and call its execute() method to perfor m SOL commands:

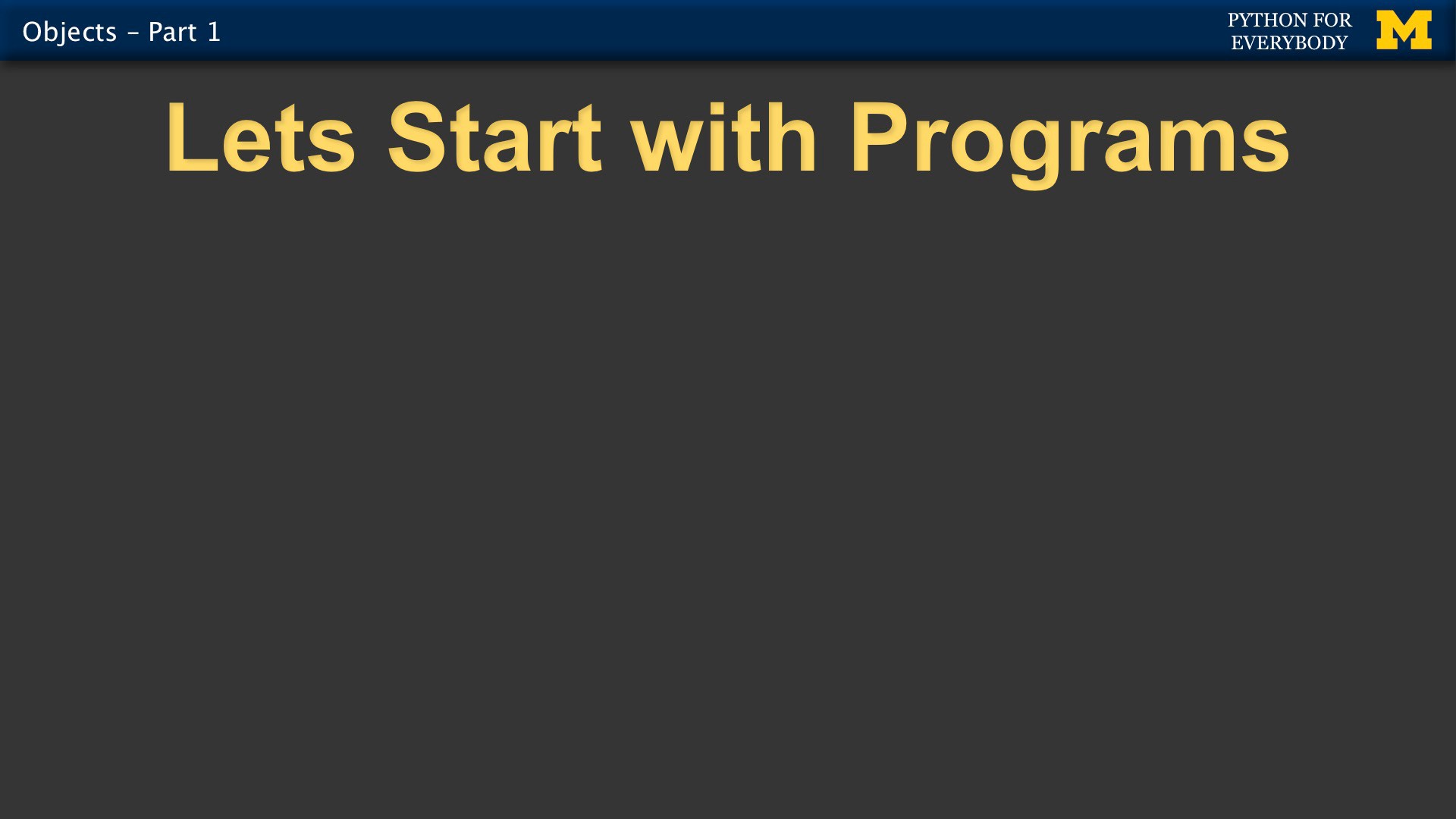
c = c o nn . c ur s or { )

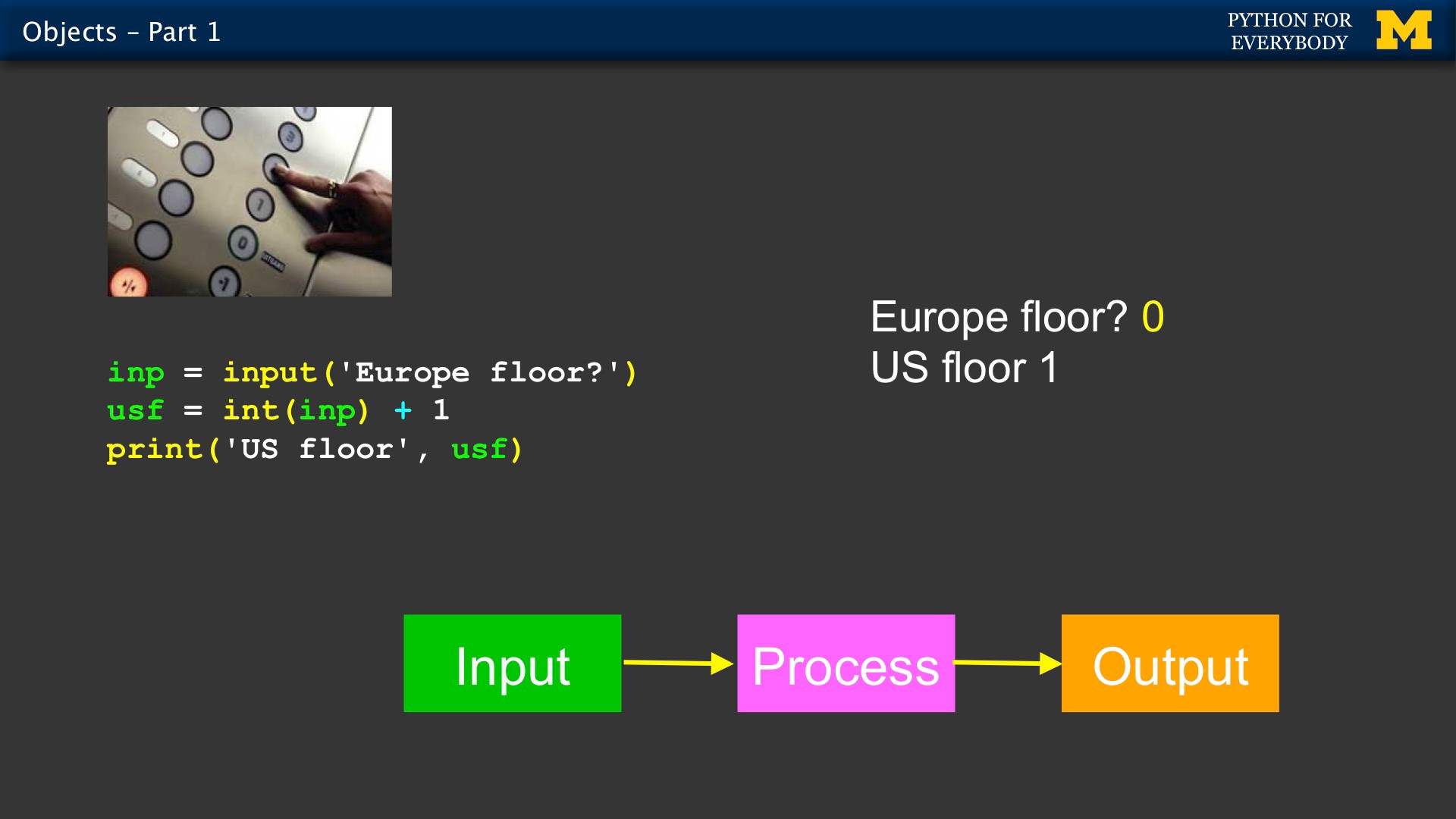
*#* create *table*

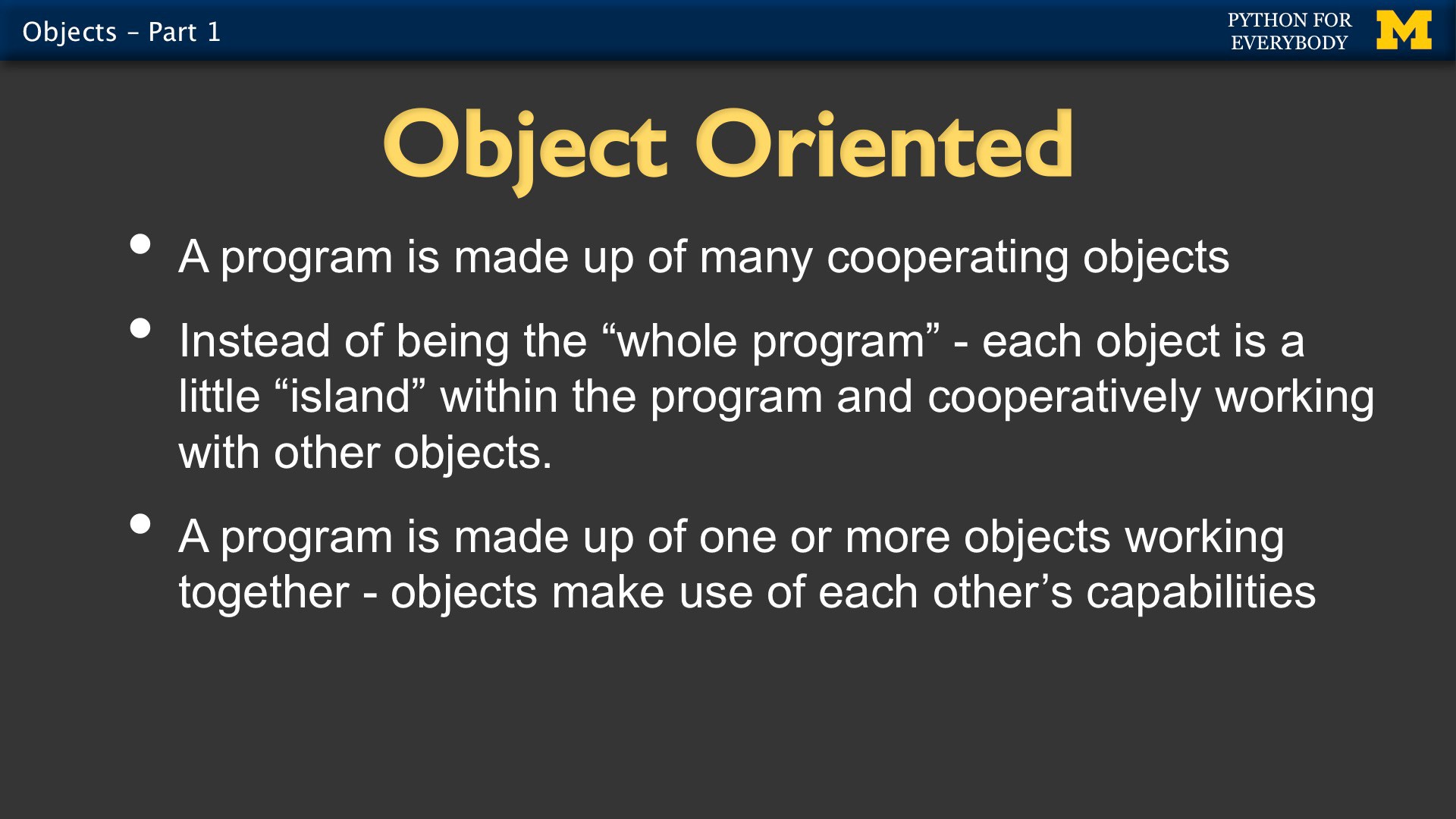
c.execute('''CREATE TABLE stocks

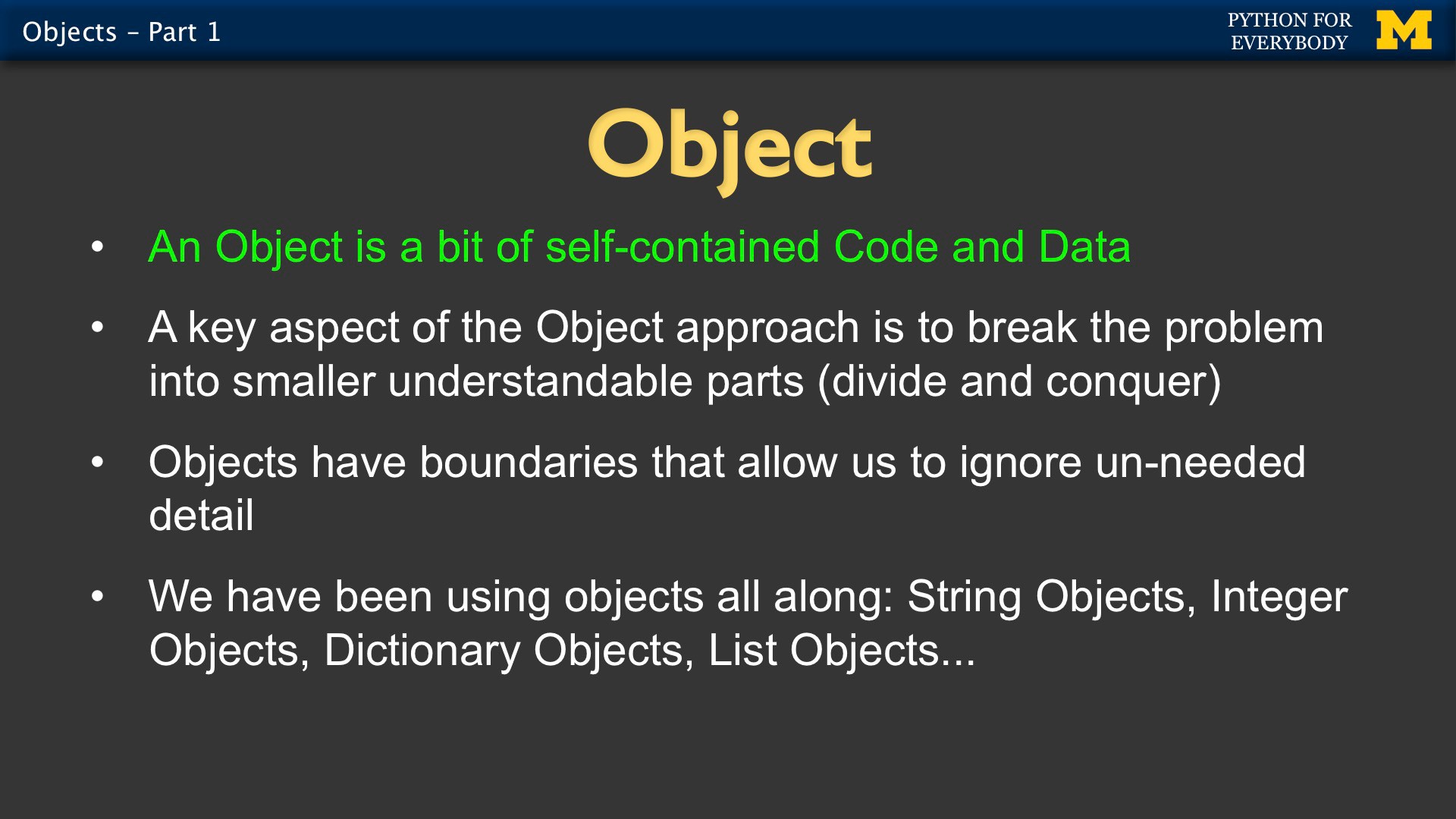
(date text, trans text, symbol text, qty real, price real)''')

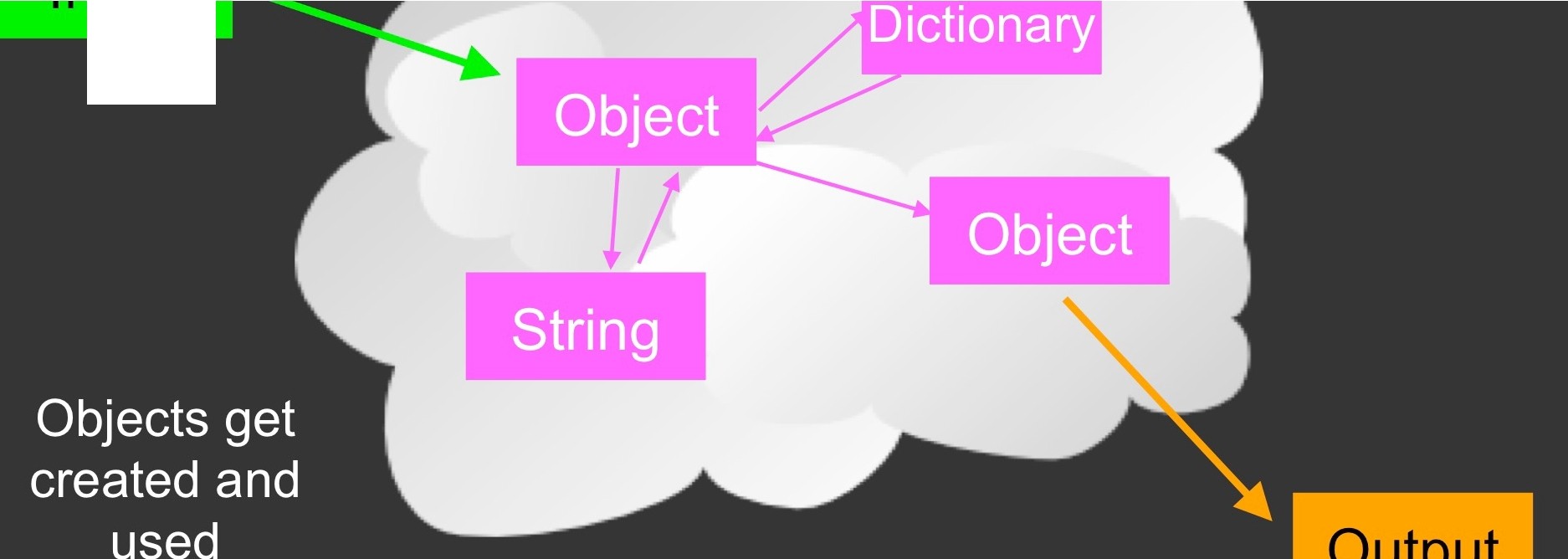
https://docs.python .org/3/1ibrary/sqIite3.htmI





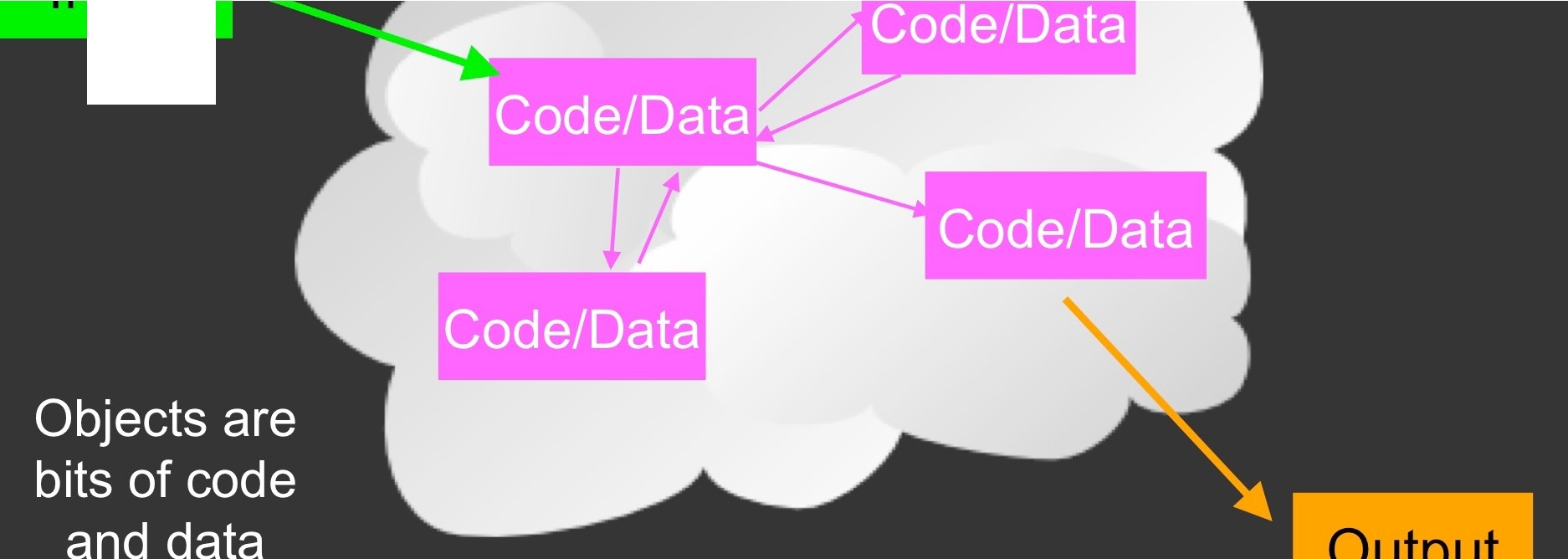






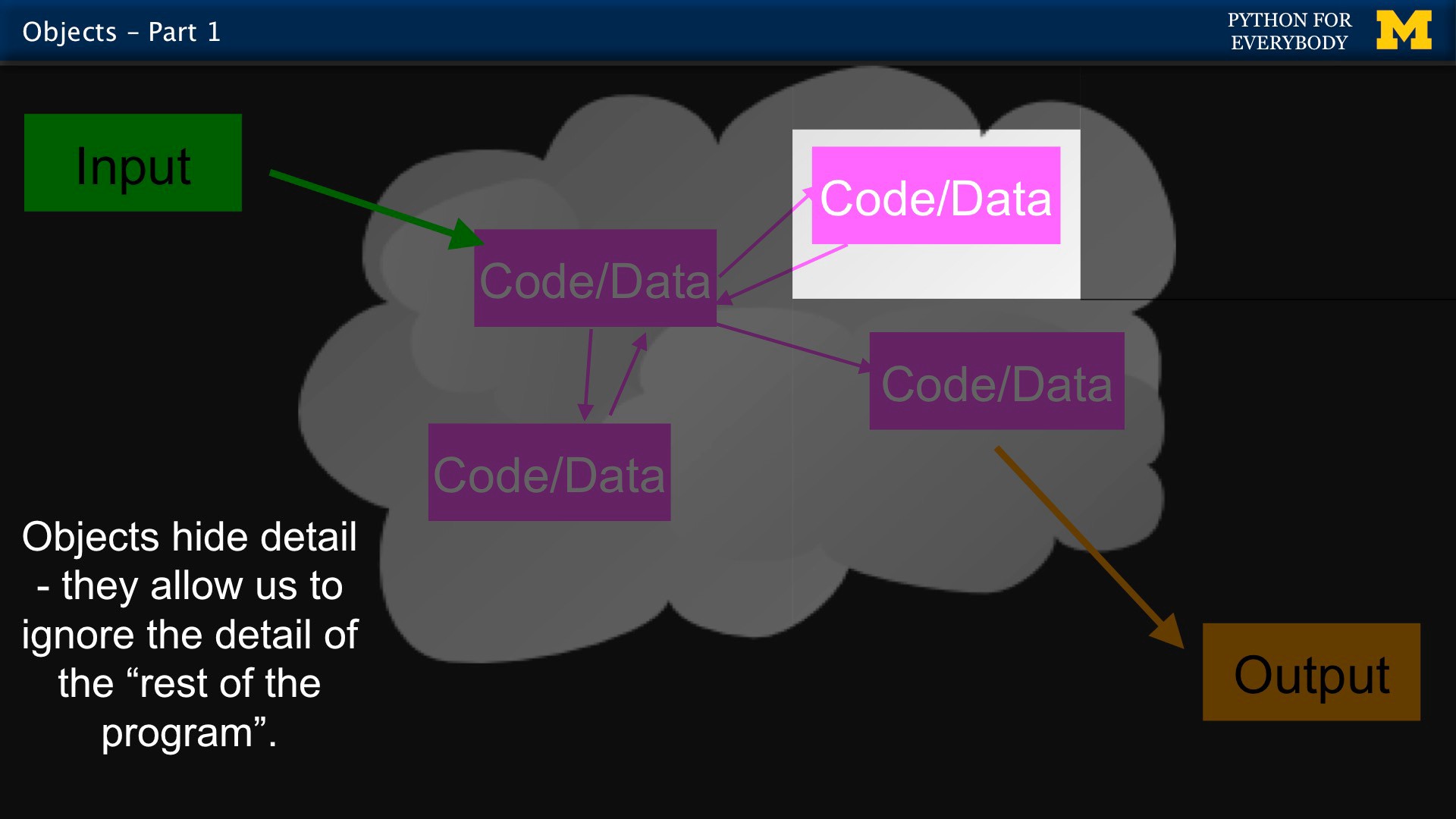
Input

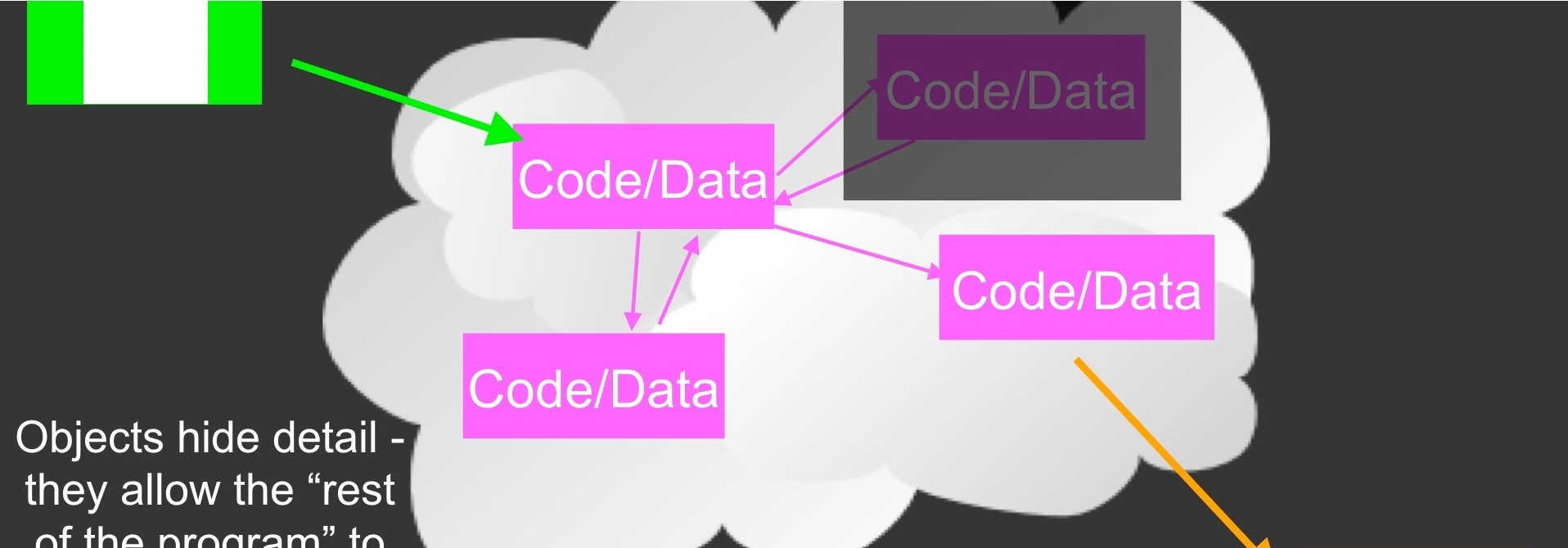
Output



Input

Output





Input

Output

