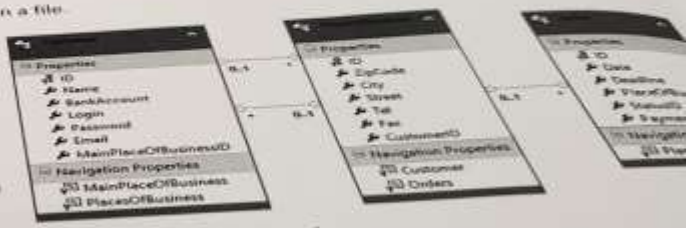
___ When using the non-clustered index, it is an additional indirection to read the actual records.
___ The optimization of learned MongoDB does not use statistics.
___ Unlocking the XML abbreviation: Extensible Markup Language
___ We can use namespaces in XML.
___ In an XPath expression, we refer to child elements with ( ).
___ We use XML correctly for data storage if we do not want to bother with migration.
___ No matter how complicated an SQL query is, the query optimizer always finds the best solution.
___ A database cursor is always linked to a query.
___ An MS SQL database table is always located in a file.

3. Consider the classes below and the relationships among them. (This is in fact an Entity Framework EDM model.) Write C# code using Linq expressions for the following tasks. The code lines begin with a collection (list, array) instance. You should only write a single C# statement and no for cycle, no writing to console, etc. (1+1+2p)



a) Find the customers with name "John ..." (whose name begin like this):

```
customers.
```

b) What is the date of the newest order:

```
orders.
```

c) Count how many place of businesses are without any associated order:

```
placesOfBusiness.
```

```
if (...) return NotFound();
.....
return NoContent();
}
```

b) In the context of the above task, provide a sample http request for updating an Employee resource! Only provide the https verb and the url (there is no need to provide the body). (1p)

c) When designing a REST API, which http verb is the right solution for creating a new resource? (1p)

d) Demonstrate with an example how the client can provide sorting related information when querying data via a REST based API following REST best practices. Provide the answer in a single line. (1p)

(12p)

4. Match the EJB transaction attributes with their meanings!
A: REQUIRED, B: REQUIRES_NEW, C: NEVER, D: SUPPORTS, E: NOT_SUPPORTED, F: MANDATORY (Hint)

If there is no active transaction when calling the method, a transaction will start. If an active transaction exists, the method joins the current transaction: ____

If there is no active transaction when calling the method, no transaction will be started. If an active transaction exists, the method joins the current transaction: ____

If there is no active transaction when calling the method, an exception is thrown. If an active transaction exists, the method joins the current transaction: ____

If there is no active transaction when calling the method, no transaction will be started. If an active transaction exists, it will be suspended, and this method runs outside of that transaction: ____

If there is no active transaction when calling the method, a transaction will start. If an active transaction exists, it will be suspended, and this method runs in a new transaction, which is independent from the original, suspended transaction: ____

If there is no active transaction when calling the method, no transaction will be started. If an active transaction exists, an exception is thrown: ____

5. Answer the following questions regarding SOAP/REST/GraphQL and dependency injection! (8p)

a) You are implementing a service that maintains data about employees. The service is implemented as an ASP.NET Core application with a REST based API. Your task is the implementation of a WebAPI controller function which can update (fully replace) the data of an existing user. Make sure to follow REST best practices! You can assume that an *Employee* model class already exists.

Provide the function header (and only that!) of the appropriate WebAPI controller function for the above task on the dotted lines, to which the method body following the dotted lines belong! (2p)

.....

.....

```
{
    if (...) return NotFound();
    .....
    return NoContent();
```

the context of the above task, provide a sample http request for updating an Employee resource! Only ovide the https verb and the url (there is no need to provide the body).

en designing a REST API, which http verb is the right solution for creating a new resource?

monstrate with an example how the client can provide sorting related information when querying da a REST based API following REST best practices. Provide the answer in a single line.

e) Define the problem of "overfetching"! How does it relate to GraphQL? Provide a 1-1 sentence answer for these questions! (2p)

Problem of overfetching is that

6. We have an Address JPA entity that stores the data of an address:

```
String country; //2-letter ISO code of the country
String city;
String street;
String nr; //house number
String zipCode;
```

Write a spring bean with a method that handles renaming of streets: it modifies the street of those addresses, that match the country/zipcode/street triple passed as input argument. The new street name is passed as argument as well. If you find it useful, you can write a Spring Data Repository interface and call it from the method. (8p)

```
@Entity
@Setter
@Getter
public class Address {
        @Id
        @GeneratedValue(strategy = GenerationType.AUTO)

        private Integer id;
        private String country;
        private String city;
        private String street;
        private String nr;
        private String zipCode;
}

Public interface AddressRepository extends JpaRepository<Address, Integer> {
        List<Address> findByCountryandZipcodeandStreet(String country, String zipcode, String street);
}

@Service
public class AddressService {
        @Autowired
        private final AddressRepository addressRepository;

        @Transactional
        public void renamingStreetofAddress(String country, String zipcode, String street, String newStreet) {
                List<Address> addresses = AddressRepository.findByCountryandZipcodeandStreet(country, zipcode, street);
                for(Address address : addresses){
                        address.setStreet(newStreet);
                }
                addressRepository.saveAll(addresses);
        }
}
```