

B4M36DS2, BE4M36DS2: **Database Systems 2**

<https://cw.fel.cvut.cz/b231/courses/b4m36ds2/>

Practical Class 4

# XML databases

**Yuliia Prokop**

[prokoyul@fel.cvut.cz](mailto:prokoyul@fel.cvut.cz)

16. 10. 2022

Author: Martin Svoboda

[martin.svoboda@matfyz.cuni.cz](mailto:martin.svoboda@matfyz.cuni.cz)

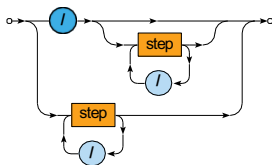
**Czech Technical University in Prague**, Faculty of Electrical Engineering



# Path Expressions

## Path expression

- Absolute / relative paths



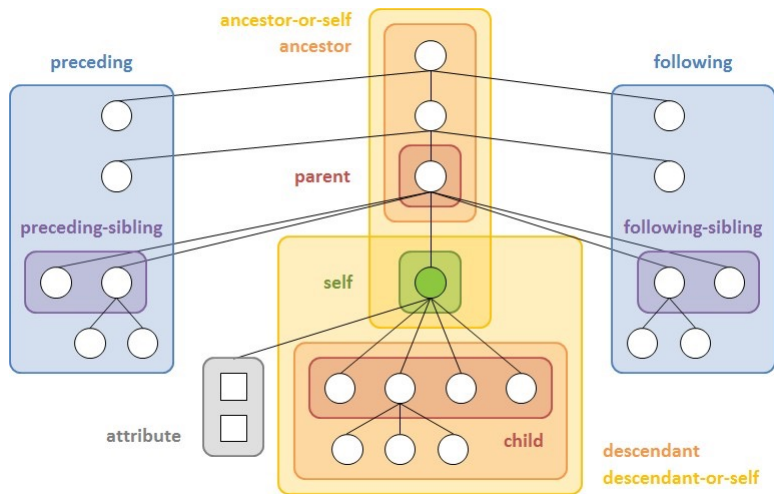
# Path Expressions

## Steps of path expressions



- **Axis**
  - Specifies the **relation of nodes** to be selected for a given node
- **Node test**
  - **Basic condition** the selected nodes must further satisfy
- **Predicates**
  - **Advanced conditions** the selected nodes must further satisfy

# Path Expressions: Axes



# Exercise 1

Express the following XPath queries

- **Names of all airline companies** (whole `airline` elements)
- **Full names of all airports** (just text content)
- **Codes of all airports** (their values)
- **The last ticket of the third flight** (in the document order)
- **Distinct codes of flight ticket classes** (without duplicities)

## Exercise 2

Express the following XPath query

- **Flight numbers operated by A6-EOQ aircraft on 2019-10-13**

# Exercise 3

Express the following XPath query

- **Flights with at least one first class ticket (*F*) or business class ticket (*C*)**

# Exercise 4

Express the following XPath query

- **Flights without any first class ticket (*F*) as well as any business class ticket (*C*)**
  - Include only flights with at least one ticket

Incorrect:

```
/airport/flights/flight[.//ticket][.//@class != "F" and .//@class != "C"]
```



# Exercise 5

Express the following XPath query

- **Numbers of flights that depart on *2019-10-18* or any date later and that have no aircraft assigned yet**

# Exercise 6

Express the following XPath query

- **Lines with duration above the overall average**

# Exercise 7

Express the following XPath query

- **Overall number of flights heading to any airport in Germany (DEU) on 2019-10-18**

# Exercise 8

Express the following XPath query

- **Passenger name on the very last ticket in the entire file**

Incorrect:

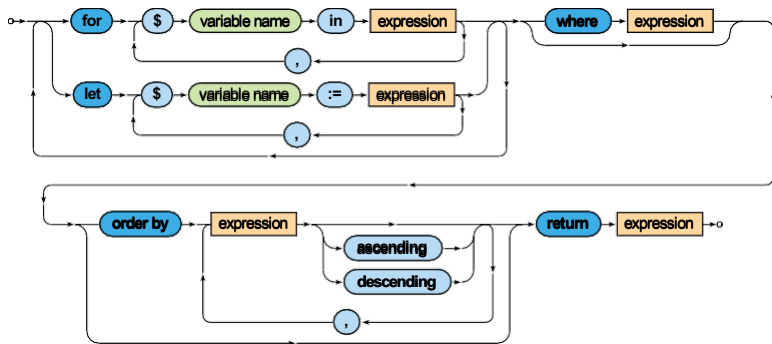
```
//ticket[last()]/text()  
//ticket[position() = last()]/text()  
/descendant-or-self::node()/ticket[position() = last()]/text()
```

Jitka Schneiderová  
Pavel Liška  
Jiří Menzel

# FLWOR Expressions

## FLWOR

- Versatile construct allowing for **iterations over sequences**



# Conditional Expressions

## Condition

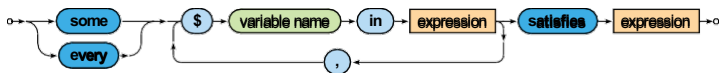
- Note that the else branch is compulsory
  - Empty sequence ( ) can be returned if needed



# Quantified Expressions

## Quantifier

- Returns `true` if and only if...
  - in case of `some` **at least one item**
  - in case of `every` **all the items**
- ... of a given sequence/s **satisfy the provided condition**



# Exercise 9

Express the following XQuery query

- **Flights heading to any airport in Germany (DEU) on 2019-10-18**

## XPath

//flight

[@date = "2019-10-18"]

[line = //line[arrival/airport/@country = "DEU"]/@number]

```
<flight date="2019-10-18">
  <line>LH1691</line>
</flight>
<flight date="2019-10-18">
  <line>LH1397</line>
</flight>
```



# Exercise 10

Express the following XQuery query

- **Sequence of lines longer than 60 minutes**
- Respect the following output structure

```
<line origin="airport-code" destination="airport-code">
  <code>line-number</code>
  <departure>departure-time</departure>
  <arrival>arrival-time</arrival>
</line>
...
```

- Propose two solutions using **direct / computed constructors** respectively

# Exercise 11

Express the following XQuery query

- **Names of airline companies such that all their flights are associated with aircrafts**

Result of the above expression applied to the above HTML file:

```
1 SAS Scandinavian Airlines  
2
```

# Exercise 12

Express the following XQuery query

- **Generate an XHTML table with data about flights from PRG**
  - Use `<i>Unknown</i>` when an aircraft is not assigned
  - Sort the flights using dates (descending order) and times of departure (ascending)

```
<table>
  <tr>
    <th>Date</th><th>Time</th><th>Number</th><th>Aircraft</th>
  </tr>
  <tr>
    <td>flight-date</td>
    <td>time-of-departure</td>
    <td>line-number</td>
    <td>aircraft-registration-or-unknown</td>
  </tr>
  ...
</table>
```

# Exercise 13

Express the following XQuery query

- **Names of passengers of *EK140* flights with at least average number of sold tickets over all *EK140* flights**
- **Respect the following output structure**

```
<passengers date="flight-date" tickets="number-of-tickets">  
  comma-separated-list-of-passenger-names  
</passengers>  
...
```

```
<passengers date="2019-10-12" tickets="3">  
  Ivan Trojan, Jiří Macháček, Jitka Schneiderová  
</passengers>  
<passengers date="2019-10-13" tickets="3">  
  Anna Geislerová, Tatiana Vilhelmová, Pavel Liška  
</passengers>  
<passengers date="2019-10-18" tickets="2">  
  Zdeněk Svěrák, Jiří Menzel  
</passengers>
```