10. Új kolléga érkezik a céghez, aki még nem dolgozott objektum orientált környezetben. Magyarázza el neki az OOP lényegét!
– Milyen programozási módszertanokat ismer?
– Mi a különbség az objektum orientált és a szekvenciális programozás között?
– Mutassa be egy osztályok felépítését!
Kulcsszavak, fogalmak:
– Zártság, öröklődés, polimorfizmus, interface-ek, virtuális metódusok
– Jellemzők, metódusok, láthatósági szintek

I'd like to explain you the OOP which stands for Object Oriented Programming. First, the basics. When using the OOP you create objects generally defined as classes and define attributes and methods to them. For example if you create the Student class you can set to it different attributes like name, age, birth date and methods like add grade. A commonly used method of the classes (objects) is the constructor. You can call this method at any part of the program to create an entity (student entity) of the class with the methods and attributes which were defined previously.

Visibility:

You can set different visibility levels to the different objects. There are three levels:

Public – In this case the object is accessible in the whole program.

Protected – The object is accessible directly through any other object that are related to that object.

Private – The methods and attributes are only visible in their own class.

Inheritance:

We can avoid creating the same class again and again with the process of inheritance. There is inheritance between the base/parent/superclass and derived/child/subclass. The child class inherits the parent's properties and behaviour. This is good because we don't have to write them again but we can define the methods more specially or override them.

Polymorphism:

We can expend our classes with new methods or attributes. For example when a child class inherit an attribute from its parent class. This attribute can be called/modified in the child class using references. Thus this attribute exists in two different class which create polymorphism.

Interfaces:

These are tools that allow communications between different classes. The advantage of the interface is that it avoids the one parent limitation (that exists between classes) and thus a child class can have many interfaces which can define various attributes and methods to it.

Virtual method:

With virtual key word you can make a method virtual. That means children of the parent class that have the virtual method can override the method for its own use.

The other commonly used programming method is the sequential programming. When you use this method you create the codes row by row. This method is mostly used in scientific fields to create less error prone code. One of the most basic form of this programming is the waterfall methodology.