Q1. I chose category, product, orderline, order, customer

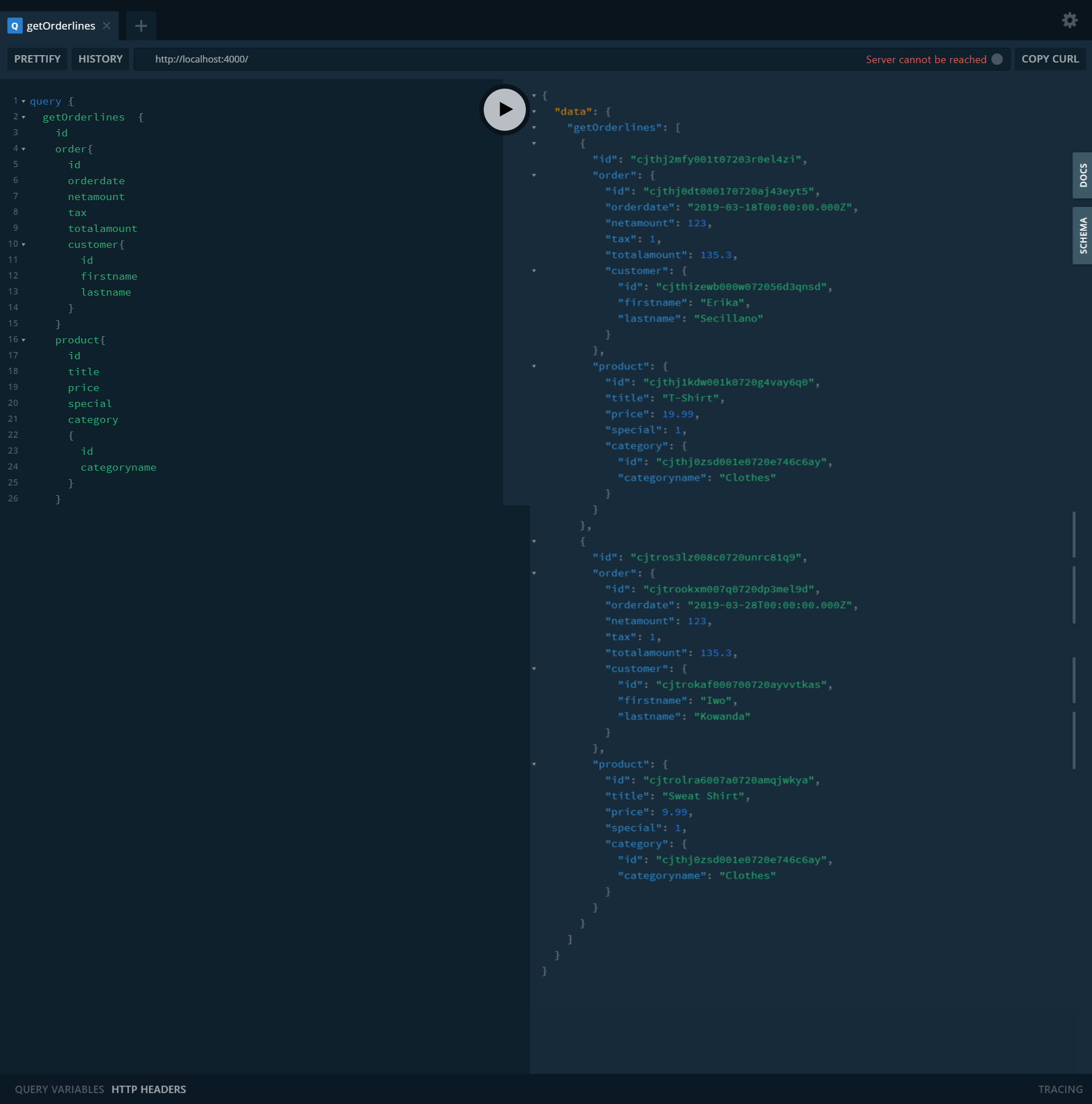
Q2,3,4 schema.graphql

|  |
| --- |
| scalar DateTime  type Query {  getCategories: [Category!]!  getProducts: [Product!]!  getOrderlines: [Orderline!]!  getOrders: [Order!]!  getCustomers: [Customer!]!  getInventories: [Inventory!]!  }  type Mutation {  createCategory(  categoryname: String!  ): Category  createInventory(  productId: ID!  quan\_in\_stock: Int  sales: Int  ): Inventory  createProduct(  categoryId: ID!,  title: String!,  actor: String,  price: Float!,  special: Float,  common\_prod\_id: Int  quan\_in\_stock: Int  ): Product  createOrderline(  orderId: ID!,  productId: ID!,  quantity: Int,  orderdate: DateTime  ): Orderline  createOrder(  orderdate: DateTime,  customerId: ID!,  netamount: Float!,  tax: Float!,  totalamount: Float!  ): Order  createCustomer(  firstname: String!,  lastname: String!,  address1: String!,  address2: String!,  city: String!,  state: String!,  zip: String!,  country: String!,  region: String!,  email: String!,  phone: String!,  creditcardtype: String,  creditcard: String,  creditcardexpiration: DateTime,  username: String!,  password: String!,  age: Int!,  income: Float,  gender: String!  ): Customer  }  type Category {  id: ID!  categoryname: String!  }  type Inventory {  id: ID!  product: Product  quan\_in\_stock: Int  sales: Int  }  type Product {  id: ID!  category: Category  title: String!  actor: String  price: Float!  special: Int  common\_prod\_id: Int  inventory: Inventory  }  type Orderline {  id: ID!  order: Order  product: Product  quantity: Int  orderdate: DateTime  }  type Order {  id: ID!  orderdate: DateTime  customer: Customer  netamount: Float!  tax: Float!  totalamount: Float!  orderline: Orderline  }  type Customer {  id: ID!  firstname: String!  lastname: String!  address1: String!  address2: String!  city: String!  state: String!  zip: String!  country: String!  region: String!  email: String!  phone: String!  creditcardtype: String  creditcard: String  creditcardexpiration: DateTime  username: String!  password: String!  age: Int!  income: Float  gender: String!  } |

Q5.

|  |
| --- |
| const server = new GraphQLServer({  typeDefs: './schema.graphql',  resolvers,  context: {  prisma  },  })  server.start(() => console.log('Server is running on http://localhost:4000')) |

Q3. Query



|  |  |
| --- | --- |
| query {  getOrderlines {  id  order{  id  orderdate  netamount  tax  totalamount  customer{  id  firstname  lastname  }  }  product{  id  title  price  special  category  {  id  categoryname  }  }    }  } | {  "data": {  "getOrderlines": [  {  "id": "cjthj2mfy001t07203r0el4zi",  "order": {  "id": "cjthj0dt000170720aj43eyt5",  "orderdate": "2019-03-18T00:00:00.000Z",  "netamount": 123,  "tax": 1,  "totalamount": 135.3,  "customer": {  "id": "cjthizewb000w072056d3qnsd",  "firstname": "Erika",  "lastname": "Secillano"  },  "orderline": {  "id": "cjthj2mfy001t07203r0el4zi"  }  },  "product": {  "id": "cjthj1kdw001k0720g4vay6q0",  "title": "T-Shirt",  "price": 19.99,  "special": 1,  "category": {  "id": "cjthj0zsd001e0720e746c6ay",  "categoryname": "Clothes"  }  }  }  ]  }  } |

Q.4

|  |  |
| --- | --- |
|  | mutation {  createCustomer (  firstname: "Erika"  lastname: "Secillano"  address1: "123"  address2: "Street"  city: "Dublin"  state: "Co. Dublin"  zip: "EUDUB18"  country: "Ireland"  region: "Europe"  email: "erika@gmail.com"  phone: "0811111111"  creditcardtype: "Visa"  creditcard: "232343454"  creditcardexpiration: "2025-12"  username: "erika"  password: "mae"  age: 21  income: 0.01  gender: "Female"  ) {  id  firstname  lastname  username  phone  country  email  }  }  {  "data": {  "createCustomer": {  "id": "cjthizewb000w072056d3qnsd",  "firstname": "Erika",  "lastname": "Secillano",  "username": "erika",  "phone": "0811111111",  "country": "Ireland",  "email": "erika@gmail.com"  }  }  } |
|  | mutation {  createOrder(  orderdate: "2019-03-18"  customerId: "cjtc1zs2w000h0732qe398lym"  netamount: 123  tax: 1.00  totalamount: 135.3    ) {  id  orderdate  totalamount  customer {  firstname  lastname  }  }  }  {  "data": {  "createOrder": {  "id": "cjthj0dt000170720aj43eyt5",  "orderdate": "2019-03-18T00:00:00.000Z",  "totalamount": 135.3,  "customer": {  "firstname": "Erika",  "lastname": "Secillano"  }  }  }  } |
|  | mutation {  createProduct (  categoryId: "cjthj0zsd001e0720e746c6ay"  title: "Bear Jumpsuit"  actor: "Bershka"  price: 16.00  special: 9  common\_prod\_id: 12  quan\_in\_stock: 44  ) {  id  title  price  category {  id  categoryname  }  inventory {  id  quan\_in\_stock  sales  }  }  }  {  "data": {  "createProduct": {  "id": "cjtrqy33y009u07209ncd22fl",  "title": "Bear Jumpsuit",  "price": 16,  "category": {  "id": "cjthj0zsd001e0720e746c6ay",  "categoryname": "Clothes"  },  "inventory": null  }  }  }  --------------  mutation {  createProduct (  categoryId: "cjthj0zsd001e0720e746c6ay"  title: "Skirt"  actor: "Bershka"  price: 19.99  special: 1.00  common\_prod\_id: 11  quan\_in\_stock: 100  ) {  id  title  price  category {  id  categoryname  }  inventory {  id  quan\_in\_stock  sales  }  }  }  {  "data": {  "createProduct": {  "id": "cjtqhy3v7004o07206i5ly6io",  "title": "Skirt",  "price": 19.99,  "category": {  "id": "cjthj0zsd001e0720e746c6ay",  "categoryname": "Clothes"  },  "inventory": null  }  }  } |
|  | mutation {  createOrderline(  orderId: "cjthj0dt000170720aj43eyt5"  productId: "cjthj1kdw001k0720g4vay6q0"  quantity: 2  orderdate: "2019-03-20"  ) {  quantity  orderdate  order {  id  totalamount  }  product {  id  title  }  }  }  {  "data": {  "createOrderline": {  "quantity": 2,  "orderdate": "2019-03-20T00:00:00.000Z",  "order": {  "id": "cjthj0dt000170720aj43eyt5",  "totalamount": 135.3  },  "product": {  "id": "cjthj1kdw001k0720g4vay6q0",  "title": "T-Shirt"  }  }  }  } |
|  | mutation {  createCategory ( categoryname: "Clothes" ) {  id  categoryname  }  }  {  "data": {  "createCategory": {  "id": "cjthj0zsd001e0720e746c6ay",  "categoryname": "Clothes"  }  }  } |