Programming Languages

Exercise 01

16-124-836
Marcel Zauder

1.1 What is a Programming Language

Louden defined a programming language to be *a notational system for describing computation in a machine-readable and human-readable form.*
In my opinion a more precise definition for a programming language would be: *a set of semantics (commands/instructions) and syntax, with which a problem-solving program or algorithm can be created.*

1.2 What is the difference between functional and logic programming style? Provide a Programming Language as an example for each type

A functional programming language uses functions and composition of functions. So function also can be used as parameter or output from another function. An example for a functional programming language are *Meta Language (ML)* or *Haskell.*
On the other hand, a logic programming language is based on mathematical logic, hence also called a declarative language. Therefore instead of using a sequence of orders, as it is the case for an imperative programming language, here we use axioms, facts, and rules. One of it's best use is for searching, because the solution for queries can be derived from the rules and facts defined before. An example for this type is *Prolog.*

1.3 What is SQL and how can one print "hello world" in SQL?

SQL is a standard QUERY language which is used for storing, manipulating and retrieving data in databases. It has become a standard of the American National Standards Institute and of the International Organization for Standardization for executing operations on databases.
To create a "Hello World" program in SQL we first need to create a table, give it a name (*HELLO*) and define a variable (*TXT*) and which type of content it stores (a *CHAR(11)* field):

```
CREATE TABLE HELLO (TXT CHAR(11))
```

Then we insert the character sequence 'Hello World' in the variable *TXT*:

```
INSERT INTO HELLO VALUES('Hello World')
```

In the end, we want to select everything from the table which prints the character sequence:

```
SELECT * FROM HELLO
```

1.4 What are the key innovations of the Object Oriented Languages?

A major key idea was to create a programming language in which models of real world objects could be created. Classes are used to create an abstraction of a given type of object, which contains all the data and procedures associated with the object itself. The Objects are then instances of those classes, whereas it is possible to create multiple instances of one class.
Another feature of OOLs is the concept of inheritance, with which we can define objects/classes upon other objects/classes. With this we can create a class *car* and make

subclasses like brands (Mercedes, Audi, Volkswagen, etc.), which are still considered as cars, but have their own specification and possible additional features.
Another concept is encapsulation, which is the idea, of data and methods being only available to the unit which needs to know about these attributes. Another object outside of this unit does not need to access these variables or methods directly. If an access is needed from outside the class, public *getter* and *setter* methods are integrated, so the values of the private variables can be changed also from the outside. Additionally in those *getter* and *setter* methods different rules can be applied such that not all values might be a valid one for the data, which therefore can prevent false behaviour in the class itself.