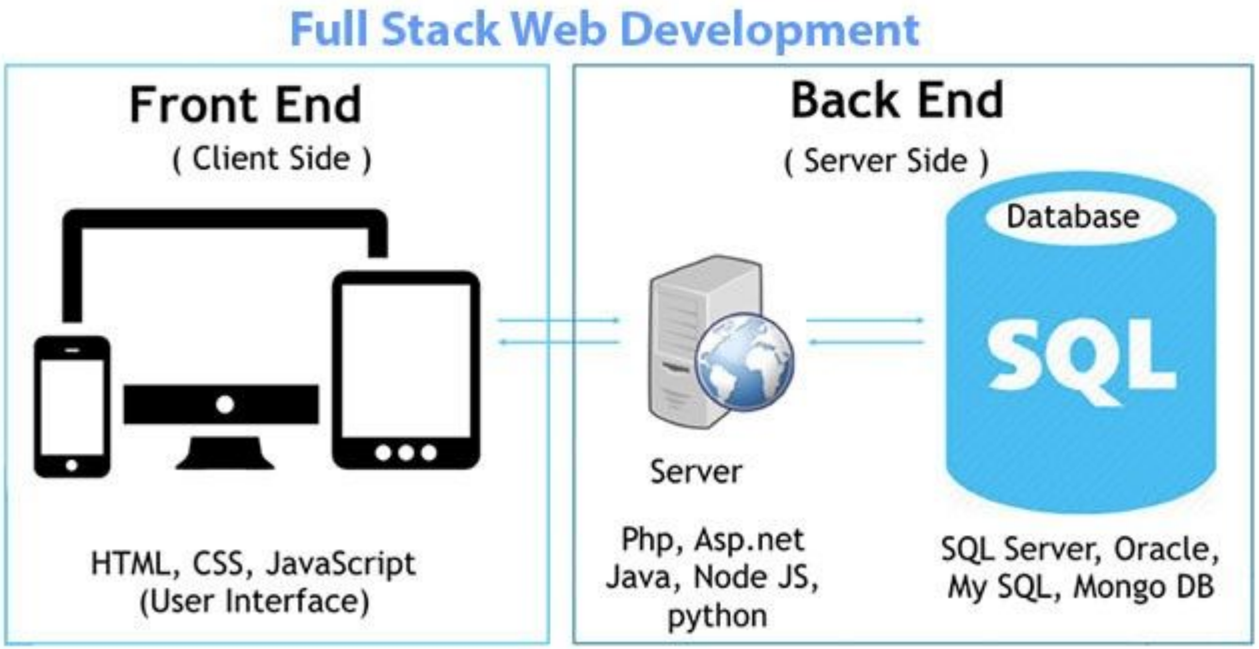
Institut für Geoinformatik
Universität Münster

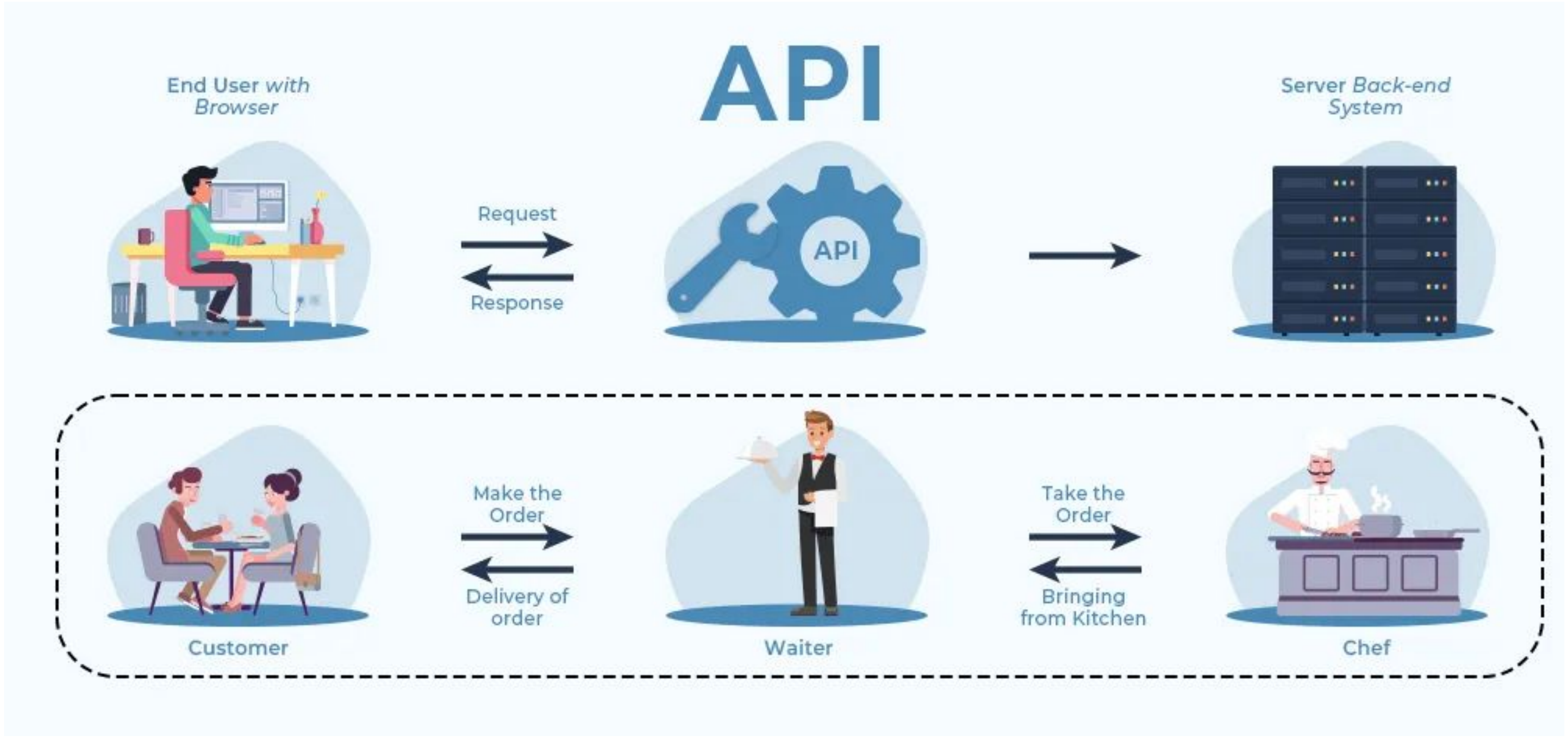
Universität Münster

REST APIs and ORM

B. Pondi & C. Knoth

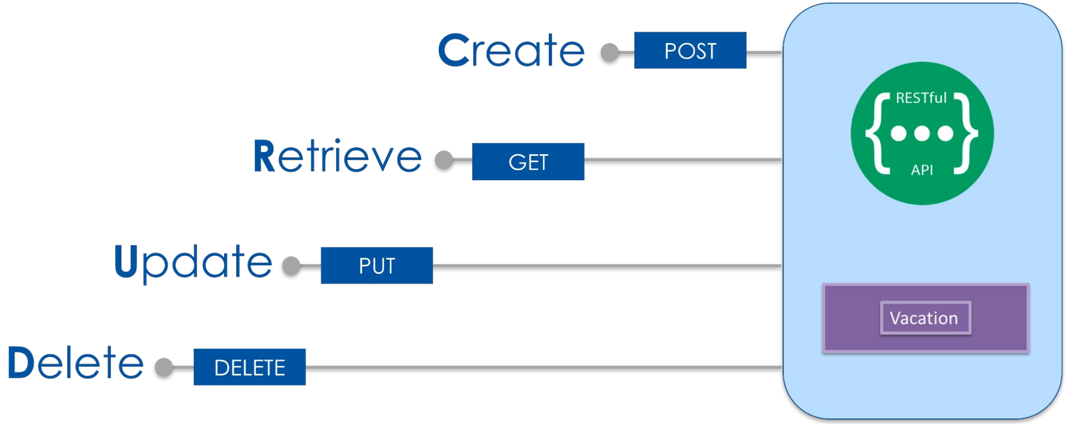
REST APIS

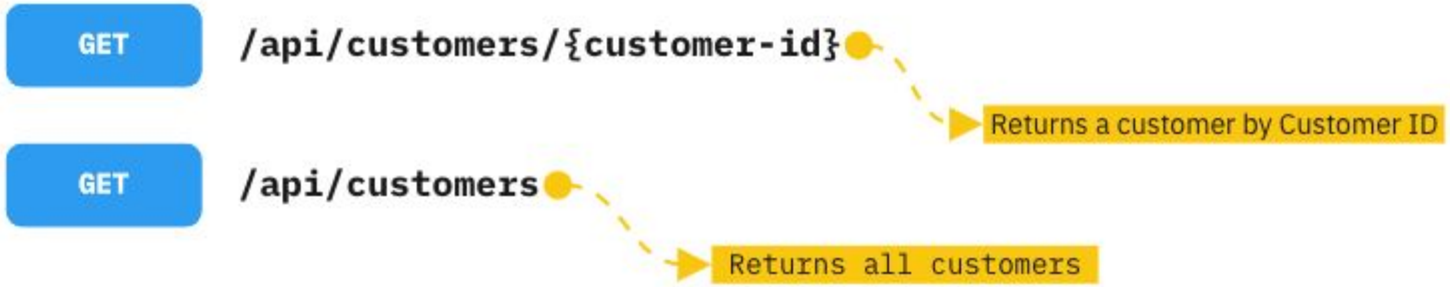

Full Stack Web Development
Front End
(Client Side)
HTML, CSS, JavaScript
(User Interface)
Back End
(Server Side)
Server
Php, Asp.net
Java, Node JS,
python
Database
SQL
SQL Server, Oracle,
My SQL, Mongo DB


API
End User *with Browser*
Request
Response
API
Server *Back-end System*
Make the Order
Delivery of order
Take the Order
Bringing from Kitchen
Customer
Waiter
Chef

HTTP Verbs

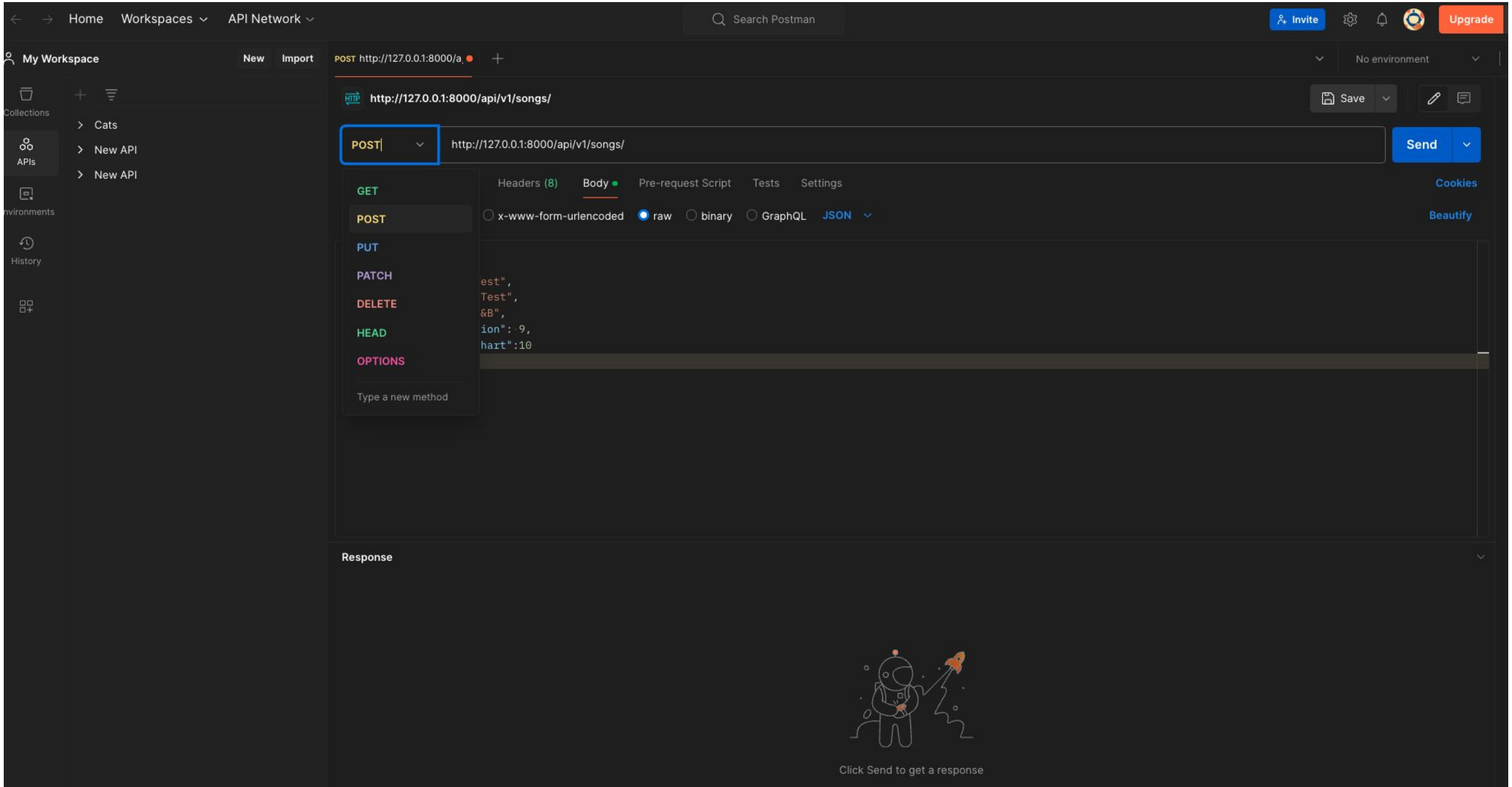
- GET
- POST
- DELETE
- PUT
- PATCH




GET
/api/customers/{customer-id}
Returns a customer by Customer ID
GET
/api/customers
Returns all customers

1XX	Informational codes	The server acknowledges and is processing the request.
2XX	Success codes	The server successfully received, understood, and processed the request.
3XX	Redirection codes	The server received the request, but there's a redirect to somewhere else (or, in rare cases, some additional action other than a redirect must be completed).
4XX	Client error codes	The server couldn't find (or reach) the page or website. This is an error on the site's side.
5XX	Server error codes	The client made a valid request, but the server failed to complete the request.

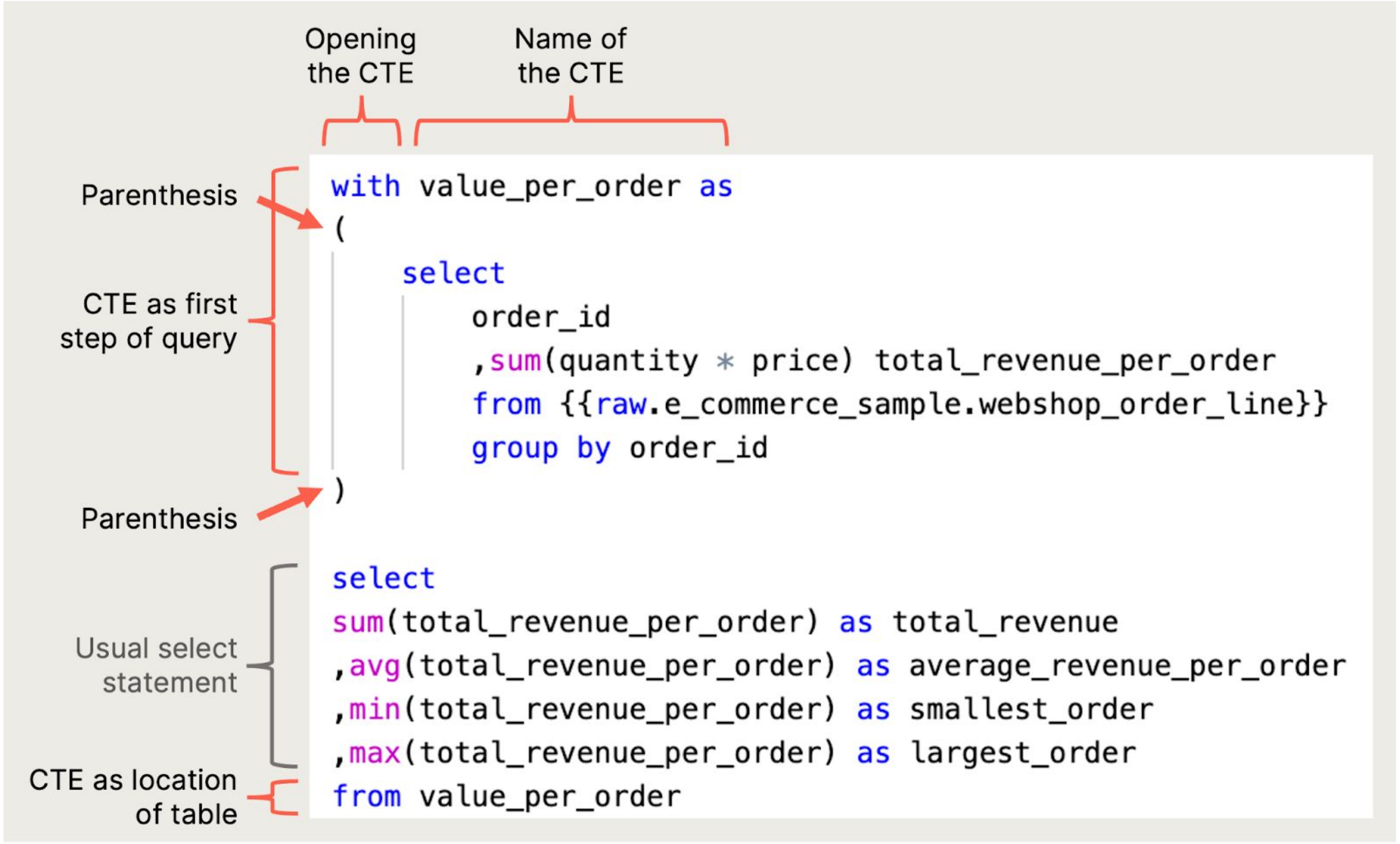
Example Tool for API Testing



POST http://127.0.0.1:8000/api/v1/songs/

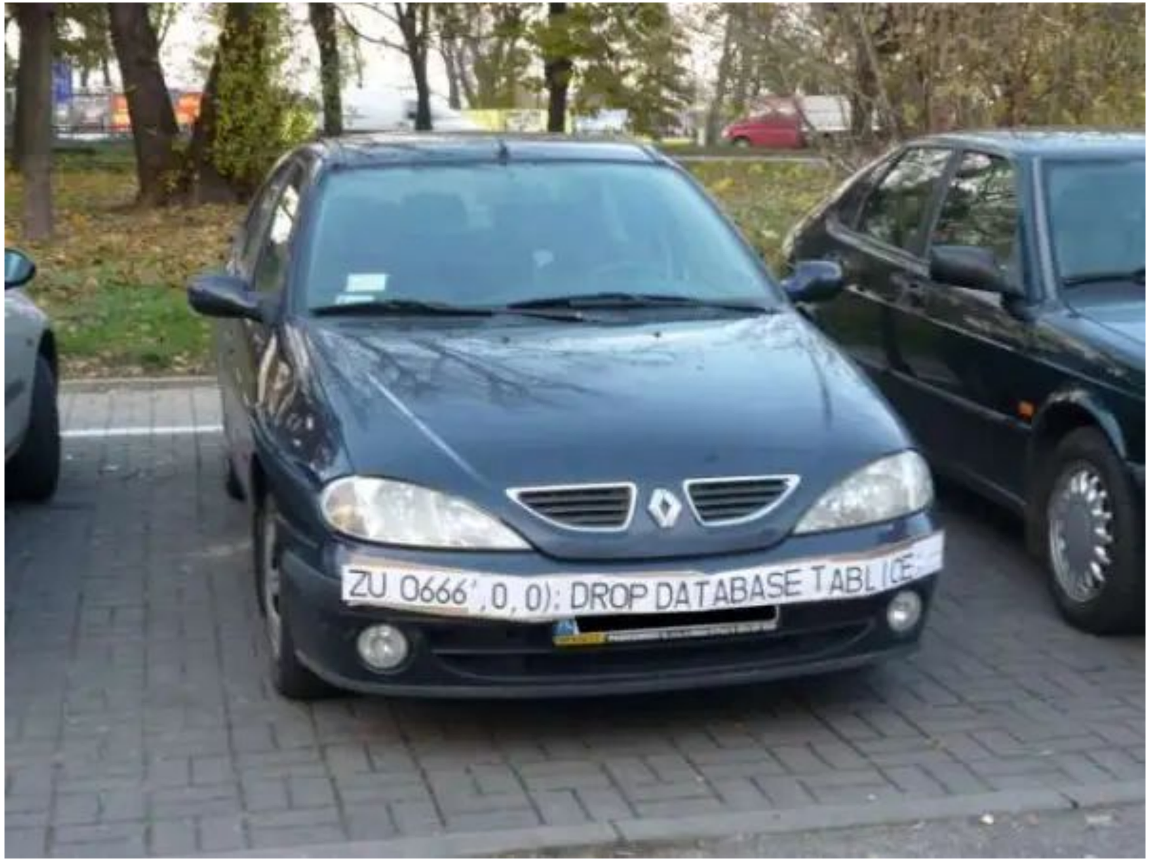
```
est",
Test",
&B",
ion": 9,
hart":10
```

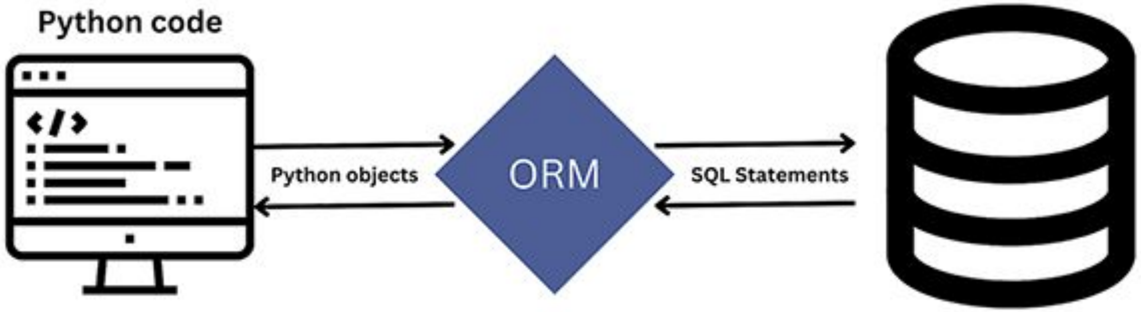
Object Relational Mapping (ORM)



```
with value_per_order as
(
    select
        order_id
        ,sum(quantity * price) total_revenue_per_order
        from {{raw.e_commerce_sample.webshop_order_line}}
        group by order_id
)

select
sum(total_revenue_per_order) as total_revenue
,avg(total_revenue_per_order) as average_revenue_per_order
,min(total_revenue_per_order) as smallest_order
,max(total_revenue_per_order) as largest_order
from value_per_order
```


ZU 0666',0,0); DROP DATABASE TABLICE;

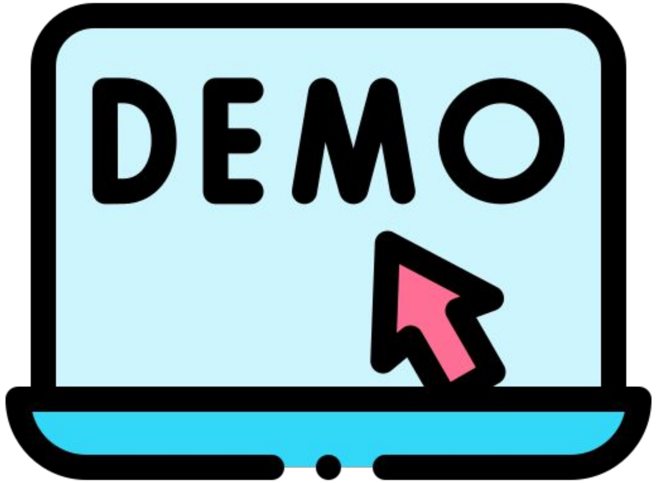

Python code
Python objects
ORM
SQL Statements

```
SELECT * FROM Books WHERE author = 'JK Rowling';
```

```
Books.objects.filter(author = 'JK Rowling')
```

```
Books.findAll({ where:
                {author = 'JK Rowling'}})
```

DEMO



Q & A

The End