# 100. Mainstream languages

## Description

Dette her kapitel indeholder forslag til hvordan vi evt. kan beskrive sproglige mekanismer i andre sprog. Der er planlagt et afsnit for hver kapitel. Og hver et sådant afvist har underkapitler for de forskellige sprog.
Så det her kapitel *101. Mainstream languages* er kun en placeholder for disse afsnit indtil de er klar til at blive flyttet ind under hovedkapitlerne.
The text below, may be placed in chapter *1. Introduction*.

In this book, the programming language qBeta is used for specifying programs and program fragments. In practice there is a large number of programming languages being used by industry. We thus plan to conclude each/most chapters in this book by showing how the language mechanisms and some of the examples may be expressed in some of the mainstream object-oriented programming languages.

To some extent, most mainstream languages have support for the common mechanisms of object-oriented languages, but there are som important differences. They also differ more or less with respect to syntax.

Some teachers and students may find it confusing that students should have to understand the syntax and semantics of mechanisms in different languages. We do believe that it is important that students get an understanding of language mechanisms in general and how they differ in various programming languages. In addition, students should be able to express programs independent of which concrete syntax to use. Of course when you write a program in a given language, its syntax matters, but syntax is easy to learn if you understand the language mechanisms.

By the term mainstream languages, we refer to a list of languages that are widely used in practice. There are many such languages so we will only be able to cover a few of these. The plan is to cover C++, Java, C# and Python. Other languages like Javascript, PHP, Typescript might have been included.